JMLR: Workshop and Conference Proceedings vol 49:1–34, 2016

Learning Communities in the Presence of Errors

Konstantin Makarychev *Microsoft Research*

Yury Makarychev *TTIC*

Aravindan Vijayaraghavan *Northwestern University*

Abstract

We study the problem of learning communities in the presence of modeling errors and give robust recovery algorithms for the Stochastic Block Model (SBM). This model, which is also known as the Planted Partition Model, is widely used for community detection and graph partitioning in various fields, including machine learning, statistics, and social sciences. Many algorithms exist for learning communities in the Stochastic Block Model, but they do not work well in the presence of errors.

In this paper, we initiate the study of robust algorithms for partial recovery in SBM with modeling errors or noise. We consider graphs generated according to the Stochastic Block Model and then modified by an adversary. We allow two types of adversarial errors, Feige–Kilian or monotone errors, and edge outlier errors. Mossel, Neeman and Sly (STOC 2015) posed an open question about whether an almost exact recovery is possible when the adversary is allowed to add $o(n)$ edges. Our work answers this question affirmatively even in the case of $k > 2$ communities.

We then show that our algorithms work not only when the instances come from SBM, but also work when the instances come from any distribution of graphs that is εm close to SBM in the Kullback–Leibler divergence. This result also works in the presence of adversarial errors. Finally, we present almost tight lower bounds for two communities.

1. Introduction

Probabilistic models are ubiquitous in machine learning and widely used to find hidden structure in unlabeled data. The Stochastic Block Model (SBM), which is also known as Planted Partition Model, is the most studied probabilistic model for community detection and graph partitioning. There has been extensive research on the model in various fields, including machine learning, statistics, computer science, and social sciences over the last three decades (this research is summarized in Section 2). Until recently, research on SBM was focused on graphs with a poly-logarithmic, in the number of vertices, average degree. In the past few years, however, most of the research has shifted toward graphs with a constant average degree, and there has been significant progress in the understanding of the conditions under which a partial recovery is possible for such graphs in SBM. In particular, Massoulié (2014) and Mossel et al. (2012, 2013) have derived sharp conditions under which a partial recovery is possible for the case of two communities (clusters).

Yet most existing algorithms are not robust — they rely on the instance being drawn exactly from the given probabilistic model, and thus may fail in the presence of noise. For instance, while spectral algorithms have good provable guarantees for learning communities in SBM, they crucially rely on strong spectral properties of random graphs, which are brittle to a small amount of noise.

Algorithms most commonly employed in practice, for learning various probabilistic models are based on maximum likelihood estimation. They have many desirable properties from a statistical

standpoint, since maximum likelihood estimation is robust to many modeling errors. However, they do not typically have polynomial running time guarantees. This leads to a natural question: *Can we design algorithms for learning communities in SBM, which are both efficient (polynomial time) and tolerant to adversarial modeling errors?*

In this paper, we present polynomial-time algorithms that perform robust recovery for the Stochastic Block Model (SBM). Our algorithms work in the presence of different types of adversarial noise: edge outlier errors, monotone errors, and a modeling error measured in the Kullback–Liebler divergence. Our results give an affirmative answer to the question posed by Mossel et al. (2015), whether an almost exact recovery is possible when the adversary is allowed to add $o(n)$ edges.

Let us now recall the definition of the Stochastic Block Model[1].

Definition 1 (Stochastic Block Model) *A graph $G_{sb}(V, E_{sb})$ with $N = nk$ vertices is generated according to the Stochastic Block Model SBM(n, k, a, b) (where $a \geq b$) as follows:*

1. *There is a equipartition $P^* = (V_1^*, V_2^*, \ldots, V_k^*)$ of vertices V with $|V_i^*| = n$ for each $i \in [k]$.*
2. *For each $i \in [k]$, and for any two vertices $u, v \in V_i^*$, there is an edge $(u, v) \in E_{sb}$ with probability a/n.*
3. *For each $i, j \in [k]$ with $i \neq j$, and for any two vertices $u \in V_i^*, v \in V_j^*$, there is an edge $(u, v) \in E_{sb}$ with probability b/n.*

We denote the expected number of edges in G by m: $m = \frac{1}{2}(nka + nk(k-1)b)$.

We consider the Stochastic Block model with two types of modeling errors (adversarial noise): the outlier errors and Feige–Kilian (1998) (monotone) errors.

Definition 2 (Stochastic Block Model with modeling errors) *In the Stochastic Block Model SBM(n, k, a, b) with modeling errors, the graph $G(V, E)$ is generated as follows. First, a random graph $G_{sb} = (V, E_{sb})$ is sampled from the Stochastic Block Model SBM(n, k, a, b). Then the adversary adds some new edges to E' and removes some existing edges from E'. Specifically, the adversary may do the following:*

1. *In the Feige–Kilian or monotone error model, the adversary may add any edges within the clusters and remove any clusters between the clusters.*
2. *In the model with εm outliers, the adversary may choose $\varepsilon_1 \geq 0$ and $\varepsilon_2 \geq 0$ with $\varepsilon_1 + \varepsilon_2 \leq \varepsilon$, then add at most $\varepsilon_1 m$ edges between the clusters and remove at most $\varepsilon_2 m$ edges within the clusters.*
3. *In the model with two types of errors, the adversary may introduce both types of errors.*

Our goal is to find the unknown planted partition $(V_1^*, \ldots, V_k^*)$ given the graph $G = (V, E)$ from the Stochastic Block Model with modelling errors. However, in this paper, we focus on the regime where the exact recovery is impossible even information–theoretically. So we are interested in designing polynomial–time algorithms that partially recover the planted partition.

1. We note that some papers denote by n not the number of vertices in each cluster but the total number of vertices. Our SBM(n, k, a, b) model is the same as their SBM$'(kn, k, ka, kb)$ model.

Definition 3 *We say that a partition $V_1, \dots, V_k$ is δ-close to the planted partition $V_1^*, \dots, V_k^*$, if each cluster V_i has size exactly n and there is a permutation σ of indices such that*

$$\Big| \bigcup_{j=\sigma(i)} V_i^* \cap V_j \Big| \geq (1-\delta)kn.$$

An algorithm $(1-\delta)$-partially recovers the planted partition if it finds a partition that is δ–close to the planted partition.

We present two algorithms for partial recovery. The first algorithm can handle instances with both monotone and outlier errors, while the second algorithm handles only instances with outlier errors. The second algorithm also has stronger requirements on a and b. However, it has a much better recovery guarantee.

Theorem 4 (First Algorithm) *Consider the stochastic block model SBM(n, k, a, b) with εm outliers and monotone errors, and suppose $a + b(k-1) \geq C_0$ for some universal constant $C_0 > 1$. There is a polynomial-time algorithm that $(1-\delta)$-partially recovers the planted partition given an instance of the model, where*

$$\delta = O\Big(\frac{\sqrt{a+b(k-1)}}{a-b} + \frac{\varepsilon\,(a+b(k-1))}{a-b}\Big).$$

The algorithm succeeds with probability at least $1 - 2\exp(-2N)$ over the randomness of the instance.

Furthermore, for any $\eta \in (1/(a+b(k-1)), \frac{1}{2})$, with probability at least $1 - 2\exp(-\eta m)$, the algorithm $(1-\delta')$-partially recovers the planted partition with

$$\delta' = O\Big(\frac{(\varepsilon + \sqrt{\eta})\,(a+b(k-1))}{a-b}\Big).$$

For the case of two communities ($k = 2$), we prove that the result of Theorem 5 is asymptotically optimal (see Theorem 27). We also note that in the special case of $k = 2$ communities, the analysis of the algorithm due to Guédon and Vershynin (2014) can be adapted to obtain similar results (up to constants). However, their approach breaks down for $k \geq 3$ communities (see Section 1.1 for details).

Theorem 5 (Second Algorithm) *Consider the stochastic block model SBM(n, k, a, b) with εm outliers (without any monotone modelling errors), and suppose $a + b(k-1) \geq 2C_0$ for some universal constant $C_0 > 1$. Assume that*

$$\frac{\sqrt{a+b(k-1)}}{a-b} + \frac{\varepsilon\,(a+b(k-1))}{a-b} \leq c/k,$$

where $c > 0$ is some absolute constant. There is a randomized polynomial-time algorithm that does the following. Let $\delta_0 \geq ke^{-\frac{(a-b)^2}{100a}}$ and $\delta = O(\delta_0 + \frac{\varepsilon m}{(a-b)kn})$. The algorithm $(1-\delta)$-partially recovers the planted partition with probability at least $1 - 3\exp(-\delta_0 kn/6)$ over the randomness of the instance and random bits used by the algorithm.

In the above theorems C_0 is some universal constant that lower bounds the average degree of a vertex ($C_0 = 11$ suffices). We do not make any efforts to optimize the constant C_0, for ease of exposition. Let us compare the performance of our algorithms to the performance of the state of the art algorithms for the Stochastic Block Model.

- If no adversarial noise is present, our first algorithm works under the same condition on parameters a, b and k:

$$\frac{(a-b)^2}{a+b(k-1)} > C \quad \text{for some absolute constant } C$$

  as the algorithm by Abbe and Sandon (2015) for SBM (the absolute constant C in our condition is different from that by Abbe and Sandon (2015)).

- Our second algorithm achieves the same recovery rate as the algorithm of Chin et al. (2015) for SBM (that, however, is not surprising, since our second algorithm uses the "boosting" technique developed by Chin et al. (2015)) .

We note that, unlike many previously known algorithms for the Stochastic Block Model, our recovery algorithms fail with probability that is exponentially small in ηm. In particular, this implies that the algorithm from Theorem 4 works even if we sample the initial graph G_{sb} not from SBM(n, k, a, b) but from a distribution that is (λm)-close to SBM(n, k, a, b) in the KL-divergence distance (see Section 7).

Theorem 6 *Let $\mathcal{G}$ be a distribution that is λm close to SBM(n, k, a, b) in the KL divergence: $D_{KL}(\mathcal{G}, \text{SBM}(n, k, a, b)) \le \lambda m$. Suppose that $a + b(k-1) \ge C_0$ for some universal constant $C_0 > 1$. Consider a model where the graph is sampled from the distribution $\mathcal{G}$ and then the adversary introduces monotone and outlier modeling errors (with parameter εm). For any $\eta > 0$, the algorithm from Theorem 4 works in this model with the same recovery guarantee:*

$$\delta = O\Big(\frac{(\varepsilon + \sqrt{\eta})\,(a + b(k-1))}{a-b} + \frac{\sqrt{a+b(k-1)}}{(a-b)}\Big).$$

It may fail with probability at most $2\lambda/\eta$.

Related Work Cai and Li (2015) proposed a stochastic block model with *outlier vertices*. In their model, the graph is generated as follows: first a graph is drawn according to SBM(n, k, a, b), then the adversary adds to the graph t extra vertices and an arbitrary set of edges incedent on these t vertices. Cai and Li (2015) give an SDP algorithm for partially recovering the communities. Their algorithm works for $a \ge C \log n$. For $a, b = O(\log n)$, it can tolerate up to $O(\log n)$ vertex outliers. Note that if $a + b(k-1) \ge C \log n$ (as in their result), robustness to edge errors is more general than robustness to vertex outliers.

In a concurrent and independent work, Moitra et al. (2015) study the problem of weak recovery in a SBM with $k = 2$ communities in the presence of monotone errors as in Feige and Kilian (1998). They do not consider the case of $k > 2$ communities; they also do not consider adversarial errors and modeling errors in the KL divergence. Due to the space limit, we give a detailed overview of related work (including (Cai and Li, 2015) and (Moitra et al., 2015)) in Section 2.

1.1. Techniques

Let us briefly describe our first algorithm. The algorithms is based on semidefinite programming (SDP). We use a variant of the standard SDP relaxation for the k-Partitioning Problem (see e.g. Krauthgamer et al. (2009)). The SDP solution assigns a unit vector $\bar{u}$ to each vertex u of the graph (see Section 3 for details). We prove that vectors $\{\bar{u}\}$ are clustered consistently with the community memberships: the vectors assigned to vertices in the same cluster are close to each other (on average), while the vectors assigned to vertices in different clusters are far from each other (on average). It follows that each cluster V_i^* has a core $\mathrm{core}(i)$ such that all vertices in the core lie close to each other, vertices in different cores are far apart, and $\cup_i \mathrm{core}(i)$ contains all but a small fraction of the vertices (see Section 5).

We give a simple greedy algorithm that, given the SDP solution, finds a partition $V_1^*, \ldots, V_k^*$ of V close to the planted partition. The algorithm considers balls of some fixed small radius around vectors $\{\bar{u}\}$ and chooses the "heaviest" among them, the one that contains most vectors $\{\bar{u}\}$. It creates a cluster consisting of the vertices, whose vectors lie in the ball, and removes them and the corresponding vectors from the consideration. Then it iteratively processes the remaining vertices. The clustering algorithm is similar to the algorithm recently developed for a different clustering problem called Correlation Clustering (Makarychev et al., 2015). Unlike the algorithm in Makarychev et al. (2015), however, the algorithm in this work is robust to adversarial errors and modeling errors, and works in the sparse regime. Importantly, our geometric structural property holds even in the presence of adversarial noise, and the probability that a random graph from SBM does not satisfy it is exponentially small. As a result of this, the algorithm works even in the presence of outlier, monotone, and modeling errors.

To prove that our geometric structural property holds, we use, in particular, some techniques developed by Guédon and Vershynin (2014). However, we cannot merely rely on their result: Guédon and Vershynin prove that the best rank-$(2k-3)$ approximation $\hat{P}$ to the SDP solution matrix $\hat{Z}$ (the Gram matrix of the SDP vectors $\{\bar{u}\}$) is close to a particular rank-$(k-1)$ matrix, the matrix that encodes the planted partition. This property suffices when $k = 2$ — then the rank-1 matrix $\hat{P}$ defines a one dimensional solution $\{x_u : u \in V\}$, and the planted partition can be approximately recovered by thresholding numbers $\{x_u\}$. Moreover, it can be shown that this algorithm for $k = 2$ communities is robust to modeling errors. However, this approach works only when $k = 2$ and does not seem to extend to the case of $k > 2$ (in particular, Guédon and Vershynin only describe an algorithm for the case of $k = 2$). Therefore, instead of directly using the result by Guédon and Vershynin (2014), we use some of their ideas to prove that the SDP solution satisfies the geometric structural property (described above), which is quite different from that in Guédon and Vershynin (2014). This property enables us to easily recover the planted clustering.

