Breaking the Small Cluster Barrier of Graph Clustering

Nir Ailon NAILON@CS.TECHNION.AC.IL
Department of Computer Science, Technion Israel Institute of Technology

Yudong Chen YDCHEN@UTEXAS.EDU
Department of Electrical and Computer Engineering, The University of Texas at Austin

Huan Xu MPEXUH@NUS.EDU.SG
Department of Mechanical Engineering, National University of Singapore

Abstract

This paper investigates graph clustering in the planted cluster model in the presence of *small clusters*. Traditional results dictate that for an algorithm to provably correctly recover the clusters, *all* clusters must be sufficiently large (in particular, $\tilde{\Omega}(\sqrt{n})$ where n is the number of nodes of the graph). We show that this is not really a restriction: by a more refined analysis of the trace-norm based matrix recovery approach proposed in Jalali et al. (2011) and Chen et al. (2012), we prove that small clusters, under certain mild assumptions, do not hinder recovery of large ones. Based on this result, we further devise an iterative algorithm to recover *almost all clusters* via a "peeling strategy", i.e., recover large clusters first, leading to a reduced problem, and repeat this procedure. These results are extended to the *partial observation* setting, in which only a (chosen) part of the graph is observed. The peeling strategy gives rise to an active learning algorithm, in which edges adjacent to smaller clusters are queried more often as large clusters are learned (and removed).

From a high level, this paper sheds novel insights on high-dimensional statistics and learning structured data, by presenting a structured matrix learning problem for which a one shot convex relaxation approach necessarily fails, but a carefully constructed sequence of convex relaxations does the job.

Proceedings of the 30^{th} International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

1. Introduction

This paper considers a classic problem in machine learning and theoretical computer science, namely graph clustering, i.e., given an undirected unweighted graph, partition the nodes into disjoint clusters, so that the density of edges within one cluster is higher than those across clusters. Graph clustering arises naturally in many applications across science and engineering, such as community detection in social network, submarket identification in E-commerce and sponsored search and co-authorship analysis in analyzing document database. From a purely binary classification theoretical point of view, the edges of the graph are (noisy) labels of *similarity* or *affinity* between pairs of objects, and the concept class consists of clusterings of the objects (encoded graphically by identifying clusters with cliques).

Many theoretical results in graph clustering (e.g., Condon & Karp, 2001; McSherry, 2001) consider the planted partition model, in which the edges are generated randomly; see Section 1.1 for more details. While numerous different methods have been proposed, their performance guarantees all share the following manner – under certain condition of the density of edges (within clusters and across clusters), the proposed method succeeds to recover the correct clusters exactly *if all clusters are larger than a threshold size*, typically $\tilde{\Omega}(\sqrt{n})$. For algorithms relying on spectral analysis the reason for this requirement is simple: The random noise gives rise to eigenvalues of order $\tilde{\Omega}(\sqrt{n})$ in the graph adjacency matrix, dominating spectral information corresponding to the clusters if they are all small.

In this paper, we aim to break this **small cluster barrier** of graph clustering for a certain family of algorithms based on convex relaxation. When all the clusters are very small, identifying them seems inher-

ently hard[1], and is not the focus of this paper. Instead, in this paper we investigate the following: Can we still recover large clusters in the presence of small clusters? Intuitively, this should be doable. To illustrate, consider the example where the given graph G consists two disjoint subgraphs G_1 and G_2, where G_1 by itself is a graph that can be correctly clustered using some existing method, G_2 is a very small clique, and there are only relative few edges connecting G_1 and G_2. G certainly violates the minimum cluster size requirement of previous results, but why should G_2 spoil our ability to recover G_1?

Our main result confirms this intuition. We show that the cluster size barrier arising in previous work (e.g., Chaudhuri et al., 2012; Bollobás & Scott, 2004; Chen et al., 2012; McSherry, 2001) is not really a restriction, but rather an artifact of the attempt to solve the problem in a single shot. Using a more careful analysis, we prove that the mixed trace-norm and ℓ_1 based convex formulation, initially proposed in Jalali et al. (2011) and Chen et al. (2012), can recover clusters of size $\tilde{\Omega}(\sqrt{n})$ even in the presence of smaller clusters.