Organization We start by presenting our SDP relaxation for the partition recovery problem (see Section 3). Then, in Section 4 we prove the geometric structural property. In Section 5, we present our first algorithm and prove Theorem 4. In Section 6, we show how to "boost" the performance of this algorithm by using the technique by Chin et al. (2015). This yields Theorem 5. We present Theorem 6 in Section 7 and describe our negative results in Appendix B. We give a detailed overview of prior work in Section 2.

2. Overview of Prior Work

We now review prior work on learning probabilistic models for graph partitioning while focusing on algorithms that give polynomial time guarantees. In what follows, C denotes a constant that is chosen to be sufficiently large.

Stochastic Block Models The Stochastic Block Model is the most widely studied probabilistic model for community detection and graph partitioning in different fields like machine learning, computer science, statistics and social sciences (see e.g. Bui et al. (1987); Holland et al. (1983); White et al. (1976); Fortunato (2010)). This model is also sometimes called the Planted Partitioning model and was studied in a series of papers, which among others include Dyer and Frieze (1986), Boppana (1987), Jerrum and Sorkin (1993), Dimitriou and Impagliazzo (1998), Condon and Karp (1999), McSherry (2001) and Coja-Oghlan (2006). The existing algorithmic guarantees for the Stochastic Block Model fall into three broad categories: *exact recovery, weak recovery, and partial recovery*.

For *exactly recovering* the communities, provable guarantees are known for many different algorithms like spectral algorithms, convex relaxations and belief propagation. These algorithms need sufficient difference between the average intra-cluster degree a and inter-cluster degree b, and a lower bound on the average degree $a + b = \Omega(\log n)$. For $k = 2$ clusters, Boppana (1987) used spectral techniques to give an algorithm that recovers the clusters when $a - b \geq C \cdot \sqrt{a \log n}$. Recently, Abbe et al. (2014) and Mossel et al. (2015) determined sharp thresholds for exact recovery in the case of $k = 2$ communities. The influential work of McSherry (2001) used spectral clustering to handle a more general class of stochastic block models with many clusters, and the guarantees have been subsequently improved in different parameter regimes of a, b, k by various works using both spectral techniques and convex relaxations (Chen et al., 2012; Ames, 2014; Vu, 2014; Wu et al., 2015; Perry and Wein, 2015).

The goal in *weak recovery* is to output a partition of the nodes which is positively correlated with the true partition with high probability. This problem was introduced by Coja-Oghlan (2010). Decelle et al. (2011) conjectured that there is a sharp phase transition in the case of $k = 2$ clusters depending on whether value of $\frac{(a-b)^2}{(a+b)} > 1$ or not, and this was settled independently by Massoulié (2014) and Mossel et al. (2012, 2013). It was also recently shown that semidefinite programs get close to this threshold (Montanari and Sen, 2015)[2]. The problem is still open for $k > 2$ communities, and the conjecture of Decelle et al. (2011) and Mossel et al. (2013) for larger k is that the clustering problem can be solved in polynomial time when $\frac{(a-b)^2}{a+(k-1)b} > 1$.

In *partial recovery*, the goal is to recover the clusters in the planted partitioning up to ηN vertices, i.e. up to ηN vertices are allowed to be misclassified in total (here η can be thought of as $o(1)$). Coja-Oghlan (2010) and Mossel et al. (2014) studied this problem for the case of $k = 2$ communities. Guédon and Vershynin (2014) analysed the semidefinite programming relaxation using the Grothendieck inequality to partially recover the communities (for $k = 2$) when $(a-b)^2 > C(a+b)/\eta^2$. These results were extended to the case of k-communities by Chin et al. (2015) and Abbe and Sandon (2015). The algorithm by Chin et al. (2015) recovers the communities up to η error when $\frac{(a-b)^2}{a} \geq Ck^2 \log(1/\eta)$.[3] These results were recently improved by Abbe and Sandon

2. The algorithm of Mossel et al. (2013) and Massoulié (2014) uses non-backtracking random walks.
3. In fact Chin et al. (2015) gives the stronger guarantee of recovering each of the clusters up to ηn vertices.

(2015) who gave algorithms and information-theoretic lower bounds for partial recovery in fairly general stochastic block models.

We note that the algorithm and analysis of Guédon and Vershynin (2014) can be adapted to work in the presence of monotone and adversarial errors for the case of $k = 2$ communities (see Section 1.1 for details). In a concurrent and independent work, Montanari and Sen (2015) (see revision 2 of their archive paper) observed that their algorithm for *testing* whether the input graph comes from the Erdős–Rényi distribution or a Stochastic Block Model with $k = 2$ communities also works in presence of $o(m)$ edge outlier errors. Their algorithm does not recover the clusters.

Semirandom models Semi-random models provide robust alternatives to average-case models by allowing much more structure then completely random instances. Research on semi-random models was initiated by Blum and Spencer (1995), who introduced and investigated semi-random models for k-coloring. Feige and Kilian (1998) studied a semi-random model for Minimum Bisection (two communities of size n each) that introduced the notion of a *monotone adversary*. The graph is generated in two steps: first a graph is generated according to SBM$(n, 2, a, b)$ and then an adversary is allowed to either add edges inside the clusters or delete some of the edges present between the clusters. They showed that semi-definite programs remain integral when $a - b \geq C \cdot \sqrt{a \log n}$. This was also extended to the case of k clusters by Chen et al. (2012) and Agarwal et al. (2015); Perry and Wein (2015).

In a concurrent and independent work, Moitra et al. (2015) consider the problem of weak recovery in a SBM with $k = 2$ communities in the presence of monotone errors as in Feige and Kilian (1998). Their main result is a statistical lower bound that indicates that the phase transition for weak recovery in SBM with $k = 2$ communities changes in the presence of monotone errors. They also present an algorithm that performs weak recovery for two communities in the presence of monotone errors.They do not consider the case of multiple communities ($k > 2$). They also do not consider adversarial errors and modeling errors in the KL divergence.

The results by Makarychev et al. (2012, 2014, 2015) use semi-definite programming to give algorithmic guarantees for various average-case models for graph partitioning and clustering problems. These works (Makarychev et al., 2012, 2014) consider probabilistic models for Balanced Cut (where the two clusters have roughly equal size) that are more general than stochastic block models, but they are incomparable to the models considered in this work. Besides, the focus of (Makarychev et al., 2012, 2014) is to find a Balanced Cut of small cost (the partitioning returned by the algorithm need not necessarily be close to the planted partitioning) and they make no structural assumptions on the graph inside the clusters. The algorithm in (Makarychev et al., 2012) also returns a partitioning closed to the planted partitioning under some mild assumptions about the expansion inside the clusters. However, it requires that $a = \tilde{\Omega}(\sqrt{\log n})$, while the focus of this work is the regime when a and b are constants.

Handling Modeling Errors The most related result in terms of modeling robustness is the recent work by Cai and Li (2015), who consider the stochastic block model in the presence of some outlier vertices. The graph is generated as follows: first a graph is drawn according to a stochastic block model $SBM(n, k, a, b)$[4]. Then, the adversary adds to the graph t outlier vertices and a set of arbitrary edges incedent on them. Cai and Li (2015) give an SDP-based algorithm for partially recovering the communities. Their algorithm works for $a \geq C \log n$ and

4. The authors also consider the case where communities can have different sizes as well.

$(a-b) > C\left(\sqrt{a\log n} + \sqrt{kb} + m\sqrt{k}\right)$ (see Condition 3.1 in Theorem 3.1). For $a, b = O(\log n)$, it can tolerate up to $O(\log n)$ outliers. To handle up to εn outliers, the algorithm needs the graph to be very dense i.e. $a, b = \Omega(\varepsilon n)$.

In the regime when $a + b(k-1) \geq C \log n$, robustness to edge outliers is more general than robustness to vertex outliers. (Because, in this regime, the degree of each vertex is tightly concentrated around $a + (k-1)b$, hence one can remove all outlier vertices whose degree is substantially larger than $a+(k-1)b$ in the given graph G. After that the number of error edges will be $O(t(a+b(k-1)))$. Using the results in our work, we can handle the case when an ε fraction of the vertices are corrupted since this corresponds to an ε fraction of the edges being corrupted in *our* outlier model. Additionally, our algorithm also performs partial recovery in the sparse regime (when $a, b = O(1)$).

Kumar and Kannan (2010) and Awasthi and Sheffet (2012) presented a spectral algorithm for clustering data that performs partial recovery as long the data satisfies some deterministic conditions (involving the spectral radius of the adjacency matrix), that are satisfied by instances that are generated by many probabilistic models for clusters. These deterministic conditions hold in graphs with degree $\Omega(\log n)$ and when the noise is more structured; in particular, they need the spectral norm of a matrix representing the errors to be small (this does not hold for adversarial modeling errors in general).

Finally, the work of Brubaker (2009) gave new algorithms for clustering data arising from a mixture of Gaussians when an $\varepsilon = O(1/(k \log^2 n))$ fraction of the data points are outliers. Surprisingly, Brubaker showed that this tolerance to noise can be achieved when the separation between the means is only a logarithmic factor more than the separation needed for learning gaussian mixtures with no noise (Kannan et al., 2005; Achlioptas and McSherry, 2005). While these results apply to very different problems in unsupervised learning, in the analogous regime, our algorithm works if up to an $\varepsilon = O(1)$ fraction of the observations come from errors. Finally, our results also handle large errors in the probabilistic model, when measured in the KL divergence (up to εm).

3. Preliminaries

3.1. Notation

Given an equipartition $(V_1^*, \ldots, V_k^*)$ of the vertices of $G(V, E)$, let $(V \times V)_{in}$ represent all the pairs of vertices inside the clusters, and $(V \times V)_{out}$ represent the pairs that go between the clusters. Similarly, let E_{in} be the edges inside the clusters, and E_{out} be the edges that go between the different clusters.

3.2. SDP Relaxation

Our partition recovery algorithms are based on semidefinite programming. In all our algorithms, we use the following basic SDP relaxation for the partition recovery problem (the SDP is presented in the vector form). For every vertex u in the graph, we have a vector variable $\bar{u}$ in the SDP relaxation.

$$\min \sum_{(u,v)\in E} \tfrac{1}{2}\|\bar{u}-\bar{v}\|^2 \tag{3.1}$$

s.t.

$$\|\bar{u}\|^2 = 1 \qquad \forall u \in V \tag{3.2}$$

$$\sum_{u,v\in V} \frac{1}{2}\|\bar{u}-\bar{v}\|^2 = n^2k(k-1) = N^2\left(1-\frac{1}{k}\right) \tag{3.3}$$

$$\langle \bar{u}, \bar{v}\rangle \geq 0 \qquad \forall u, v \in V \tag{3.4}$$

The summation in constraint (3.3) is over all N^2 pairs of vertices.

Our SDP relaxation is standard. Note that we do not use ℓ_2^2-triangle inequalities which are often used in SDP relaxations for graph partitioning problems. We also do not use strong spreading constaints (see e.g. Krauthgamer et al. (2009); Bansal et al. (2014)) and instead use a weaker constraint 3.3.

We denote the optimal value of this SDP relaxation by sdp. Consider the following feasible SDP solution corresponding to the planted partition. Assign $\bar{u} = e_i$ for all $u \in V_i^*$ and all i, where $e_1, \ldots, e_k$ is an orthonormal basis. It is easy to see that this is a feasible SDP solution. Its value is equal to the number of edges going between partitions. Since the value of the optimal SDP solution is at most the value of this solution,

$$\mathrm{sdp} \leq |\{(u,v) \in E : u \in V_i^*, v \in V_j^* \text{ for some } i \neq j\}|. \tag{3.5}$$

4. Structure of the Optimal SDP Solution

In this section, we analyze the geometric structure of the optimal SDP solution. We show that SDP vectors for vertices in the same cluster are close to each other (on average); SDP vectors for vertices in different clusters are far away from each other (on average).

We denote the average distances assigned by the SDP to pairs of vertices inside clusters and between clusters by α and β, respectively. Formally,

$$\alpha = \operatorname*{Avg}_{(u,v)\in(V\times V)_{in}} \tfrac{1}{2}\|\bar{u}-\bar{v}\|^2 \quad \text{and} \quad \beta = \operatorname*{Avg}_{(u,v)\in(V\times V)_{out}} \tfrac{1}{2}\|\bar{u}-\bar{v}\|^2.$$

It follows from constraint (3.3) that the values of α and β satisfy:

$$\alpha + (k-1)\beta = k-1. \tag{4.1}$$

In the following theorem, we prove that α is small and β is close to 1.

Theorem 7 *Let $G(V,E)$ be a graph generated according to the stochastic block model SBM(n,k,a,b) with εm outliers and arbitrary monotone errors. Suppose that $(a+b(k-1)) > C$ for some absolute constant C. Then, for every $s \geq 1$, the average intra-cluster distance α and inter-cluster distance β satisfy the following bounds with probability at least $1 - 2e^{-\frac{9s^2 N}{4+8s/\sqrt{a+b(k-1)}}}$:*

$$\alpha \leq \frac{c_7(\sqrt{a+b(k-1)})s}{a-b} + \frac{(a+b(k-1))\,\varepsilon}{a-b}, \tag{4.2}$$

and $\beta \geq 1 - \alpha/(k-1)$*, where* $c_7 \leq 6K_G + 4$ *is an absolute constant and* $K_G < 1.783$ *is the Grothendieck constant.*

Proof Denote $f(s) = e^{-\frac{9s^2 N}{4+8s/\sqrt{a+b(k-1)}}}$. For notational convenience, we assume that all vertices in the graph are ordered. Let G_{sb} be the graph generated in the stochastic block model $SBM(n,k,a,b)$ without the adversarial errors; and let G be the graph obtained from G_{sb} by introducing arbitrarily many monotone errors and at most εm non-monotone errors (here $m = (a+b(k-1))n/2$ is the expected number of edges in graphs from SBM(n,k,a,b)). Denote by $\mathrm{planted}(G)$ and $\mathrm{planted}(G_{sb})$ the cost of the planted partition in graphs G and G_{sb}, respectively. Denote by $\mathrm{sdp}(G_{sb}, \{\tilde{u}\})$ the cost of a feasible SDP solution $\{\tilde{u}\}$ in the graph G_{sb}. Let $\mathrm{sdp}(G)$ be the cost of the optimal SDP solution in G. Our goal is to estimate $\mathrm{planted}(G) - \mathrm{sdp}(G)$. Note that the value of the SDP relaxation is at most the value of the planted partition (see inequality (3.5)), thus $\mathrm{planted}(G) - \mathrm{sdp}(G) \geq 0$. We prove that with probability at least $1 - 2f(s)$,

$$\mathrm{planted}(G) - \mathrm{sdp}(G) \leq \frac{N\big(-\alpha(a-b) + c_7\sqrt{(a+(k-1)b)}s + 2\varepsilon m\big)}{2}. \tag{4.3}$$

This bound immediately implies the statement of the theorem: since $\mathrm{sdp}(G) \leq \mathrm{planted}(G)$, we have $\alpha(a-b) \leq c_7\sqrt{(a+(k-1)b)}s + 2\varepsilon m$. We first bound the value of $\mathrm{planted}(G_{sb}) - \mathrm{sdp}(G_{sb}, \{\bar{u}\})$ for the graph G_{sb}, where $\{\bar{u}\}$ is the optimal SDP solution for the graph G.

Lemma 8 *The following inequality holds with probability at least* $1 - 2f(s)$*:*

$$\mathrm{planted}(G_{sb}) - \mathrm{sdp}(G_{sb}, \{\bar{u}\}) \leq \frac{N\big(-\alpha(a-b) + c_7\sqrt{(a+(k-1)b)}s\big)}{2}. \tag{4.4}$$

Proof We upper bound $\mathrm{planted}(G_{sb})$. The expected size of the planted cut equals $\mathbb{E}[\mathrm{planted}(G_{sb})] = bN(k-1)/2$. Thus, by the Bernstein inequality,

$$\mathrm{planted}(G_{sb}) \leq \frac{bN(k-1)}{2} + 2\sqrt{a+b(k-1)}Ns \tag{4.5}$$

with probability at least $1 - f(s)$ (see Lemma 25 in Appendix A for details).

We now lower bound $\mathrm{sdp}(G_{sb}, \{\bar{u}\})$. Let $A = (a_{uv})$ be the adjacency matrix of G, and let $\mathbb{E}[A]$ be the expectation of the adjacency matrix. Denote $\Delta a_{uv} = a_{uv} - \mathbb{E}[a_{uv}]$. We use the following theorem, which is very similar to Lemma 4.1 in Guédon and Vershynin (2014). For completeness, we prove Theorem 9 in Appendix A.3.

Theorem 9 *Let* $G_{sb}(V,E)$ *be a graph generated according to the stochastic block model* $SBM(n,k,a,b)$*. Suppose* $a + (k-1)b \geq 11$*. Then, with probability* $1 - f(s)$ *the following inequality holds for all feasible SDP solutions* $\{\tilde{u}\}$*:*

$$\Big|\sum_{u<v} \Delta a_{uv}\|\tilde{u} - \tilde{v}\|^2\Big| \leq 6K_G\sqrt{a+b(k-1)}Ns. \tag{4.6}$$

For the rest of the proof we assume that inequalities (4.5) and (4.6) hold. This happens with probability at least $1 - 2f(s)$. We apply inequality (4.6) to the optimal SDP solution $\{\bar{u}\}$ for the graph G. We have

$$\mathrm{sdp}(G_{sb}, \{\bar{u}\}) = \frac{1}{2}\sum_{u<v} a_{uv}\|\bar{u} - \bar{v}\|^2 \geq \frac{1}{2}\sum_{u<v} \mathbb{E}[a_{uv}]\|\bar{u} - \bar{v}\|^2 - 3K_G\sqrt{a+b(k-1)}Ns.$$

The set of edges E_{sb} comes from the stochastic block model, hence $\mathbb{E}[a_{uv}] = a/n$, if $(u,v) \in (V \times V)_{in}$; and $\mathbb{E}[a_{uv}] = b/n$, if $(u,v) \in (V \times V)_{out}$. Therefore,

$$\frac{1}{2}\sum_{u<v} \mathbb{E}[a_{uv}]\|\bar{u}-\bar{v}\|^2 = \frac{a}{n}\sum_{\substack{(u,v)\in(V\times V)_{in}\\ u<v}} \frac{\|\bar{u}-\bar{v}\|^2}{2} + \frac{b}{n}\sum_{\substack{(u,v)\in(V\times V)_{out}\\ u<v}} \frac{\|\bar{u}-\bar{v}\|^2}{2}.$$

By the definition of α and β, the first term on the right hand side equals $(a/n)\cdot \alpha k n^2/2 = a\alpha N/2$; the second term equals $b\beta N(k-1)/2$. Using that $(k-1)\beta = (k-1) - \alpha$, we get

$$\begin{aligned} \mathrm{sdp}(G_{sb}, \{\bar{u}^*\}) &\geq \frac{a\alpha N + b\beta(k-1)N}{2} - 3K_G\sqrt{a+b(k-1)}Ns \\ &= \frac{(a-b)\alpha N + b(k-1)N}{2} - 3K_G\sqrt{a+b(k-1)}Ns. \end{aligned}$$

Combining this inequality with (4.5), we get bound (4.4). ■

Consider a sequence of operations – edge additions and edge removals – that transform the graph G_{sb} into the graph G. Let $G_0 = G_{sb}, \ldots, G_T = G$ be the sequence of graphs obtained after performing these operations. Observe that every time we make a monotone change the value of $\mathrm{planted}(G_t) - \mathrm{sdp}(G_t, \{\bar{u}\})$ does not increase: When we remove an edge between two vertices u and v in distinct clusters, we decrease $\mathrm{planted}(G_t)$ by 1 and $\mathrm{sdp}(G_t, \{\bar{u}\})$ by $\|\bar{u}-\bar{v}\|^2/2 = 1 - \langle \bar{u}, \bar{v}\rangle \leq 1$ (here we use the SDP constraint $\langle \bar{u}, \bar{v}\rangle \geq 0$). Similarly, when we add an edge between two vertices u and v from the same cluster, we do not change $\mathrm{planted}(G_t)$, but increase $\mathrm{sdp}(G_t, \{\bar{u}\})$ by $\|\bar{u}-\bar{v}\|^2/2 \geq 0$. When we add or remove a non-monotone edge, however, the value of $\mathrm{planted}(G_t) - \mathrm{sdp}(G_t, \{\bar{u}\})$ may increase by 1. Hence,

$$\begin{aligned} \mathrm{planted}(G) - \mathrm{sdp}(G, \{\bar{u}\}) \leq \mathrm{planted}(G_{sb}) - \mathrm{sdp}(G_{sb}, \{\bar{u}\}) + \varepsilon m \leq \\ \leq \frac{-(a-b)\alpha N + c_7\sqrt{a+b(k-1)}Ns + 2\varepsilon m}{2}. \end{aligned}$$

This completes the proof. ■

For $\eta \in (0, 1/2]$ and $s = \sqrt{\eta(a+b(k-1))}$, we get the following corollary.

Corollary 10 *Under conditions of Theorem 7, for some absolute constant c_{10}, and any $\eta \in [1/(a+b(k-1)), 1/2]$*

$$\mathbb{P}\left(\alpha \leq \frac{(a+b(k-1))\,(\varepsilon + c_{10}\sqrt{\eta})}{a-b}\right) \geq 1 - 2e^{-\eta m}. \tag{4.7}$$

5. First Algorithm

In this section, we present our first algorithm for a partial recovery. The algorithm given the SDP solution finds a partition $V_1, \ldots, V_k$ of V, which is close to the planted partition $V_1^*, \ldots, V_k^*$.

Definition 11 *Consider a feasible SDP solution $\{\bar{u}\}_{u\in V}$. We define the center $\bar{W}_i$ of cluster V_i^* as*

$$\bar{W}_i = \operatorname*{Avg}_{u\in V_i^*} \bar{u}.$$

For every vertex u let $R_u = \|\bar{u} - \bar{W}_i\|$, where W_i is the center of the cluster V_i^* that contains u. Let $\alpha_i = \frac{1}{2}\operatorname{Avg}_{u,v\in V_i^*}\|\bar{u}-\bar{v}\|^2$.

Definition 12 *Let $\rho = 1/5$ and $\Delta = 6\rho = 6/5$. We define the core of cluster V_i^* as*

$$\mathrm{core}(i) = \{u \in V_i^* : \|\bar{u} - \bar{W}_i\| < \rho\}.$$

We say that centers $\bar{W}_i$ and $\bar{W}_j$ are well-separated if $\|\bar{W}_i - \bar{W}_j\| \geq \Delta$. A set of clusters $\mathcal{S}$ is well-separated if centers of every two clusters in $\mathcal{S}$ are well-separated.

We show that most clusters V_i^* are well separated. First, we establish some basic properties of centers $\bar{W}_i$ and parameters α, α_i, β.

Lemma 13 *We have: (1) $\mathrm{Avg}_i\, \alpha_i = \alpha$; (2) $\mathrm{Avg}_{u \in V_i^*} R_u^2 = \alpha_i$; (3) $\mathrm{Avg}_{u \in V} R_u^2 = \alpha$; (4) $\mathrm{Avg}_{i \neq j} \langle W_i, W_j \rangle = 1 - \beta = \alpha/(k-1)$; (5) $\|\bar{W}_i\|^2 = 1 - \alpha_i$.*

Proof 1. This follows immediately from the definitions of α and α_i.

2. Write,

$$\begin{aligned} 2\alpha_i &= \operatorname*{Avg}_{u,v \in V_i^*} \|\bar{u} - \bar{v}\|^2 = \operatorname*{Avg}_{u,v \in V_i^*} \|(\bar{u} - \bar{W}_i) - (\bar{v} - \bar{W}_i)\|^2 \\ &= \operatorname*{Avg}_{u,v \in V_i^*} (\|\bar{u} - \bar{W}_i\|^2 + \|\bar{v} - \bar{W}_i\|^2) - 2 \operatorname*{Avg}_{u,v \in V_i^*} \langle \bar{u} - \bar{W}_i, \bar{v} - \bar{W}_i \rangle \\ &= 2 \operatorname*{Avg}_{u \in V_i^*} R_u^2 + 0 = 2 \operatorname*{Avg}_{u \in V_i^*} R_u^2. \end{aligned}$$

3. This follows from items 1 and 2.

4. Write,

$$\begin{aligned} \beta &= \frac{1}{2} \operatorname*{Avg}_{i \neq j} \operatorname*{Avg}_{u \in V_i^*, v \in V_j^*} \|\bar{u} - \bar{v}\|^2 = \operatorname*{Avg}_{i \neq j} \operatorname*{Avg}_{u \in V_i^*, v \in V_j^*} (1 - \langle \bar{u}, \bar{v} \rangle) \\ &= 1 - \operatorname*{Avg}_{i \neq j} \Big(\langle \operatorname*{Avg}_{u \in V_i^*} \bar{u}, \operatorname*{Avg}_{v \in V_j^*} \bar{v} \rangle \Big) = 1 - \operatorname*{Avg}_{i \neq j} \langle \bar{W}_i, \bar{W}_j \rangle. \end{aligned}$$

We get that $\mathrm{Avg}_{i \neq j} \langle \bar{W}_i, \bar{W}_j \rangle = 1 - \beta = \alpha/(k-1)$.

5. Write,

$$\alpha_i = \operatorname*{Avg}_{u \in V_i^*} R_u^2 = \operatorname*{Avg}_{u \in V_i^*} \|\bar{W}_i - \bar{u}\|^2 = \|\bar{W}_i\|^2 + 1 - 2 \operatorname*{Avg}_{u \in V_i^*} \langle \bar{W}_i, \bar{u} \rangle = 1 - \|\bar{W}_i\|^2.$$

The claim follows. ■

Lemma 14 *Let $V' = \bigcup_i V_i^* \setminus \mathrm{core}(i)$. That is, a vertex u lies in V' if it does not lie in the core of the cluster that contains it. Then*

$$|V'| \leq \frac{\alpha}{\rho^2}\, kn.$$

Proof Note that $u \in V'$ if and only if $R_u \geq \rho$, or, equivalently, $R_u^2 \geq \rho^2$. Since $\mathrm{Avg}_{u \in V} R_u^2 = \alpha$, we get by the Markov inequality that $|V'| \leq (\alpha/\rho^2)|V| = (\alpha/\rho^2)kn$. ■

We now prove that by removing at most a δ fraction of all clusters, we can obtain a well-separated set of clusters.

Lemma 15 *Let $\delta = 6\alpha/(2-\Delta^2)$. There exists a set $\mathcal{S} \subset \{V_1^*, \dots, V_k^*\}$ of well-separated clusters of size at least $(1-\delta)k$.*

Proof Let $\mu = \alpha/\delta$. From the Markov inequality and item 4 in Lemma 13, we get that there are at most

$$\frac{\alpha}{(k-1)\mu} \times \frac{k(k-1)}{2} = \frac{\alpha k}{2\mu} = \frac{\delta k}{2}$$

unordered pairs $\{i,j\}$ with $\langle W_i, W_j\rangle \geq \mu$. We choose one of the elements in each pair and remove the corresponding clusters. We obtain a set of clusters $\mathcal{S}_0$ of size at least $(1-\delta/2)k$. By the construction, for every distinct V_i^* and V_j^* in $\mathcal{S}_0$, we have $\langle W_i, W_j\rangle < \mu$.

Let $\mathcal{S}_1$ be the set of clusters V_i^* with $\alpha_i \leq 2\alpha/\delta$. By the Markov inequality and item 1 in Lemma 13, the set $\mathcal{S}_1$ contains at least $(1-\delta/2)k$ clusters.

Finally, let $\mathcal{S} = \mathcal{S}_0 \cap \mathcal{S}_1$. Clearly, $|\mathcal{S}| \geq (1-\delta)k$. For every two clusters V_i^* and V_j^* in $\mathcal{S}$, we have

$$\|\bar{W}_i - \bar{W}_j\|^2 = \|\bar{W}_i\|^2 + \|\bar{W}_j\|^2 - 2\langle \bar{W}_i, \bar{W}_j\rangle > (1-\alpha_i) + (1-\alpha_j) - 2\mu \geq 2(1-3\alpha/\delta) = \Delta^2.$$

■

Now we are ready to present a greedy algorithm that finds a partition close to the planted partition. The algorithm resembles the clustering algorithms by Charikar et al. (2001) and by Makarychev et al. (2015).

Recovery Algorithm

Input: an optimal SDP solution $\{\bar{u}\}_{u\in V}$.

Output: partition $V_1, \dots, V_{k'}$ of V into k' clusters (k' might not be equal to k).