The main implication of this result is that one can apply an iterative "peeling" strategy, recovering smaller and smaller clusters. The intuition is simple – suppose the *number* of clusters is limited, then either all clusters are large, or the sizes of the clusters vary significantly. The first case is already covered by existing results. The second one is equally easy: use the aforementioned convex formulation, the larger clusters can be correctly identified. If we remove all nodes from these larger clusters, the remaining subgraph contains significantly fewer nodes than the original graph, which leads to a much lower threshold on the size of the cluster for correct recovery, making it possible for correctly clustering some remaining smaller clusters. By repeating this procedure, indeed, we can recover the cluster structure for almost all nodes *with no lower bound on the minimal cluster size*. We summarize our main contributions and techniques:

(1) We provide a refined analysis of the mixed trace norm and ℓ_1 convex relaxation approach for exact recovery of clusters proposed in Jalali et al. (2011) and Chen et al. (2012), focusing on the case where small clusters exist. We show that if there is a number x in $\tilde{\Omega}(\sqrt{n})$ such that each cluster is either larger than x or smaller than $x/\log^2 n$, and at least one cluster is large, then with high probability, the convex relaxation leads to a unique solution correctly identifying all big clusters while "ignoring" the small ones. We call such a solusion *admissible*. Notice that the multiplicative gap between the two thresholds is logarithmic w.r.t. n. In addition, it is possible to arbitrarily increase x, thus turning a "knob" in quest of an interval $(x/\log^2 n, x)$ that is disjoint from the set of cluster sizes. The analysis is done by identifying a certain feasible solution to the convex program and proving its almost sure optimality using a careful construction of a *dual certificate*. This method has been performed before only in the case where all clusters are large.

(2) We provide a converse of the result just described. More precisely, we show that if for some value of the knob x an optimal solution is admissible, then the solution is useful (in the sense that it correctly identifies big clusters), even if there exist clusters in the interval $(x/\log^2 n, x)$.

(3) The last two points imply that if some interval of the form $(x/\log^2 n, x)$ is free of cluster sizes, then an exhaustive search of this interval will constructively find big clusters. This gives rise to an iterative algorithm, using a "peeling strategy", to recover smaller and smaller clusters that were not recoverable in a one shot convex relaxation step. We then prove that as long as the *number* of clusters is bounded by $\Omega(\log n/\log\log n)$, regardless of the cluster sizes, we can correctly recover the cluster structure for an overwhelming fraction of nodes.

(4) We extend the result to the partial observation case, where only a fraction of similarity labels (i.e., edge/no edge) is known. As expected, smaller observation rates allow identification of larger clusters. Hence, the observation rate serves as the "knob". This gives rise to an *active learning algorithm* for graph clustering based on adaptively increasing the rate of sampling in order to hit a corresponding interval free of cluster sizes, and concentrating on smaller inputs as we identify big clusters and peel them off.

Beside these technical contributions, this paper provides novel insights into low-rank matrix recovery and more generally high-dimensional statistics, where data are typically assumed to obey certain low-dimensional structure. Numerous methods have been developed to exploit this *a priori* information so that a consistent estimator is possible even when the dimensionality of data is larger than the number of samples. Our result shows that one may combine these methods with a "peeling strategy" to further push the envelope of learning structured data – By iteratively recovering the easier structure and then reducing the problem

[1]Indeed, even in a more lenient setup where one clique (i.e., a perfect cluster) of size K is embedded in an Erdos-Renyi graph of n nodes and 0.5 probability of forming an edge, to recover this clique, the best known polynomial method requires $K = \Omega(\sqrt{n})$ and it has been a long standing open problem to relax this requirement.

size, it is possible to learn structures that are otherwise difficult using previous approaches.

1.1. Previous work

The literature of graph clustering is too vast for a detailed survey here; we concentrate on the most related work, and in specific those provide theoretical guarantees on cluster recovery.