$i = 1$; $\rho = 0.27$

Define an auxiliary graph $G_{aux} = (V, E_{aux})$ with $E_{aux} = \{(u,v) : \|\bar{u} - \bar{v}\| < 2\rho\}$

(note that, $(u,u) \in E_{aux}$ for every $u \in V$)

while $V \setminus (V_1 \cup \dots V_{i-1}) \neq \varnothing$

Let u be the vertex of maximum degree in $G_{aux}[V \setminus (V_1 \cup \dots V_{i-1})]$.

Let $V_i = \{v \notin V_1 \cup \dots \cup V_{i-1} : (u,v) \in E_{aux}\}$

If $|V_i| > n$, remove $|V_i| - n$ vertices from $|V_i|$ arbitrarily, so that $|V_i| = n$.

$i = i + 1$

return clusters $V_1, \dots, V_{i-1}$.

We will show now that the algorithm finds a "good" partition $V_1, \dots, V_k$. However, the clusters $V_1, \dots, V_k$ are not necessarily all of the same size. So we cannot say that the partition is δ-close to the planted partition according to Definition 3. We will be able, however, to prove that the partition is δ-close to the planted partition in the weak sense.

Definition 16 (cf. with Definition 3) *We say that a partition $V_1, \dots, V_{k'}$ is δ-close to the planted partition $V_1^*, \dots, V_k^*$ in the weak sense, if each cluster V_i has size at most n and there is a partial matching σ between $1, \dots, k$ and $1, \dots, k'$ such that*

$$\Big|\bigcup_{j=\sigma(i)} V_i^* \cap V_j\Big| \geq (1-\delta)kn$$

(the union is over all i such that $\sigma(i)$ is defined).

We say that $V_1, \dots, V_k$ is δ-close to $V_1^*, \dots, V_k^*$ in the strong sense, if it is δ-close according to Definition 3.

Theorem 17 *The Recovery Algorithm finds a partitioning $V_1, \dots, V_{k'}$ of V that is (72α)-close to the planted partition in the weak sense.*

Proof Let $\mathcal{S}$ be the set of clusters from Lemma 15. Consider a cluster V_j. We first show that it cannot intersect the cores of two distinct clusters $V_{i_1}^* \in \mathcal{S}$ and $V_{i_2}^* \in \mathcal{S}$. Assume to the contrary that it does. Let u_1 be a vertex in $\mathrm{core}(i_1) \cap V_j$, and u_2 be a vertex in $\mathrm{core}(i_2) \cap V_j$. Then $\|\bar{W}_{i_1} - \bar{u}_1\| < \rho$ and $\|\bar{W}_{i_2} - \bar{u}_2\| < \rho$. Since $u_1, u_2 \in V_j$, vertices u_1 and u_2 have a common neighbor u in the auxiliary graph $G_{aux} = (V, E_{aux})$, and, therefore, $\|\bar{u}_1 - \bar{u}_2\| < 4\rho$. We get that

$$\|\bar{W}_{i_1} - \bar{W}_{i_2}\| \le \|\bar{W}_{i_1} - \bar{u}_1\| + \|\bar{W}_{i_2} - \bar{u}_2\| + \|\bar{u}_1 - \bar{u}_2\| < 6\rho = \Delta,$$

which is impossible since $\mathcal{S}$ is a well separated set of clusters.

We now construct a partial matching σ between clusters V_i^* and V_j. We match every cluster $V_i^* \in \mathcal{S}$ with the first cluster V_j that intersects $\mathrm{core}(i)$ (then we let $\sigma(i) = j$). Since each vertex belongs to some V_j, we necessarily match every $V_i^* \in \mathcal{S}$ with some V_j. Moreover, we cannot match distinct clusters $V_{i_1}^*$ and $V_{i_2}^*$ with the same V_j because V_j cannot intersect both cores $\mathrm{core}(i_1)$ and $\mathrm{core}(i_2)$.

Let $Y = \bigcup_{V_i^* \in \mathcal{S}} \mathrm{core}(i)$ and $Z = V \setminus Y$. By Lemmas 14 and 15,

$$|Z| \le \Big|\bigcup_i V_i^* \setminus \mathrm{core}(i)\Big| + \Big|\bigcup_{V_i^* \notin \mathcal{S}} V_i^*\Big| \le \left(\frac{1}{\rho^2} + \frac{6}{2 - \Delta^2}\right) \alpha k n < 36\alpha k n.$$

Consider a cluster V_i^* and the matching cluster V_j. As we proved, V_j does not intersect $\mathrm{core}(i')$ of any $V_{i'} \in \mathcal{S}$ other than V_i. Therefore, $V_j \subset \mathrm{core}(i) \cup Z$. We now show that

$$|V_i^* \cap V_j| \ge |\mathrm{core}(i)| - |Z \cap V_j|.$$

Observe that every two vertices $v_1, v_2 \in \mathrm{core}(i)$ are connected with an edge in E_{aux} since

$$\|\bar{v}_1 - \bar{v}_2\| \le \|\bar{v}_1 - \bar{W}_i\| + \|\bar{v}_2 - \bar{W}_i\| < 2\rho.$$

In particular, every vertex $v \in \mathrm{core}(i)$ has degree at least $|\mathrm{core}(i)|$ in $G_{aux}[V \setminus (V_1 \cup \dots V_{j-1})]$. Let u be the vertex that we chose in iteration j. Since u is a vertex of maximum degree in $G_{aux}[V \setminus (V_1 \cup \dots V_{j-1})]$, it must have degree at least $|\mathrm{core}(i)|$. Now, either V_i consists of all neighbors of u in $G_{aux}[V \setminus (V_1 \cup \dots V_{j-1})]$ then $|V_j| \ge |\mathrm{core}(i)|$, or we removed some vertices from V_j because it contained more than n vertices, then $|V_j| = n \ge |\mathrm{core}(i)|$. In either case, $|V_j| \ge |\mathrm{core}(i)|$. We have,

$$|V_i^* \cap V_j| \ge |\mathrm{core}(i) \cap V_j| = |V_j| - |V_j \setminus \mathrm{core}(i)| = |V_j| - |V_j \cap Z| \ge |\mathrm{core}(i)| - |V_j \cap Z|.$$

Finally, using that all sets $V_j \cap Z$ are disjoint, we get

$$\sum_{j=\sigma(i)} |V_i^* \cap V_j| \ge \Big(\sum_{V_i^* \in \mathcal{S}} \mathrm{core}(i)\Big) - |Z| = |Y| - |Z| = |V| - 2|Z| \ge (1 - 72\alpha)kn.$$

■

Lemma 18 *There is a linear-time algorithm that given a partition $V_1, \dots, V_{k'}$ of V that is δ-close to the planted partition in the weak sense, outputs a partition $V'_1, \dots, V'_k$ that is (2δ)-close to the planted partition in the strong sense.*

Proof By the definition of the weak δ-closeness, every set V_i has size at most n. Therefore, $k' \geq k$. We choose k largest clusters among $V_1, \dots, V_{k'}$. Let $V'_1, \dots, V'_k$ be these clusters. We distribute, in an arbitrary way, all vertices from other clusters between $V'_1, \dots, V'_k$ so that each of the clusters V'_i contains exactly n vertices.

We now show that partition $V'_1, \dots, V'_k$ is (2δ)-close to the planted partition in the strong sense. We may assume without loss of generality that we chose clusters $V_1, \dots, V_k$ and that V'_i consists of V_i and some vertices from clusters V_j with $j > k$.

Let σ be the partial matching between clusters V^*_i and V_j (from the definition of the δ-closeness). We first let $\sigma'(i) = \sigma(i)$ if $\sigma(i)$ is defined and $\sigma(i) \leq k$. We get a partially defined permutation on $\{1, \dots, k\}$. Then we extend σ to a permutation defined everywhere in an arbitrary way. Write,

$$\Big|\bigcup_{j=\sigma'(i)} V^*_i \cap V'_j\Big| \geq \Big|\bigcup_{j=\sigma(i)\leq k} V^*_i \cap V_j\Big| = \Big|\bigcup_{j=\sigma(i)} V^*_i \cap V_j\Big| - \Big|\bigcup_{j=\sigma(i)\in\{k+1,\dots,k'\}} V^*_i \cap V_j\Big|$$
$$\geq (1-\delta)kn - \Big|\bigcup_{j=\sigma(i)\in\{k+1,\dots,k'\}} V_j\Big|.$$

Let

$$J_1 = \{j \in \{k+1, \dots, k'\} : j = \sigma(i)\}$$
$$J_2 = \{j \in \{1, \dots, k\} : j \neq \sigma(i) \text{ for every } i\}.$$

Since σ takes at most k values, $|J_1| \leq |J_2|$. Also, $|V_{j_1}| \leq |V_{j_2}|$ for every $j_1 \in J_1$ and $j_2 \in J_2$ by our choice of $V_1, \dots, V_k$. Therefore,

$$\Big|\bigcup_{j\in J_1} V_j\Big| \leq \Big|\bigcup_{j\in J_2} V_j\Big| \leq \Big|\bigcup_{V_j \text{ is not matched}} V_j\Big| \leq \delta nk.$$

We conclude that

$$\Big|\bigcup_{j=\sigma'(i)} V^*_i \cap V'_j\Big| \geq (1-2\delta)nk.$$

■

Now we are ready to prove Theorem 4.

Proof [Proof of Theorem 4] We solve the SDP relaxation. Consider the parameter α, which is defined by (4). From Theorem 7, we get that α satisfies bounds (4.2) with $s = 2$ and (4.7) with probabilities at least $1 - 2\exp(-2N)$ and $1 - 2\exp(-\eta m)$. Now we run the Recovery Algorithm and find a partition $(V_1, \dots, V_k)$. By Theorem 17, it is (72α)-close to the planted partition in the weak sense. Finally, using the algorithm from Lemma 18, we transform this partition to the desired partition $V'_1, \dots, V'_k$, which is (144α)-close to the planted partition. ■

6. Second Algorithm

In this section, we present our second algorithm and prove Theorem 19. Theorem 5 follows immediately from Theorem 19.

Theorem 19 *Suppose that there is a polynomial-time algorithm $\mathcal{A}$ that given an instance of SBM(n, k, $a/2$, $b/2$) with εm outliers finds a partition $V_1, \dots, V_k$ that is $1/(10k)$-close to the planted partition (in the strong sense) with probability at least $1-\tau$. Then there is a randomized polynomial-time algorithm that given an instance of SBM(n, k, a, b) with εm outliers finds a partition $U_1, \dots, U_k$ that satisfies the following property. For every $\delta_0 \geq ke^{-\frac{(a-b)^2}{100a}}$, the partition $U_1, \dots, U_k$ is δ-close to the planted partition (in the strong sense), where*

$$\delta = 4\delta_0 + \frac{80\varepsilon m}{(a-b)kn}\,,$$

with probability at least $1 - \tau - \exp(-\delta_0 kn/6)$.

Proof Recall that in the stochastic-block model with outliers we generate the set of edges E in two steps. First, we generate a random set of edges $E' = E_{sb}$. Then, the adversary adds and removes some edges from E', and we obtain the set of edges E.

Let us partition all edges in E' and E into two groups. To this end, independently color all edges of $E \cup E'$ in two colors 1 and 2 uniformly at random. Let E_1 and E_2 be the subsets of edges in E colored in 1 and 2, respectively; similarly, let E'_1 and E'_2 be the subsets of edges in E' colored in 1 and 2. Denote $E_i^{\Delta} = E_i \Delta E'_i$ for $i \in \{1, 2\}$. Note that (V, E_1) is an instance of SBM(n, k, $a/2$, $b/2$) with εm outliers.

Given the graph $G = (V, E)$, we generate sets of edges E_1 and E_2 (it is important that to do so, we do not have to know E'). We first use edges in E_1 to find a partition that is $1/(10k)$-close to the planted partition. To this end, we run algorithm $\mathcal{A}$ on (V, E_1) and obtain a partition $V_1, \dots, V_k$ of V.

Now we use edges from E_2 to find a partition that is δ-close to the planted partition. We do this in two steps. First, we define a partition $U_1^0, \dots, U_k^0$, which is close to the planted partition but not necessarily balanced – some sets U_i may contain more then n vertices. Then we transform $U_1^0, \dots, U_k^0$ to a balanced partition $U_1, \dots, U_k$.

Let us start with defining the partition $U_1^0, \dots, U_k^0$. For technical reasons (to ensure that certain events that we consider below are independent), it will be convenient to us to partition each set V_i into two sets V_i^L and V_i^R containing $n/2$ vertices each (we assume that n is even; otherwise we can take sets of sizes $(n-1)/2$ and $(n+1)/2$). Denote $n' = |V_i^L| = |V_i^R| = n/2$. Let $V^L = \bigcup_i V_i^L$ and $V^R = \bigcup_i V_i^R$.

For every vertex $u \in V^L$, we count the number of its neighbors w.r.t. edges in E_2 in each of the sets $V_1^R, \dots, V_k^R$. We find the set V_i^R that has most neighbors of u and add u to U_i^0 (we break ties arbitrarily). Similarly, for every vertex $u \in V^R$, we count the number of its neighbors w.r.t. edges in E_2 in each of the sets $V_1^L, \dots, V_k^L$, find the set V_i^L that has most neighbors of u, and add u to U_i^0. We obtain a partition $U_1^0, \dots, U_k^0$.

Now we make sure that all clusters have the same size. To this end, we redistribute vertices from clusters of size greater than n among other clusters so that each cluster has size n. Formally, we first let $U_i^1 = U_i^0$ if $|U_i^0| \leq n$, and let U_i^1 be an arbitrary subset of n vertices of U_i^0 if $|U_i^0| > n$.

Then we arbitrarily assign all remaining vertices (i.e., vertices from $\bigcup_i U_i^0 \setminus U_i^1$) among all clusters so that each cluster contains exactly n vertices. We obtain a partition $U_1, \ldots, U_k$.

Let us analyze this algorithm. We may assume without loss of generality that the matching between the partition $V_1, \ldots, V_k$ and the planted partition is given by the identity permutation. Then

$$\sum_{i=1}^{k} |V_i^* \cap V_i| \geq nk(1 - 1/(10k)) = nk - n/10.$$

In particular, for every cluster V_i, we have

$$\begin{aligned} &|V_i \cap V_i^*| \geq 9n/10, \\ &|V_i \cap V_j^*| \leq n/10 \qquad \text{for every } j \neq i. \end{aligned}$$

Also, for every set V_i^R (and similarly for every set V_i^L), we have

$$|V_i^R \cap V_i^*| \geq 9n/10 - n/2 = 4n'/5, \tag{6.1}$$

$$|V_i^R \cap V_j^*| \leq n/10 = n'/5 \qquad \text{for every } j \neq i. \tag{6.2}$$

Let us say that a vertex u is *corrupted* if it is incident on at least $T = (a-b)/20$ edges in E_2^Δ.

Claim 20 *The total number of corrupted edges is at most* $2\varepsilon m/T$.

Proof Each edge in E_2^Δ is incident to at most two corrupted vertices. The total number of edges in E_2^Δ is at most εm. Therefore, the number of corrupted vertices is at most $2\varepsilon m/T$. ■

Consider a vertex $u \in V_i^*$. Assume that it is not corrupted. We are going to show that $u \in U_i^0$ with probability at least $1 - ke^{-\frac{(a-b)^2}{100a}}$.

We assume without loss of generality that $u \in V^L$. Let random variable Z_j be the number of neighbors of u in V_j^R w.r.t. edges in E_2'. Consider the event $\mathcal{E}_u$ that $Z_i \geq (3a+2b)/20$ and $Z_j \leq (2a+3b)/20$ for every $j \neq i$. We will prove now that if $\mathcal{E}_u$ happens then $u \in U_i^0$. After that we will show that the probability that $\mathcal{E}_u$ does not happen is exponentially small.

Assume to the contrary that $\mathcal{E}_u$ happens but $u \in U_j^0$ for some $j \neq i$. Then u has at least as many neighbors in V_j^R as is in V_i^R. Let A_+ be the number of edges $e \in E_2 \setminus E_2'$ from u to vertices in V_j (edges added by the adversary); and A_- be the number of edges in $e \in E_2' \setminus E_2$ from u to V_i (edges removed by the adversary). Then $A_+ + A_- < T$ since u is not corrupted. Observe that there are at most $Z_j + A_+$ edges $e \in E_2$ from u to V_j^R; there are at least $Z_i - A_-$ edges $e \in E_2$ from u to V_i^R. Therefore, $Z_j + A_+ \geq Z_i - A_-$, and hence (using that event $\mathcal{E}_u$ happens)

$$T > A_+ + A_- \geq Z_i - Z_j \geq \frac{3a+2b}{20} - \frac{2a+3b}{20} = \frac{a-b}{20} = T,$$

we get a contradiction.

We use the Bernstein inequality to upper bound the probability that $\mathcal{E}_u$ does not happen. Note that for every j (including $j = i$), u is connected to every vertex in $V_j^R \cap V_i^*$ by an edge in E_2' with probability $a/(2n)$; u is connected to every vertex in $V_j^R \setminus V_i^*$ by an edge in E_2' with probability

$b/(2n)$. Using bound (6.1), we get that the expected number of neighbors of u in V_i^R is at least $(4a+b)/10$. That is, $\mathbb{E}[Z_i] \geq (4a+b)/10$. By the Bernstein inequality,

$$\mathbb{P}[Z_i < (3a+2b)/10] \leq e^{-\frac{(a-b)^2/100}{2((4a+b)/10+(a-b)/30)}} \leq e^{-\frac{(a-b)^2}{100a}}.$$

Similarly, using bound (6.2), we get that for every $j \neq i$, $\mathbb{E}[Z_j] \leq (a+4b)/5$. By the Bernstein inequality,

$$\mathbb{P}[Z_j > (2a+3b)/10] \leq e^{-\frac{(a-b)^2/100}{2((a+4b)/10+(a-b)/30)}} \leq e^{-\frac{(a-b)^2}{100a}}.$$

By the union bound, $\mathbb{P}[\mathcal{E}_u] \geq 1 - ke^{-\frac{(a-b)^2}{100a}}$.

We proved that for every $u \in V_i^*$, $\mathbb{P}[u \in U_i^0] \geq 1 - \delta_0$ (recall that $\delta_0 \geq ke^{-\frac{(a-b)^2}{100a}}$). Let B_L be the number of vertices in V_L such that $\mathcal{E}_u$ does not happen, and B_R be the number of vertices in V_R such that $\mathcal{E}_u$ does not happen.

Note that $\mathbb{E}[B_L] \leq \delta_0 kn'$. Also all events $\mathcal{E}_u$ with $u \in V_L$ are independent since each event $\mathcal{E}_u$ depends only on the subset of edges of E_2 that goes from u to V_R. Therefore, by the Chernoff bound

$$\mathbb{P}[B_L \geq 2\delta_0 kn'] < e^{-\delta_0 kn'/3} = e^{-\delta_0 kn/6}.$$

Similarly, $\mathbb{P}[B_R \geq 2\delta_0 kn'] < e^{-\delta_0 kn/6}$, and $\mathbb{P}[B_L + B_R \geq 2\delta_0 kn] < 2e^{-\delta_0 kn/6}$.

Assume now that $\mathbb{P}[B_L + B_R < 2\delta_0 kn]$. Then

$$\mathbb{E}\Big[\sum_{i=1}^{k} |V_i^* \cap U_i^0|\Big] \geq (1-2\delta_0)kn - 40\varepsilon m/(a-b) = (1-\delta/2)kn.$$

here, $40\varepsilon m/(a-b)$ is the upper bound on the number of corrupted vertices from Claim 20, and $\delta = 4\delta_0 + \frac{80\varepsilon m}{(a-b)kn}$ as in the statement of the theorem. Now,

$$\sum_{i=1}^{k} |V_i^* \cap U_i| \geq \sum_{i=1}^{k} |V_i^* \cap U_i^1| \geq \sum_{i=1}^{k} |V_i^* \cap U_i^0| - \sum_{i:|U_i^0|>n} (|U_i^0| - n) \geq (1-\delta/2)kn - (\delta/2)kn = (1-\delta)kn.$$

We proved that $U_1, \ldots, U_k$ is δ-close to the planted partition, when algorithm $\mathcal{A}$ succeeds and $B_L + B_R < 2\delta_0 kn$; that is, with probability at least $1 - \tau - \exp(-\delta_0 kn/6)$. ■

Now we present the proof of Theorem 5.

Proof [Proof of Theorem 5] Observe that under our assumption that

$$\frac{\sqrt{a+b(k-1)}}{a-b} + \frac{\varepsilon\,(a+b(k-1))}{a-b} \leq c/k,$$

our first algorithm finds a partition that is $1/(10k)$-close to the planted partition given an instance of SBM$(n, k, a/2, b/2)$. Hence, we can apply Theorem 19 and get a partition that is δ-close to the planted partition, as desired. ■

7. KL-divergence

Proof [Proof of Theorem 6] Theorem 6 almost immediately follows from Theorem 4 and Lemma 21.

Lemma 21 *Consider two distributions P, Q over the same sample space Ω. For every event $\mathcal{E} \subset \Omega$, we have*

$$Q(\mathcal{E}) \leq \max\left(\frac{2d_{KL}(Q,P)}{-\log P(\mathcal{E})+1}, e\sqrt{2P(\mathcal{E})}\right), \tag{7.1}$$

where $d_{KL}(Q,P)$ is the Kullback–Leibler divergence of P from Q.

Consider the worst adversary A for the algorithm from Theorem 4 — that is, the adversary for which the algorithm succeeds to recover a $(1-\delta)$ fraction of vertices with the smallest probability. The adversary takes the graph $G \sim \mathcal{G}$ and transforms it to $A(G)$. Without loss of generality we may assume that the adversary is deterministic. Let $\mathcal{E}$ be the set of graphs G for which our algorithm fails to recover δ fraction of vertices on the corrupted graph $A(G)$. By Theorem 4, the probability of $\mathcal{E}$ in the Stochastic Block Model distribution is at most $2\exp(-\eta m)$. Thus, by Lemma 21, the probability of $\mathcal{E}$ in the distribution of $\mathcal{G}$ is bounded as

$$\delta \leq \max\left(\frac{2\lambda m}{\eta m}, 2e^{\frac{-\eta m}{2}+1}\right) = \max\left(\frac{2\lambda}{\eta}, 2e^{\frac{-\eta m}{2}+1}\right).$$

■

We now prove an auxiliary Lemma 22 and then Lemma 21.

Lemma 22 *Consider two distributions P, Q over the same sample space Ω. Suppose that Ω is the union of disjoint events $\mathcal{E}_i$. Then*

$$d_{KL}(Q,P) \geq \sum_i Q(\mathcal{E}_i)\log\frac{Q(\mathcal{E}_i)}{P(\mathcal{E}_i)}.$$

Proof By the definition, KL divergence equals

$$d_{\mathrm{KL}}(Q,P) = \sum_{\sigma\in\Omega} Q(\sigma)\log\frac{Q(\sigma)}{P(\sigma)} = \sum_i \sum_{\sigma\in\mathcal{E}_i} Q(\sigma)\log\frac{Q(\sigma)}{P(\sigma)}.$$

We lower bound each of the terms on the right hand side using the log-sum inequality (which follows from the convexity of the function $x \mapsto x\log x$ and Jensen's inequality).

Claim 23 (Log-sum inequality see e.g. Csiszar and Körner (2011)) *Let $q_1, \ldots, q_T$ and $p_1, \ldots, p_T$ be nonnegative numbers. Then,*

$$\sum_i q_i \log\frac{q_i}{p_i} \geq \left(\sum_i q_i\right)\log\left(\frac{\sum_i q_i}{\sum_i p_i}\right).$$

We get

$$d_{\mathrm{KL}}(Q,P) \geq \sum_i \left(\sum_{\sigma\in\mathcal{E}_i} Q(\sigma)\right)\log\frac{\sum_{\sigma\in\mathcal{E}_i} Q(\sigma_i)}{\sum_{\sigma\in\mathcal{E}_i} P(\sigma_i)} = \sum_i Q(\mathcal{E}_i)\log\frac{Q(\mathcal{E}_i)}{P(\mathcal{E}_i)}.$$

■

Proof [Proof of Lemma 21] We apply Lemma 22 to the events $\mathcal{E}$ and $\bar{\mathcal{E}} = \Omega \setminus \mathcal{E}$:

$$d_{\mathrm{KL}}\left(Q, P\right) \geq Q(\mathcal{E}) \log \frac{Q(\mathcal{E})}{P(\mathcal{E})} + Q(\bar{\mathcal{E}}) \log \frac{Q(\bar{\mathcal{E}})}{P(\bar{\mathcal{E}})}. \tag{7.2}$$

We bound the second term on the right hand side using the inequality $x \log x \geq (x-1) \log e$ for $x \geq 0$:

$$Q(\bar{\mathcal{E}}) \log \frac{Q(\bar{\mathcal{E}})}{P(\bar{\mathcal{E}})} = P(\bar{\mathcal{E}}) \times \left[\frac{Q(\bar{\mathcal{E}})}{P(\bar{\mathcal{E}})} \log \frac{Q(\bar{\mathcal{E}})}{P(\bar{\mathcal{E}})}\right] \geq P(\bar{\mathcal{E}}) \times \left[\frac{Q(\bar{\mathcal{E}})}{P(\bar{\mathcal{E}})} - 1\right] \log e =$$
$$= (Q(\bar{\mathcal{E}}) - P(\bar{\mathcal{E}})) \log e = (P(\mathcal{E}) - Q(\mathcal{E})) \log e \geq -Q(\mathcal{E}) \log e.$$

We have

$$d_{\mathrm{KL}}\left(Q, P\right) \geq Q(\mathcal{E}) \log \frac{Q(\mathcal{E})}{P(\mathcal{E})} - Q(\mathcal{E}) \log e = Q(\mathcal{E}) \log \frac{Q(\mathcal{E})}{e \cdot P(\mathcal{E})}.$$

Thus, either $Q(\mathcal{E}) \leq e\sqrt{2P(\mathcal{E})}$, or $Q(\mathcal{E})/(eP(\mathcal{E})) \geq \sqrt{2/P(\mathcal{E})}$, and, consequently,

$$Q(\mathcal{E}) \leq \frac{2d_{\mathrm{KL}}\left(Q, P\right)}{-\log(P(\mathcal{E})) + 1}.$$

■

Acknowledgments

We would like to thank Elchanan Mossel for some useful discussions.

References

Emmanuel Abbe and Colin Sandon. Community detection in general stochastic block models: fundamental limits and efficient recovery algorithms. In *Proceedings of the Symposium on Foundations of Computer Science*, 2015.

Emmanuel Abbe, Afonso S. Bandeira, and Georgina Hall. Exact recovery in the stochastic block model. *CoRR*, abs/1405.3267, 2014. URL http://arxiv.org/abs/1405.3267.

Dimitris Achlioptas and Frank McSherry. On spectral learning of mixtures of distributions. In *Learning Theory*, volume 3559 of *Lecture Notes in Computer Science*, pages 458–469. 2005.

Naman Agarwal, Afonso S. Bandeira, Konstantinos Koiliaris, and Alexandra Kolla. Multisection in the stochastic block model using semidefinite programming. *CoRR*, abs/1507.02323, 2015. URL http://arxiv.org/abs/1507.02323.

Brendan P.W. Ames. Guaranteed clustering and biclustering via semidefinite programming. *Mathematical Programming*, 147(1-2):429–465, 2014.

Pranjal Awasthi and Or Sheffet. Improved spectral-norm bounds for clustering. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 7408 of *Lecture Notes in Computer Science*, pages 37–49, 2012. ISBN 978-3-642-32511-3.

Nikhil Bansal, Uriel Feige, Robert Krauthgamer, Konstantin Makarychev, Viswanath Nagarajan, Joseph Seffi, and Roy Schwartz. Min-max graph partitioning and small set expansion. *SIAM Journal on Computing*, 43(2):872–904, 2014.

Avrim Blum and Joel Spencer. Coloring random and semi-random k-colorable graphs. *J. Algorithms*, 19:204–234, September 1995.

Ravi B. Boppana. Eigenvalues and graph bisection: An average-case analysis. In *Symposium on Foundations of Computer Science*, pages 280–285, 1987.

M. Braverman, K. Makarychev, Y. Makarychev, and A. Naor. The Grothendieck constant is strictly smaller than Krivine's bound. In *Foundations of Computer Science*, pages 453–462, 2011.

S. Charles Brubaker. Robust PCA and clustering in noisy mixtures. In *Proceedings of the Symposium on Discrete Algorithms*, pages 1078–1087, 2009.

Thang Nguyen Bui, F. Thomson Leighton, Soma Chaudhuri, and Michael Sipser. Graph bisection algorithms with good average case behavior. *Combinatorica*, 7:171–191, June 1987.