Planted partition model: The setup we study is the classical *planted partition* model (Condon & Karp, 2001), also known as the *stochastic block* model (Holland et al., 1983). Here, n nodes are partitioned into subsets, referred as the "true clusters", and a graph is randomly generated as follows: for each pair of nodes, depending on whether they belong to a same subset, an edge connecting them is generated with a probability p or q respectively. The goal is to correctly recover the clusters given the random graph. Earlier work on the planted partition model focused on the 2-partition or more generally l-partition case with $l = O(1)$, i.e., the minimal cluster size is $\Theta(n)$ (Condon & Karp, 2001; Carson & Impagliazzo, 2001; Bollobás & Scott, 2004). Recently, several works have proposed methods to handle sublinear cluster sizes. These works can be roughly classified into three approaches: randomized algorithms (e.g., Shamir & Tsur, 2007), spectral clustering (e.g., McSherry, 2001; Giesen & Mitsche, 2005; Chaudhuri et al., 2012; Rohe et al., 2011)), and algorithms based on convex optimization (Jalali et al., 2011; Chen et al., 2012; Ames & Vavasis, 2011; Oymak & Hassibi, 2011; Mathieu & Schudy, 2010). While these work differs in the methodology, they all impose constraints on the size of the minimum true cluster – the best result up-to-date requires it to be $\tilde{\Omega}(\sqrt{n})$.

Correlation Clustering This problem, originally defined by Bansal et al. (2004), also considers graph clustering but in an adversarial noise setting. The problem is NP-Hard to approximate to within some constant factor. Prominent work includes Demaine et al. (2006); Ailon et al. (2008); Charikar et al. (2005). A PTAS is known in case the number of clusters is fixed (Giotis & Guruswami, 2006).

Low rank matrix decomposition via trace norm: Motivated from robust PCA, it has recently been shown (Chandrasekaran et al., 2011; Candès et al., 2011), that it is possible to recover a low-rank matrix from sparse errors of arbitrary magnitude, where the key ingredient is using trace norm (a.k.a. nuclear norm) as a convex surrogate of the rank. A similar result is also obtained when the low rank matrix is corrupted by other types of noise (Xu et al., 2012). Of particular relevance to this paper is Jalali et al. (2011), Chen et al. (2012) and Jalali & Srebro (2012), where the authors apply this approach to graph clustering, and specifically to the planted partition model. Indeed, Chen et al. (2012) achieve state-of-art performance guarantees for the planted partition problem. However, they don't overcome the $\tilde{\Omega}(\sqrt{n})$ minimal cluster size lower bound.

Active learning/Active clustering Another line of work that motivates this paper is study of active learning algorithms (a settings in which labeled instances are chosen by the learner, rather than by nature), and in particular active learning for clustering. The most related work is Ailon et al. (2012), who investigated active learning for correlation clustering. The authors obtain a $(1+\varepsilon)$-approximate solution with respect to the optimal, while (actively) querying no more than $O(n\,\mathrm{poly}(\log n, k, \varepsilon^{-1}))$ edges. The result imposed no restriction on cluster sizes and hence inspired this work, but differs in at least two major ways. First, Ailon et al. (2012) did not consider *exact recovery* as we do. Second, their guarantees fall in the ERM (Empirical Risk Minimization) framework, with no running time guarantees. Our work recovers true cluster exactly using a convex relaxation algorithm, and is hence computationally efficient. The problem of active learning has also been investigated in other clustering setups including clustering based on distance matrix (Voevodski et al., 2012; Shamir & Tishby, 2011), and hierarchical clustering (Eriksson et al., 2011; Krishnamurthy et al., 2012). These setups differ from ours and cannot be easily compared.

2. Notation and Setup

Throughout, V denotes a ground set of elements, which we identify with the set $[n] = \{1, \ldots, n\}$. We assume a true ground truth clustering of V given by a pairwise disjoint covering $V_1, \ldots, V_k$, where k is the number of clusters. We say $i \sim j$ if $i, j \in V_a$ for some $a \in [k]$, otherwise $i \not\sim j$. We let $n_i = |V_i|$ for all $i \in [k]$. For any $i \in [n]$, $\langle i \rangle$ is the unique index satisfying $i \in V_{\langle i \rangle}$.