T. Tony Cai and Xiaodong Li. Robust and computationally feasible community detection in the presence of arbitrary outlier nodes. *Ann. Statist.*, 43(3):1027–1059, 06 2015.

Moses Charikar, Samir Khuller, David M. Mount, and Giri Narasimhan. Algorithms for facility location problems with outliers. In *Proceedings of the Symposium on Discrete Algorithms*, SODA '01, pages 642–651, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics. ISBN 0-89871-490-7. URL http://dl.acm.org/citation.cfm?id=365411.365555.

Yudong Chen, Sujay Sanghavi, and Huan Xu. Clustering sparse graphs. In *Advances in Neural Information Processing Systems 25*, pages 2204–2212. Curran Associates, Inc., 2012.

Peter Chin, Anup Rao, and Van Vu. Stochastic block model and community detection in sparse graphs: A spectral algorithm with optimal rate of recovery. In *Proceedings of Conference on Learning Theory*, pages 391–423, 2015.

Amin Coja-Oghlan. A spectral heuristic for bisecting random graphs. *Random Structures & Algorithms*, 29(3):351–398, 2006.

Amin Coja-Oghlan. Graph partitioning via adaptive spectral techniques. *Comb. Probab. Comput.*, 19(2):227–284, March 2010.

Anne Condon and Richard Karp. Algorithms for graph partitioning on the planted partition model. In *Randomization, Approximation, and Combinatorial Optimization. Algorithms and Techniques*, volume 1671 of *Lecture Notes in Computer Science*, pages 221–232. Springer Berlin / Heidelberg, 1999.

Imre Csiszar and János Körner. *Information theory: coding theorems for discrete memoryless systems*. Cambridge University Press, 2011.

Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Phys. Rev. E*, 84:066106, Dec 2011.

Tassos Dimitriou and Russell Impagliazzo. Go with the winners for graph bisection. In *Proceedings of the Symposium on Discrete Algorithms*, pages 510–520, 1998.

M. E. Dyer and A. M. Frieze. Fast solution of some random NP-hard problems. In *27th Annual Symposium on Foundations of Computer Science*, pages 331–336, 1986.

Uriel Feige and Joe Kilian. Heuristics for finding large independent sets, with applications to coloring semi-random graphs. In *Proceedings of Symposium on Foundations of Computer Science*, pages 674–683, 1998.

Santo Fortunato. Community detection in graphs. *Physics Reports*, 486:75–174, 2010.

A. Grothendieck. Résumé de la théorie métrique des produits tensoriels topologiques. *Bol. Soc. Mat. Sao Paulo*, 8:1–79, 1953.

Olivier Guédon and Roman Vershynin. Community detection in sparse networks via Grothendieck's inequality. *CoRR*, abs/1411.4686, 2014.

Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109 – 137, 1983.

Mark Jerrum and Gregory Sorkin. Simulated annealing for graph bisection. In *Proceedings of the Symposium on Foundations of Computer Science*, pages 94–103, 1993.

Ravindran Kannan, Hadi Salmasian, and Santosh Vempala. The spectral method for general mixture models. In *Learning Theory*, volume 3559 of *Lecture Notes in Computer Science*, pages 444–457. Springer Berlin Heidelberg, 2005.

Robert Krauthgamer, Joseph Seffi Naor, and Roy Schwartz. Partitioning graphs into balanced components. In *Proceedings of the Symposium on Discrete Algorithms*, pages 942–949, 2009.

Jean-Louis Krivine. Sur la constante de Grothendieck. *CR Acad. Sci. Paris Ser. AB*, 284(8):A445–A446, 1977.

Amit Kumar and Ravindran Kannan. Clustering with spectral norm and the k-means algorithm. In *Proceedings of Symposium on Foundations of Computer Science*, pages 299–308, 2010.

Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Approximation algorithms for semi-random partitioning problems. In *Proceedings of Symposium on Theory of Computing*, pages 367–384, 2012.

Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Constant factor approximation for balanced cut in the random PIE model. In *Proceedings of Symposium on Theory of Computing*, 2014.

Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Correlation clustering with noisy partial information. In *Proceedings of the Conference on Learning Theory (COLT)*, 2015.

Laurent Massoulié. Community detection thresholds and the weak Ramanujan property. In *Symposium on Theory of Computing*, pages 694–703, 2014.

Colin McDiarmid. Concentration. In *Probabilistic methods for algorithmic discrete mathematics*, pages 195–248. Springer, 1998.

Frank McSherry. Spectral partitioning of random graphs. In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, pages 529–537, 2001.

Ankur Moitra, William Perry, and Alexander S. Wein. How robust are reconstruction thresholds for community detection. *CoRR*, abs/1511.01473, 2015.

Andrea Montanari and Subhabrata Sen. Semidefinite programs on sparse random graphs. *CoRR*, abs/1504.05910, 2015. URL http://arxiv.org/abs/1504.05910.

Elchanan Mossel, Joe Neeman, and Allan Sly. Stochastic block models and reconstruction. *arXiv preprint arXiv:1202.1499*, 2012.

Elchanan Mossel, Joe Neeman, and Allan Sly. A proof of the block model threshold conjecture. *CoRR*, abs/1311.4115, 2013. URL http://arxiv.org/abs/1311.4115.

Elchanan Mossel, Joe Neeman, and Allan Sly. Belief propagation, robust reconstruction and optimal recovery of block models. In *Proceedings of the Conference on Learning Theory*, pages 356–370, 2014.

Elchanan Mossel, Joe Neeman, and Allan Sly. Consistency thresholds for the planted bisection model. In *Proceedings of the Symposium on Theory of Computing*, pages 69–75, 2015.

William Perry and Alexander S. Wein. A semidefinite program for unbalanced multisection in the stochastic block model. *CoRR*, abs/1507.05605, 2015. URL http://arxiv.org/abs/1507.05605.

Van Vu. A simple svd algorithm for finding hidden partitions. *arXiv preprint arXiv:1404.3918*, 2014.

Harrison C. White, Scott A. Boorman, and Ronald L. Breiger. Social structure from multiple networks. i. blockmodels of roles and positions. *American Journal of Sociology*, 81(4):730–780, 1976.

Y. Wu, J. Xu, and B. Hajek. Achieving exact cluster recovery threshold via semidefinite programming under the stochastic block model. In *2015 49th Asilomar Conference on Signals, Systems and Computers*, pages 1070–1074, Nov 2015. doi: 10.1109/ACSSC.2015.7421303.

Anderson Y. Zhang and Harrison H. Zhou. Minimax rates of community detection in stochastic block models. *CoRR*, abs/1507.05313, 2015. URL http://arxiv.org/abs/1507.05313.

Appendix A. Concentration Inequalities

A.1. Bernstein Inequality

We will use the following standard inequality known as the Bernstein inequality or the Hoeffding inequality (see e.g., Theorem 2.7. in McDiarmid (1998)).

Fact 24 *Let $X_1, \dots, X_n$ be independent random variables with $X_i - \mathbb{E}[X_i] \le B$ for all i. Then*

$$\mathbb{P}\left[\sum_i X_i - \mathbb{E}\sum_i X_i > t\right] \le \exp\left(-\frac{t^2/2}{\sigma^2 + Bt/3}\right), \tag{A.1}$$

where $\sigma^2 = \sum_i \mathbb{V}\mathrm{ar}[X_i]$. For Bernoulli random variables taking values 0 and 1, $\sigma^2 \le \sum_i \mathbb{E}\, X_i$.

A.2. Size of the Planted Cut

In this section, we upper bound the size of the planted cut. For convenience, we give the same probability estimate as in Theorem 9.

Lemma 25 *For a random graph G_{sb} from the Stochastic Block model SBM(n, k, a, b), we have*

$$\mathbb{P}(\mathrm{planted}(G_{sb}) \le b(k-1)N/2 + 2\sqrt{a + b(k-1)}Ns) \ge 1 - e^{-\frac{9s^2}{4+8s/\sqrt{a+b(k-1)}}N}.$$

Proof The expected size of the planted cut is $b(k-1)N/2$. Thus, by the Bernstein inequality, we have

$$\mathbb{P}(\mathrm{planted}(G_{sb}) \le b(k-1)N/2 + t) \le e^{-\frac{t^2}{b(k-1)N + 2t/3}}.$$

For $t = 2\sqrt{a + b(k-1)}Ns$, we get

$$\mathbb{P}(\mathrm{planted}(G_{sb}) < b(k-1)N + t) \le e^{-\frac{s^2 N}{1/4 + s/(3\sqrt{a+b(k-1)})}} < e^{-\frac{9s^2}{4+8s/\sqrt{a+b(k-1)}}N}.$$

■

A.3. Proof of Theorem 9

In this section, we prove Theorem 9, which is an analog of Lemma 4.1 in Guédon and Vershynin (2014). The proof closely follows their proof. In the proof, we will use the Grothendieck inequality (see Grothendieck (1953); Krivine (1977); Braverman et al. (2011)).

Theorem 26 (Grothendieck inequality) *For every $n \times n$ matrix M, the following inequality holds*

$$\max_{\|U_i\|, \|V_j\| = 1} \left| \sum_{i,j=1}^n M_{ij} \langle U_i, V_j \rangle \right| \le K_G \cdot \max_{x, y \in \{-1,1\}^n} \sum_{i,j=1}^n M_{ij} x_i y_j, \tag{A.2}$$

where $K_G \le 1.783$ is the Grothendieck constant. The first maximum is over all unit vectors $U_1, \dots, U_n$ and $V_1, \dots, V_n$.

Proof [Proof of Theorem 9] Let L be the Laplacian of the graph G_{sb} and $\Delta L = L - \mathbb{E}[L]$. For any feasible SDP solution $\{\tilde{u}\}$, we have

$$\sum_{u<v} \Delta a_{uv} \|\tilde{u} - \tilde{v}\|^2 = \sum_{u,v} \Delta L_{uv} \langle \tilde{u}, \tilde{v} \rangle.$$

We upper bound the right hand side using the Grothendieck inequality (with $U_i = V_i = \tilde{u}$, where u is the i-th vertex in the graph):

$$\begin{aligned} \max_{\{\tilde{u}\}} \Big| \sum_{u,v} \Delta L_{uv} \langle \tilde{u}, \tilde{v} \rangle \Big| &\leq K_G \max_{x,y \in \{-1,1\}^n} \sum_{u,v} \Delta L_{uv}\, x_u y_v \\ &= K_G \max_{x,y \in \{-1,1\}^n} \sum_{u<v} \Delta a_{uv} (x_u - x_v)(y_u - y_v). \end{aligned} \tag{A.3}$$

Note that each Δa_{uv} is a Bernoulli random variable taking values $-\mathbb{E}[a_{uv}]$ and $1 - \mathbb{E}[a_{uv}]$ with probabilities $1 - \mathbb{E}[a_{uv}]$ and $\mathbb{E}[a_{uv}]$, respectively. All values $(x_u - x_v)(y_u - y_v)$ lie in the set $\{-4, 0, 4\}$. By the Bernstein inequality, for fixed $x, y \in \{-1, 1\}^n$, we have

$$\mathbb{P}\Big(\sum_{u<v} \Delta a_{uv} (x_u - x_v)(y_u - y_v) \geq t \Big) \leq e^{-\frac{t^2}{2\sigma^2(x,y) + 8t/3}},$$

where

$$\sigma^2(x,y) = \sum_{u<v} \mathbb{V}\mathrm{ar}\left[\Delta a_{uv} (x_u - x_v)(y_u - y_v) \right] = \sum_{u<v} \mathbb{V}\mathrm{ar}[\Delta a_{uv}] (x_u - x_v)^2 (y_u - y_v)^2.$$

Since the set of edges E_{sb} comes from the stochastic block model, we have $\mathbb{E}\, a_{uv} = a/n$ if $(u,v) \in (V \times V)_{in}$, and $\mathbb{E}\, a_{uv} = b/n$ if $(u,v) \in (V \times V)_{out}$. Note that $\mathbb{V}\mathrm{ar}[\Delta a_{uv}] = \mathbb{E}[a_{uv}]\,(1 - \mathbb{E}[a_{uv}]) < \mathbb{E}[a_{uv}]$. Thus,

$$\begin{aligned} \sigma^2(x,y) &\leq \frac{a}{n} \sum_{\substack{(u,v) \in (V \times V)_{in} \\ u<v}} (x_u - x_v)^2 (y_u - y_v)^2 + \frac{b}{n} \sum_{\substack{(u,v) \in (V \times V)_{out} \\ u<v}} (x_u - x_v)^2 (y_u - y_v)^2 \\ &\leq \frac{4a}{n} \sum_{\substack{(u,v) \in (V \times V)_{in} \\ u<v}} (x_u - x_v)^2 + \frac{4b}{n} \sum_{\substack{(u,v) \in (V \times V)_{out} \\ u<v}} (x_u - x_v)^2 \\ &= \frac{4(a-b)}{n} \sum_{\substack{(u,v) \in (V \times V)_{in} \\ u<v}} (x_u - x_v)^2 + \frac{4b}{n} \sum_{\substack{(u,v) \in V \times V \\ u<v}} (x_u - x_v)^2. \end{aligned}$$

For any set $S \subset V$,

$$\sum_{\substack{(u,v) \in S \times S \\ u<v}} (x_u - x_v)^2 = 4\, |\{u \in S : x_u = -1\}| \cdot |\{v \in S : x_v = 1\}| \leq |S|^2.$$

Hence,

$$\sigma^2(x,y) \leq \frac{4(a-b)}{n} \times k \times n^2 + \frac{4b}{n} \times (nk)^2 = 4aN + 4b(k-1)N.$$

Consequently,

$$\mathbb{P}\Big(\sum_{u<v}\Delta a_{uv}(x_u-x_v)(y_u-y_v)\ge t\Big)\le e^{-\frac{t^2}{8(a+b(k-1))N+8t/3}}.$$

Using the union bound over all $x,y\in\{-1,1\}^V$, we get

$$\mathbb{P}\Big(\max_{x,y\in\{-1,1\}^n}\sum_{u<v}\Delta a_{uv}(x_u-x_v)(y_u-y_v)\ge t\Big)\le 2^{2N}e^{-\frac{t^2}{8(a+b(k-1))N+8t/3}}.$$

By (A.3),

$$\mathbb{P}\Big(\max_{\{\tilde u\}}\Big|\sum_{u<v}\Delta a_{uv}\|\tilde u-\tilde v\|^2\Big|\ge K_G\,t\Big)\le 2^{2N}e^{-\frac{t^2}{8(a+b(k-1))N+8t/3}}=e^{-\frac{t^2}{8(a+b(k-1))N+8t/3}+2N\ln 2}.$$