For a matrix $X \in \mathbb{R}^{n \times n}$ and a subset $S \subseteq [n]$ of size m, the matrix $X[S] \in \mathbb{R}^{m \times m}$ is the principal minor of X corresponding to the set of indexes S. For a matrix M, $\Gamma(M)$ denotes the support of M, namely, the set of index pairs (i, j) such that $M(i, j) \neq 0$.

The ground truth clustering matrix, denoted K^*, is defined so that $K^*(i, j) = 1$ is $i \sim j$, otherwise 0. This is a block diagonal matrix, each block consisting of 1's only. Its rank is k. The input is a symmetric matrix A, a noisy version of K^*. It is generated using

the well known *planted clustering* model, as follows. There are two fixed edge probabilities, $p > q$. We think of A as the adjacency matrix of an undirected random graph, where edge (i, j) is in the graph for $i > j$ with probability p if $i \sim j$, otherwise with probability q, independent of other choices. The error matrix is denoted by $B^* := A - K^*$. We let $\Omega := \Gamma(B^*)$ denote the *noise locations*.

Note that our results apply to the more practical case in which the edge probability of (i, j) is p_{ij} for each $i \sim j$ and q_{ij} for $i \not\sim j$, as long as $(\min p_{ij}) =: p > q := (\max q_{ij})$.

3. Results

We remind the reader that the trace norm of a matrix is the sum of its singular values, and we define the ℓ_1 norm of a matrix M to be $\|M\|_1 = \sum_{ij} |M(ij)|$. For a set $\Phi \subseteq [n] \times [n]$, $\mathcal{P}_\Phi(M)$ denotes the matrix obtained from M by setting $M(i, j) = 0$ for all $(i, j) \in \Phi$. Consider the following convex program, combining the trace norm of a matrix variable K with the ℓ_1 norm of another matrix variable B using two parameters c_1, c_2 that will be determined later:

$$\begin{aligned} \text{(CP1)} \quad \min \quad & \|K\|_* + c_1 \left\|\mathcal{P}_{\Gamma(A)} B\right\|_1 + c_2 \left\|\mathcal{P}_{\Gamma(A)^c} B\right\|_1 \\ \text{s.t.} \quad & K + B = A \\ & 0 \le K_{ij} \le 1, \forall (i, j). \end{aligned}$$

Theorem 1. *There exist constants $b_1, b_3, b_4 > 0$ such that the following holds with probability at least $1 - n^{-3}$. For any parameter $\kappa \ge 1$ and $t \in [\frac{1}{4}p + \frac{3}{4}q, \frac{3}{4}p + \frac{1}{4}q]$, define*

$$\ell_\sharp = b_3 \frac{\kappa\sqrt{p(1-q)n}}{p-q} \log^2 n \quad \ell_\flat = b_4 \frac{\kappa\sqrt{p(1-q)n}}{p-q} \ . \quad (1)$$

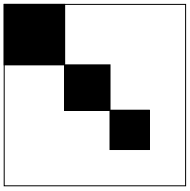
If for all $i \in [k]$, either $n_i \ge \ell_\sharp$ or $n_i \le \ell_\flat$ and if $(\hat{K}, \hat{B})$ is an optimal solution to (CP1), with

$$c_1 = \frac{b_1}{\kappa\sqrt{n \log n}} \sqrt{\frac{1-t}{t}} \quad c_2 = \frac{b_1}{\kappa\sqrt{n \log n}} \sqrt{\frac{t}{1-t}} \ , \quad (2)$$

then $(\hat{K}, \hat{B}) = (\mathcal{P}_\sharp K^*, A - \hat{K})$, where for a matrix M, $\mathcal{P}_\sharp M$ is the matrix defined by

$$(\mathcal{P}_\sharp M)(i, j) = \begin{cases} M(i, j) & \max\{n_{\langle i \rangle}, n_{\langle j \rangle}\} \ge \ell_\sharp \\ 0 & \text{otherwise} \end{cases} .$$

(Note that by the theorem's premise, $\hat{K}$ is the matrix obtained from K^* after zeroing out blocks corresponding to clusters of size at most $\ell_\flat$.) The proof is based on Chen et al. (2012) and is deferred to the supplemental material due to lack of space. The main novelty in this work compared to previous work is the treatment of small clusters of size at most $\ell_\flat$, whereas in previous work only large clusters were treated, and the existence of small clusters did not allow recovery of the big clusters.



Black represents 1, white represents 0. $\sigma_{\min}(K)$ is the side length of the smallest black square.

Figure 1. A partial clustering matrix K.

Definition 2. An $n \times n$ matrix K is a partial clustering matrix if there exists a collection of pairwise disjoint sets $U_1, \ldots, U_r \subseteq [n]$ (the *induced clusters*) such that $K(i, j) = 1$ if and only if $i, j \in U_s$ for some $s \in [r]$, otherwise 0. If K is a partial clustering matrix then $\sigma_{\min}(K)$ is defined as $\min_{i=1}^r |U_i|$.

The definition is depicted in Figure 1. Theorem 1 tells us that by choosing κ (and hence c_1, c_2) properly such that no cluster size falls in the range $(\ell_\flat, \ell_\sharp)$, the unique optimal solution $(\hat{K}, \hat{B})$ to convex program (CP1) is such that $\hat{K}$ is a partial clustering induced by big ground truth clusters.

In order for this fact to be useful algorithmically, we also need a type of converse: there exists an event with high probability (in the random process generating the input), such that for all values of κ, if an optimal solution to the corresponding (CP1) looks like the solution $(\hat{K}, \hat{B})$ defined in Theorem 1, then the blocks of $\hat{K}$ correspond to actual clusters.

Theorem 3. *There exists constants $C_1, C_2 > 0$ such that with probability at least $1 - n^{-2}$, the following holds. For all $\kappa \ge 1$ and $t \in [\frac{3}{4}q + \frac{1}{4}p, \frac{1}{4}q + \frac{3}{4}p]$, if (K, B) is an optimal solution to (CP1) with c_1, c_2 as defined in Theorem 1, and additionally K is a partial clustering induced by $U_1, \ldots, U_r \subseteq V$, and also*

$$\sigma_{\min}(K) \ge \max\left\{ \frac{C_1 k \log n}{(p-q)^2}, \frac{C_2 \kappa\sqrt{p(1-q)n \log n}}{p-q} \right\} , \quad (3)$$

then $U_1, \ldots, U_r$ are actual ground truth clusters, namely, there exists an injection $\phi : [r] \mapsto [k]$ such that $U_i = V_{\phi(i)}$ for all $i \in [r]$.

(Note: Our proof of Theorem 3 uses Hoeffding tail bounds for simplicity, which are tight for p, q bounded away from 0 and 1. Bernstein tail bounds can be used to strengthen the result for other classes of p, q. We elaborate on this in Section 3.1.)

The combination of Theorems 1 and 3 implies that, as long as there exists a relatively small interval which is disjoint from the set of cluster sizes, and such that at least one cluster size is larger than this interval (and large enough), we can recover at least one (large) cluster using (CP1). This is made clear in the following.

Corollary 4. *Assume we have a guarantee that there exists a number* $\alpha \geq b_4 \frac{\sqrt{p(1-q)n}}{p-q}$, *such that no cluster size falls in the interval* $(\alpha, \frac{b_3}{b_4}\alpha \log^2 n)$ *and at least one cluster size is of size at least* $s := \max\{\frac{b_3}{b_4}\alpha \log^2 n, (C_1 k \log n)/(p-q)^2, C_2\sqrt{p(1-q)n \log n}/(p-q)\}$. *Then with probability at least* $1 - n^{-2}$, *we can recover at least one cluster of size at least* s *efficiently by solving (CP1) with* $\kappa = \alpha / \left(b_4 \frac{\sqrt{p(1-q)n}}{p-q}\right)$.