Let $t=6\sqrt{a+b(k-1)}Ns$. Then,

$$\begin{aligned}\frac{t^2}{8(a+b(k-1))N+8t/3}-2N\ln 2 &= \Big(\frac{9s^2}{2+4s/\sqrt{a+b(k-1)}}-2\ln 2\Big)N\\ &\ge \frac{9s^2}{4+8s/\sqrt{a+b(k-1)}}\,N.\end{aligned}$$

The last inequality holds for $s\ge 1$ and $a+b(k-1)\ge 11$. Therefore,

$$\mathbb{P}\Big(\max_{\tilde u}\Big|\sum_{u<v}\Delta a_{uv}\|\tilde u-\tilde v\|^2\Big|\ge 6K_G\sqrt{a+b(k-1)}Ns\Big)\le e^{-\frac{9s^2}{4+8s/\sqrt{a+b(k-1)}}N}.$$

■

Appendix B. Lower Bounds

In this section we give lower bounds on the partial recovery in the model with two communities. We show that it is not possible to recover a δ fraction of all vertices in the pure Stochastic Block Model if

$$(a-b)<C\sqrt{(a+b)\ln 1/\delta}, \tag{B.1}$$

for some constant C, and it is not possible to recover a δ fraction of all vertices in the Stochastic Block Model with Outliers (where the adversary is allowed to add at most $\varepsilon(a+b)n$ edges) if

$$(a-b)<C\varepsilon\delta^{-1}(a+b). \tag{B.2}$$

We note that very recently Zhang and Zhou (2015) showed a lower bound with a dependence similar to (B.1). For simplicity of exposition we slightly alter the Stochastic Block Model. We consider graphs with parallel edges. The number of edges between two vertices u and v in the new model is not a Bernoulli random variable with parameter a/n or b/n as in the standard Stochastic Block Model, but a Poisson random variable with parameter a/n or b/n. Note that recovering

partitions in the Poisson model with very slightly modified parameters $a' = n\ln(1 - a/n)$ and $b' = n\ln(1 - b/n)$, is not harder than in the Bernoulli model, since the algorithm may simply replace parallel edges with single edges and obtain a graph from the standard Stochastic Block Model.

Before proceeding to the formal proofs, we informally discuss why these bounds hold. Consider two vertices u and v lying in the opposite clusters. Suppose we give the algorithm not only the graph G, but also the correct clustering of all vertices but u and v. The algorithm needs now to decide where to put u and v. It turns out that the only *useful* information the algorithm has about u and v are the four numbers – the number of neighbors u and v have in the left and right clusters. These numbers are distributed according to the Poisson distribution with parameters a and b. So the algorithm is really given four numbers: two numbers X_1, Y_1 for vertex u and two numbers Y_2, X_2 for vertex v. The algorithm needs to decide whether

(a) X_1 and X_2 have the Poisson distribution with parameter a, and Y_1 and Y_2 have the Poisson distribution with parameter b; or

(b) X_1 and X_2 have the Poisson distribution with parameter b, and Y_1 and Y_2 have the Poisson distribution with parameter a.

We show in Corollary 36 that no test distinguishes (a) from (b) with error probability less than δ given by the bound (B.1). This implies (B.1).

To prove the bound (B.2), we first specify what the adversary does in the model with outlier edges (noise). It picks δn fraction of all vertices on the left side and on the right side. For each chosen vertex, it adds approximately $(a-b)$ extra edges going to the opposite side. After that every chosen vertex has the same distribution of edges going to the opposite cluster as to its own cluster. Hence, the chosen vertices on the left side and chosen vertices on the right side are statistically indistinguishable. To add $(a-b)$ extra edges to every chosen vertex, the adversary needs $2(a-b)\delta n$ edges, but he has a budget of $\Theta(\varepsilon(a+b)n)$ edges. This gives the bound (B.2).

In the rest of the section, we use the ideas outlined above to prove the following theorem. In the proof, we couple the distribution of the random variables $(X_1, Y_1), (Y_2, X_2)$ with the distribution of graphs in the Stochastic Block Model.

Theorem 27 *It is statistically impossible to recover more than δ fraction of all vertices if the bound B.1 holds in the Stochastic Block Model, and if the bound B.2 holds in the Stochastic Block Model with Outliers, where the adversary can add at most $O(\varepsilon(a+b)n)$ edges. The constant C is a universal constant.*

B.1. Adversary in the SBM with Outliers

We first describe the adversary for generating graphs in the SBM with Outlier edges. The adversary fixes two sets $L' \subset L$ and $R' \subset R$ in the left and right clusters of size ρn each for $\rho = \Theta(\varepsilon(a+b)/(a-b))$. Let $L'' = L \setminus L'$ and $R'' = R \setminus R'$. Then it generates a graph according to the pure Stochastic Block Model. The adversary counts the number of edges going from L' to R'', and the number of edges going from R' to L''. Denote these numbers by $Z_{L'}$ and $Z_{R'}$ respectively. Then, the adversary independently computes two numbers $\kappa_{L'} = \hat{\kappa}(Z_{L'})$ and $\kappa_{R'} = \hat{\kappa}(Z_{R'})$ using a random function $\hat{\kappa}$ we describe in a moment. He adds $\kappa_{L'}$ edges between L' and R'' and $\kappa_{R'}$ edges between

R' and L''. He adds the edges one by one every time adding one edge between a random vertex in L' and a random vertex in R'' or between a random vertex in R' and a random vertex in L''.

Denote $M = \rho(1-\rho)n$. In Corollary 39, we show that there exists a function $\hat{\kappa}$ upper bounded by $(a-b)M$ such that the total variation distance between P_1 and $\hat{\kappa}(P_2)$ is at most $1/2$, where P_1 and P_2 are Poisson random variables with parameters aM and bM. The adversary uses this function $\hat{\kappa}$. Note that he adds at most

$$4(a-b)M = 4(a-b)\rho(1-\rho)n \leq 4(a-b)\rho n = \Theta(\varepsilon(a+b)n),$$

edges.

B.2. Restricted Partitioning

Let us partition the sets L and R into two sets each: $L = L' \cup L''$ and $R = R' \cup R''$. We partition the sets before we generate the graph from the Stochastic Block Model, and thus the partitioning does not depend on the edges present in the graph. Consider the following classification task: the classifier gets the graph G generated according to the Stochastic Graph Model (with or without the adversary) and the sets L', R', L'' and R'' (which were chosen before the graph was generated). We specify that $L'' \subset L$ and $R'' \subset R$. However, we swap the order of L' and R' with probability $1/2$. Thus the classifier does not know whether $L' \subset L$ or $L' \subset R$ and whether $R' \subset L$ or $R' \subset R$. Its goal is to guess whether $L' \subset L$ or $L' \subset R$ and, consequently, whether $R' \subset L$ or $R' \subset R$. We call this classifier a restricted classifier.

Lemma 28 (Restricted Classifier for pure Stochastic Block Model) *If there exists a procedure that recovers partitions in the pure Stochastic Block Model with accuracy at least $1-\delta$, then there exists a restricted classifier (as above) for sets $L' = \{u\}$, $R' = \{v\}$, $L'' = L \setminus \{u\}$ and $R'' = \setminus\{v\}$ that errs with probability at most $2\delta + 1/n$.*

Proof The classifier works as follows. It executes the recovery procedure for the input graph $G = (V, E)$ and gets two sets S^* and T^*. It picks at random $w' \in \{u, v\}$ and $w'' \in L''$. Now if w' and w'' lie in the same set S^* or T^*, then the algorithm returns "$w' \in L$", otherwise it returns "$w' \in R$". What is the error probability of this classifier?

Since the distribution of graphs in the Stochastic Block Model is invariant under permutation of vertices in L and in R, the error probability will not change if we alter the process as follows: the classifier first runs the recovery procedure, then we pick two random vertices $u \in L$ and $v \in R$ and give these vertices to the classifier. Note that the classifier does not need u and v to run the recovery procedure. Let us compute the error probability. Suppose that the recovery procedure misclassified δ^* fraction of all vertices, and say S^* corresponds to L i.e. $|S^* \cap L| = (1-\delta^*)n$. If the algorithm picks $w' = u \in L$, then the probability that $w', w'' \in S^*$ equals $(1-\delta^*)((1-\delta^*)n - 1)/n \geq 1 - 2\delta^* - 1/n$. Similarly, if $w' = v \in R$, then the probability that $w' \in T^*$ and $w'' \in S^*$ equals $(1-\delta^*)^2 \geq 1 - 2\delta^*$.

Since the expected value of δ^* is at most δ we get the desired result. ■

We now prove a similar lemma for Stochastic Block Model with Outlier edges.

Lemma 29 (Restricted Classifier for Stochastic Block Model with Outliers) *If there exists a procedure that recovers partitions in the Stochastic Block Model with Outlier Edges with accuracy at*

least $1-\delta$*, then there exists a restricted classifier for sets* L'*,* R'*,* $L'' = L \setminus L'$ *and* $R'' = \setminus R'$ *with* $|L'| = |R'| < n/2$ *that errs with probability at most* $\delta n/|L'|$*.*

Proof As before, the classifier executes the recovery procedure for the input graph $G = (V, E)$ and gets two sets S^* and T^*. Then, the classifier picks sets $W' \in \{L', R'\}$ and $W'' = \{L'', R''\}$ at random. It also picks random vertices $w' \in W'$ and $w'' \in W''$. If w' and w'' lie in the same set S^* or T^*, the classifier returns "W' and W'' are on the same side of the cut"; otherwise, it returns "W' and W'' are on different sides of the cut". Note that the classifier knows whether $W'' = L''$ or $W'' = R''$, and hence whether W'' lies on the left or right side of the cut.

Let δ^* be the fraction of misclassified vertices. Further, let δ' be the fraction of misclassified vertices in $L' \cup R'$; and δ'' be the fraction of misclassified vertices in $L'' \cup R''$. Note that $\delta^* = \big(\delta'(|L'| + |R'|) + \delta''(|L''| + |R''|)\big)/(2n)$. The error probability of the classifier given the partition S^* and T^* is at most

$$1 - (1-\delta')(1-\delta'') \leq \delta_1 + \delta_2 \leq \frac{2\delta^* n}{|L'| + |R'|} = \frac{\delta^* n}{|L'|}.$$

The error probability over random choices of the graph is at most $\mathbb{E}[\delta^* n/|L'|] = \delta n/|L'|$. ■

In the next subsection, we argue that, in a way, the only useful information the restricted classifier can use about the graph given the sets L', R', L'' and R'' are the number of edges between sets L', L'', R' and R''.

B.3. Tests for Pairs of Distributions

Let D_1 and D_2 be two distributions; and let $D_{Left} = D_1 \times D_2$ and $D_{Right} = D_2 \times D_1$ be the product distributions – distributions of pairs (X, Y) and (Y, X), where X and Y are independent random variables distributed as D_1 and D_2 respectively. In this section, we consider tests that given two independent pairs of random variables (X_1, Y_1) and (Y_2, X_2) distributed according to D_{Left} and D_{Right} needs to decide which pair is drawn from D_{Left} and which from D_{Right}. The test gets the pairs as an unordered set $\{(X_1, Y_1), (Y_2, X_2)\}$. We show that the restricted classifier is essentially a test for distributions D_1 and D_2, where D_1 is the distribution of the total number of edges between L' and L''; D_2 is the distribution of the number of edges between R' and R''.

Lemma 30 *Consider the Block Stochastic Model with sets* L'*,* R'*,* L''*,* R'' *as in Lemma 28, or the Stochastic Model with Outlier edges, with sets* L'*,* R'*,* L''*,* R'' *as in Lemma 29. When we have outlier edges (noise), we assume that the adversary behaves as described in Section B.1 and the sets* L' *and* R' *he chooses are the same sets as above. Let* D_1 *be the distribution of the number of edges between* L' *and* L''*, and* D_2 *be the distribution of the number of edges between* L' *and* R''*. (Note, that the number of edges between* R' *and* R'' *is also distributed as* D_1*; the number of edges between* R' *and* L'' *is distributed as* D_2*.) Then, if there exists a restricted classifier (see the previous section) with error probability at most* δ*, then there exists a test that decides whether*

- $(X_1, Y_1) \sim D_1 \times D_2$ *and* $(Y_1, X_1) \sim D_2 \times D_1$*; or*
- $(X_1, Y_1) \sim D_2 \times D_1$ *and* $(Y_1, X_1) \sim D_1 \times D_2$

with error probability at most δ*.*

Proof Suppose we are given a restricted classifier with error probability at most δ. We construct a test for pairs $D_1 \times D_2$ and $D_2 \times D_1$. The test procedure receives two pairs (X_1, Y_1) and (Y_2, X_2). Then it generates a graph from the model (the pure SBM, or the one with outlier edges) as follows. It creates four sets of vertices A, B, L'' and R''. It adds edges to the subgraphs on $A \cup B$ and $L'' \cup R''$ as in the Stochastic Block Model with planted cuts (A, B) and (L'', R'') respectively. Then, it adds X_1, Y_1, X_2, Y_2 edges between A and L'', A and R'', B and R'', B and L'' respectively. These edges are added at random one by one: say, to add an edge between A and L'', the test procedure picks a random vertex in A and a random vertex in L'' and connects these vertices with an edge. Once the graph is generated, the procedure executes the restricted classifier. If the classifier tells that A and L'' are on the same side of the cut, the test returns that $X_1, X_2 \sim D_1$ and $Y_1, Y_2 \sim D_2$; otherwise, $X_1, X_2 \sim D_2$ and $Y_1, Y_2 \sim D_1$.

We now analyze the tester. We claim that the graph obtained by the procedure above is distributed according to the model (the pure SBM, or the one with outlier edges), and the planted cut is $(A \cup L'', B \cup R'')$ if $X_1, X_2 \sim D_1$ and $Y_1, Y_2 \sim D_2$; the planted cut is $(B \cup L'', A \cup R)$ if $X_1, X_2 \sim D_2$ and $Y_1, Y_2 \sim D_1$. For the proof, assume without loss of generality that $X_1, X_2 \sim D_1$ and $Y_1, Y_2 \sim D_2$.