Of course we do not know what α (and hence κ) is. We could exhaustively search for a $\kappa \geq 1$ and hope to recover at least one large cluster. A more interesting question is, when is such a κ guaranteed to exist? Let $g = \frac{b_3}{b_4}\log^2 n$. The number g is the (multiplicative) gap size, equaling the ratio between $\ell_\sharp$ and $\ell_\flat$ (for any κ). If the number of clusters k is a priori bounded by some k_0, we both ensure that there is at least one cluster of size n/k_0, and by the pigeonhole principle, that one of the intervals in the sequence $(n/gk_0, n/k_0), (n/g^2k_0, n/gk_0), \ldots, (n/g^{k_0+1}k_0, n/g^{k_0}k_0)$. is disjoint of cluster sizes. If, in addition, the smallest interval in the sequence is not too small and n/k_0 is not too small so that Corollary 4 holds, then we are guaranteed to recover at least one cluster using Algorithm 1. We find this condition difficult to work with. An elegant, useful version of the idea is obtained if we assume p, q are some fixed constants.[2] As the following lemma shows, in this regime, k_0 can be assumed to be almost logarithmic in n to ensure recovery of at least one cluster.[3] In what follows, notation such as $C(p,q), C_3(p,q), \ldots$ denotes universal positive functions depending on p, q only.

Lemma 5. *There exists* $C_3(p,q), C_4(p,q), C_5 > 0$ *such that the following holds. Assume that* $n > C_4(p,q)$, *and that we are guaranteed that* $k \leq k_0$, *where* $k_0 = \frac{C_3(p,q)\log n}{\log\log n}$. *Then with probability at least* $1 - n^{-2}$ *Algorithm 1 will recover at least one cluster in at most* $C_5 k_0$ *iterations.*

The proof is deferred to the supplemental material section. Lemma 5 ensures that by trying at most a logarithmic number of values of κ, we can recover at least one large cluster, assuming the number of clusters is roughly logarithmic in n. The next proposition tells us that as long as this step recovers the clusters covering at most all but a vanishing fraction of elements, the step can be repeated.

Proposition 6. *A pair of numbers* (n', k') *is called good if* $n' \leq n, k' \leq k$ *and* $k' \leq \frac{C_3(p,q)\log n'}{\log\log n}$. *If* (n', k') *is good, then* (n'', k'') *is good for all* n'', k'' *satisfying* $n' \geq n'' \geq n'/(\log n)^{1/C_3(p,q)}$ *and* $k' - 1 \geq k'' \geq 1$.

The proof is trivial. The proposition implies an inductive process in which at least one big (with respect to the current unrecovered size) cluster can be efficiently removed as long as the previous step recovered at most a $(1 - (\log n)^{-1/C_3(p,q)})$-fraction of its input. Combining, we proved the following:

Theorem 7. *Assume* n, k *satisfy the requirements of Lemma 5. Then with probability at least* $1 - 2n^{-1}$ *Algorithm 2 recovers clusters covering all but at most a* $((\log n)^{-1/C_3(p-q)})$ *fraction of the input in the full observation case, without any restriction of the minimal cluster size. Moreover, if we assume that* k *is bounded by a* constant k_0, *then the algorithm will recover clusters covering all but a constant number of input elements.*

3.1. Partial Observations

We now consider the case where the input matrix A is not given to us in entirety, but rather that we have oracle access to $A(i,j)$ for (i,j) of our choice. Unobserved

Algorithm 1 RecoverBigFullObs(V, A, p, q)

require: ground set V, $A \in \mathbb{R}^{V\times V}$, probs p, q
$n \leftarrow |V|$
$t \leftarrow \frac{1}{4}p + \frac{3}{4}q$ (or anything in $[\frac{1}{4}p + \frac{3}{4}q, \frac{3}{4}p + \frac{1}{4}q]$)
$\ell_\sharp \leftarrow n$, $g \leftarrow \frac{b_3}{b_4}\log^2 n$
// (If have prior bound k_0 on num clusters,
// take $\ell_\sharp \leftarrow n/k_0$)
while $\ell_\sharp \geq \max\left\{\frac{C_1 k \log n}{(p-q)^2}, \frac{C_2\sqrt{p(1-q)n\log n}}{p-q}\right\}$ **do**
  solve for κ using (1), set c_1, c_2 as in (2)
  $(\hat{K}, \hat{B}) \leftarrow$ optimal solution to (CP1) with c_1, c_2
  if $\hat{K}$ partial clustering matrix with $\sigma_{\min}(\hat{K}) \geq \ell_\sharp$ **then**
    return induced clusters $\{U_1, \ldots, U_r\}$ of $\hat{K}$
  end if
  $\ell_\sharp \leftarrow \ell_\sharp / g$
end while
return $\emptyset$

[2] In fact, we need only fix $(p-q)$, but we wish to keep this exposition simple.