Let N_{uv} be the number of edges between vertices u and v. In the pure Stochastic Block Model, we need to verify that random variables N_{uv} are independent; and N_{uv} has the Poisson distribution with parameter a/n for $(u, v) \in A \times L''$ and $(u, v) \in B \times R''$; N_{uv} has the Poisson distribution with parameter b/n for $(u, v) \in A \times R''$ and $(u, v) \in B \times L''$. This immediately follows from the following Poisson Thinning Property, since X_1, X_2, Y_1 and Y_2 have Poisson distributions with parameters $(a/n)|A| \cdot |L'|$, $(a/n)|B| \cdot |R|'$, $(b/n)|A| \cdot |L'|$, $(b/n)|B| \cdot |R|'$ respectively.

Fact 31 *Suppose we pick a number P according to the Poisson distribution with parameter λ. Then, we distribute P balls into m bin as follows: We pick balls one by one and through them into random bins (independently). Then the number of balls in bins are independent and are distributed according to the Poisson distribution with parameter λ/m.*

In the model with outlier edges, D_2 is the distribution of the random variable $Z_{P_1} + \hat{\kappa}(Z_{P_1})$, where P_1 is a Poisson random variable with parameter bM (see Section B.1). Since $Y_1, Y_2 \sim D_2$, we may assume that $Y_1 = Z_{L'} + \hat{\kappa}(Z_{L'})$ and $Y_2 = Z_{R'} + \hat{\kappa}(Z_{R'})$ for some Poisson random variables $Z_{L'}$ and $Z_{R'}$ with parameter bM. If the test procedure added $Z_{L'}$ and $Z_{R'}$ edges between A and R'' and between B and L'', it would get a graph from the pure Stochastic Block Model with the planted cut $(A \cup L'', B \cup R'')$. But adding extra $\hat{\kappa}(Z_{L'})$ and $\hat{\kappa}(Z_{L'})$ edges it gets a graph from the SBM with outlier edges.

We showed that if $(A \cup L'', B \cup R'')$ is the planted cut, then $X_1, X_2 \sim D_1$ and $Y_1, Y_2 \sim D_2$; if $(B \cup L'', A \cup R'')$ is the planted cut, $X_1, X_2 \sim D_2$ and $Y_1, Y_2 \sim D_1$. This the restricted classifier outputs the correct cut with probability $1 - \delta$, this test errs also with probability δ. ■

We will need the following simple lemma.

Lemma 32 *Consider two distributions D_1 and D_2. Suppose that there exists a joint distribution D_{12} of random variables X and Y such that $X \sim D_1$ and $Y \sim D_2$, and*

$$\mathbb{P}(X = Y) \geq \eta.$$

Then, for any test for pairs of distributions D_1, D_2 (see above) errs with probability at least $\eta^4/2$.

Proof Consider four independent pairs of random variables (X_1, Y_1), (X_2, Y_2), (X_3, Y_3), and (X_4, Y_4). Each pair (X_i, Y_i) is distributed according to D_{12}. Let ζ be the error probability of T. Consider two experiments: In the first experiment we apply the test to the pairs (X_1, Y_2) and (X_3, Y_4); in the second, we apply the test to (Y_1, X_2) and (Y_3, X_4). Observe that the random variables X_1, Y_2, X_3 and Y_4 are independent; and $X_1 \sim D_1$, $Y_2 \sim D_2$, $X_3 \sim D_1$, $Y_4 \sim D_2$. The random variables Y_1, X_2, Y_3 and X_4 are also independent; but $Y_1 \sim D_2$, $Y_2 \sim D_1$, $X_3 \sim D_2$, $Y_4 \sim D_1$. So the test should output opposite results in the first and second experiments. However, with probability at least η^4, we get $X_1 = Y_1$, $X_2 = Y_2$, $X_3 = Y_3$, $X_4 = Y_4$. In this case, the test returns the incorrect answer either in the first or second experiments. ■

We now prove Theorem 27.

Proof [Proof of Theorem 27] By Corollary 36, which we prove in the next section, there exists a coupling of two Poisson random variables P_1, P_2 with parameters a and b, such that

$$\delta \equiv \mathbb{P}(P_1 = P_2) \geq C_1 e^{-\frac{C_2(a-b)^2}{a+b}}$$

for some absolute constants C_1 and C_2. By Lemma 32, the error probability of any test for P_1, P_2 is at least δ. Since the number of neighbours of a fixed vertex u on the same side and on the opposite side are distributed as the Poisson distribution with parameters a and b, by Lemma 30, we get that any restricted classifier has error probability at least δ. Finally, by Lemma 28, the expected number of misclassified vertices is at lest $\delta/2 - O(1/n) = (C_1/2)e^{-\frac{C_2(a-b)^2}{a+b}} - O(1/n)$. This proves the bound B.1.

In the model with outlier edges, the total number of edges between the set L' and L'' has the Poisson distributed with parameter $(a/n)|L'| \cdot |L \setminus L'| = a\rho(1-\rho)n$. The total number of edges between L' and R'' has the same distribution as $P_1 + \hat{\kappa}(P_1)$, where P_1 is the Poisson distribution with parameter b (see Corollary 39). By Lemma 32 and Corollary 39, the error probability of any test for these two distributions is at lest 1/2. Hence, by Lemma 30 and Lemma 29, the expected number of misclassified vertices is at least (see Section B.1)

$$\delta \geq \frac{|L'|}{2n} = \Omega\Big(\frac{\varepsilon(a-b)}{a+b}\Big).$$

This proves the bound B.2. ■

B.4. Poisson Distribution

Fact 33 (Median of the Poisson distribution) *For every Poisson random variable P with parameter $\lambda > 0$,*

$$\mathbb{P}(P \geq \lfloor \lambda \rfloor) \geq \frac{1}{2}.$$

Lemma 34 *There exists a constant $C > 0$ such that for a Poisson random variable with parameter $\lambda \geq 1$ and every $t \geq 1$, the following inequality holds:*

$$\mathbb{P}(P \geq \lambda + t\sqrt{\lambda}) \geq e^{-Ct^2}.$$

Proof Let $S' = \{k \in \mathbb{Z}^+ : \lfloor \lambda \rfloor \leq k < \lambda + t\sqrt{\lambda}\}$ and $S'' = \{k \in \mathbb{Z}^+ : k \geq \lambda + t\sqrt{\lambda}\}$. The union $S' \cup S''$ is the set of all integers greater than $\lfloor \lambda \rfloor$. Hence,

$$\mathbb{P}(P \in S' \cup S'') = \mathbb{P}(P \geq \lfloor \lambda \rfloor) \geq 1/2.$$

If $\mathbb{P}(P \in S') \leq 1/4$, then $\mathbb{P}(P \in S'') \geq 1/4$, and we are done. So we assume that $\mathbb{P}(P \in S') \geq 1/4$. Let $\Delta = \lceil t\sqrt{\lambda} \rceil + 1$. Notice that $S' + \Delta \equiv \{k + \Delta : k \in S'\} \subset S''$, and, consequently, $\mathbb{P}(P \in S'') \geq \mathbb{P}(P \in S' + \Delta)$. We lower bound $\mathbb{P}(P \in S' + \Delta)$ using the following lemma.

Lemma 35 *Let S be a subset of natural numbers. Suppose that all elements in S are upper bounded by K. Then,*

$$\frac{\mathbb{P}(P \in S)}{\mathbb{P}(P \in S + 1)} \leq 1 + \frac{K - \lambda + 1}{\lambda},$$

where P is a Poisson random variable with parameter λ.

Proof Write,

$$\frac{\mathbb{P}(P \in S)}{\mathbb{P}(P \in S + 1)} = \frac{\sum_{k \in S} \mathbb{P}(P = k)}{\sum_{k \in S} \mathbb{P}(P = k + 1)} \leq \max_{k \in S} \frac{\mathbb{P}(P = k)}{\mathbb{P}(P = k + 1)}.$$

For each $k \in S$, we have

$$\frac{\mathbb{P}(P = k)}{\mathbb{P}(P = k + 1)} = \frac{e^{-\lambda}\lambda^k/k!}{e^{-\lambda}\lambda^{k+1}/(k+1)!} = \frac{k+1}{\lambda} \leq \frac{K+1}{\lambda}.$$

Hence,

$$\frac{\mathbb{P}(P \in S)}{\mathbb{P}(P \in S + 1)} \leq 1 + \frac{K - \lambda + 1}{\lambda}$$

■

Applying this lemma Δ times to the set S with $K = \lambda + 2\Delta$, we get

$$\frac{\mathbb{P}(P \in S')}{\mathbb{P}(P \in S' + \Delta)} \leq \left(1 + \frac{2\Delta + 1}{\lambda}\right)^{\Delta} = \exp\left(\Delta \ln\left(1 + \frac{2\Delta + 1}{\lambda}\right)\right) \leq \exp\left(\frac{\Delta(2\Delta + 1)}{\lambda}\right).$$

Since $\mathbb{P}(P \in S') \geq 1/4$ and $\Delta = \lceil t\sqrt{\lambda} \rceil + 1$, we get for constant $C > 0$,

$$\mathbb{P}(P \in S' + \Delta) \geq \frac{e^{-\frac{2\Delta^2 + \Delta}{\lambda}}}{4} \geq \frac{e^{-3\Delta^2/\lambda}}{4} \geq e^{-Ct^2}.$$

This finishes the proof. ■

Corollary 36 (Coupling of two Poisson random variables) *There exists positive constants $C_1, C_2 > 0$ such that for all positive λ_1 and λ_2, there exists a joint distribution of two Poisson random variables P_1 and P_2 with parameters λ_1 and λ_2 such that*

$$\mathbb{P}(P_1 = P_2) \geq C_1 e^{-\frac{C_2(\lambda_1 - \lambda_2)^2}{\lambda_1 + \lambda_2}}.$$

Proof Consider the coupling of P_1 and P_2 that maximizes the probability of the event $\{P_1 = P_2\}$. The probability that P_1 and P_2 are equal can be expressed in terms of the total variation distance between the distributions of P_1 and P_2:

$$\mathbb{P}(P_1 = P_2) = 1 - \|P_1 - P_2\|_{TV} = \sum_{k=0}^{\infty} \min(\mathbb{P}(P_1 = k), \mathbb{P}(P_2 = k)).$$

Assume without loss of generality that $\lambda_1 \leq \lambda_2$. We now consider several cases.

I. If $\lambda_1 \geq 1$ and $\lambda_2 \leq 2\lambda_1$, then

$$\mathbb{P}(P_1 = P_2) \geq \sum_{k>\lambda_2}^{\infty} \min(\mathbb{P}(P_1 = k), \mathbb{P}(P_2 = k)) = \sum_{k>\lambda_2}^{\infty} \mathbb{P}(P_1 = k) = \mathbb{P}(P_1 \geq \lambda_2).$$

By Lemma 34,

$$\mathbb{P}(P_1 = P_2) \geq \mathbb{P}(P_1 \geq \lambda_2) \geq e^{-C\frac{(\lambda_1-\lambda_2)^2}{\lambda_1}} \geq e^{-9C\frac{(\lambda_1-\lambda_2)^2}{\lambda_1+\lambda_2}}.$$

II. If $\lambda_2 \geq 2\lambda_1$, then

$$\mathbb{P}(P_1 = P_2) \geq \min(\mathbb{P}(P_1 = 0), \mathbb{P}(P_2 = 0)) = \mathbb{P}(P_2 = 0) = e^{-\lambda_2} \geq e^{-\frac{9/2(\lambda_1-\lambda_2)^2}{\lambda_1+\lambda_2}}.$$

In the last inequality we used that

$$\frac{(\lambda_1 - \lambda_2)^2}{\lambda_1 + \lambda_2} \geq \frac{1/2\lambda_2}{3/2\lambda_2} = \frac{2\lambda_2}{9}.$$

III. Finally, if $\lambda_1 \leq 1$ and $\lambda_2 \leq 2\lambda_1$, then, as in the previous case,

$$\mathbb{P}(P_1 = P_2) \geq \mathbb{P}(P_2 = 0) = e^{-\lambda_2} \geq e^{-2} \geq e^{-2}e^{-\frac{(\lambda_1-\lambda_2)^2}{\lambda_1+\lambda_2}}.$$

This finishes the proof. ■

Lemma 37 *For every positive $\lambda_1 \leq \lambda_2$ there exists a joint distribution of two Poisson random variables P_1 and P_2 such that*

$$\mathbb{P}\left(P_2 \geq P_1 \text{ and } P_2 - P_1 \leq 2(\lambda_2 - \lambda_1)\right) \geq \frac{1}{2}.$$

Proof Observe that the Poisson distribution with parameter λ_2 stochastically dominates the Poisson distribution with parameter λ_1 (simply because a Poisson random with parameter λ_2 can be expressed as the sum of two independent Poisson random variables with parameters λ_1 and $\lambda_2 - \lambda_1$). Thus, there exists a coupling of P_1 and P_2 such that $P_2 \geq P_1$ a.s. We have $\mathbb{E}[P_2 - P_1] = \lambda_2 - \lambda_1$, and, by Markov's inequality, $\mathbb{P}((P_2 - P_1) \geq 2(\lambda_1 - \lambda_1)) \leq 1/2$. ■

Corollary 38 *For every positive $\lambda_1 < \lambda_2$, there exists a random function $\kappa : \mathbb{Z}^{\geq 0} \to \mathbb{Z}^{\geq 0}$ such that $P + \kappa(P)$ has the Poisson distribution with parameter λ_2 if P has the Poisson distribution with parameter λ_1 (P and κ are independent).*

Proof Consider Poisson random variables P_1 and P_2 as in Lemma 37. Let $\kappa(i) = j$ with probability $\mathbb{P}(P_2 = i + j \mid P_1 = i)$. Then, clearly, $\kappa(P_1)$ is distributed as P_2. ■

Corollary 39 *For every positive $\lambda_1 < \lambda_2$, there exists a random function $\hat{\kappa} : \mathbb{Z}^{\geq 0} \to \mathbb{Z}^{\geq 0}$ such that $\hat{\kappa}(P_1) \leq 2(\lambda_2 - \lambda_1)$ a.s. for a Poisson random variable P_1 with parameter λ_1 and there exists a coupled Poisson random variable P_2 with parameter λ_2 such that*

$$\mathbb{P}(P_2 = P_1 + \hat{\kappa}(P_1)) \geq 1/2.$$

Proof We let $\hat{\kappa}(i) = \min(\kappa(i), 2(\lambda_2 - \lambda_1))$ and $P_2 = P_1 + \kappa(P_1)$. Then, clearly $\hat{\kappa}(i) \leq 2(\lambda_2 - \lambda_1)$ and $P_1 + \hat{\kappa}(P_1) = P_1 + \kappa(P_1)$ with probability at least $1/2$. ■