[3] In comparison, (Ailon et al., 2012) require k_0 to be constant for their guarantees, as do the Correlation Clustering PTAS (Giotis & Guruswami, 2006).

Algorithm 2 RecoverFullObs(V, A, p, q)

require: ground set V, matrix $A \in \mathbb{R}^{V\times V}$, probs p, q
$\{U_1, \dots, U_r\} \leftarrow$ RecoverBigFullObs(V, A, p, q)
$V' \leftarrow [n] \setminus (U_1 \cup \dots \cup U_r)$
if $r = 0$ **then**
  return $\emptyset$
else
  return RecoverFullObs($V', A[V'], p, q$) $\cup$ $\{U_1, \dots, U_r\}$
end if

values are formally marked with $A(i,j) =?$.

Consider a more particular setting in which the edge probabilities defining A are p' (for $i \sim j$) and q' (for $i \not\sim j$), and we observe $A(i,j)$ with probability ρ, for each i, j, independently. More precisely: For $i \sim j$ we have $A(i,j) = 1$ with probability $\rho p'$, 0 with probability $\rho(1-p')$ and ? with remaining probability. For $i \not\sim j$ we have $A(i,j) = 1$ with probability $\rho q'$, 0 with probability $\rho(1-q')$ and ? with remaining probability. Clearly, by pretending that the values ? in A are 0, we emulate the full observation case with $p = \rho p'$, $q = \rho q'$.

Of particular interest is the case in which p', q' are held fixed and ρ tends to zero as n grows. In this regime, we have the following result, which follows directly from Theorem 1 by setting $\kappa = 1$, $p = \rho p'$ and $q = \rho q'$ (note that Theorem 1 allows p and q to be $o(1)$).

Corollary 8. *There exist constants $b_1(p',q'), b_3(p',q'), b_4(p',q'), b_5(p',q') > 0$ such that for any sampling rate parameter ρ the following holds with probability at least $1 - n^{-3}$. define*

$$\ell_\sharp = b_3(p',q')\frac{\sqrt{n}}{\sqrt{\rho}}\log^2 n \qquad \ell_\flat = b_4(p',q')\frac{\sqrt{n}}{\sqrt{\rho}}\ .$$

If for all $i \in [k]$, either $n_i \geq \ell_\sharp$ or $n_i \leq \ell_\flat$ and if $(\hat{K}, \hat{B})$ is an optimal solution to (CP1), with

$$c_1 = \frac{b_1(p',q')}{\sqrt{n\log n}}\sqrt{\frac{1-b_5(p',q')\rho}{b_5(p',q')\rho}}$$
$$c_2 = \frac{b_1(p',q')}{\sqrt{n\log n}}\sqrt{\frac{b_5(p',q')}{1-b_5(p',q')\rho}}\ ,$$

then $(\hat{K},\hat{B})$=$(\mathcal{P}_\sharp K^*, A$–$\hat{K})$, with $\mathcal{P}_\sharp$ defined in Theorem 1.

(Note: We've abused notation by reusing previously defined global constants (e.g. b_1) with global *functions* of p', q' (e.g. $b_1(p',q')$).) Notice now that the observation probability ρ can be used as a knob for controlling the cluster sizes we are trying to recover, instead of κ. We would also like to obtain a version of Theorem 3. In particular, we would like to understand its asymptotics as ρ tends to 0.

Theorem 9. *There exist constants $C_1(p',q'), C_2(p',q') > 0$ such that for all observation rate parameters $\rho \leq 1$, the following holds with probability at least $1 - n^{-2}$. If (K, B) is an optimal solution to (CP1) with c_1, c_2 as defined in Theorem 8, and additionally K is a partial clustering induced by $U_1, \dots, U_r \subseteq V$, and also*

$$\sigma_{\min}(K) \geq \max\left\{\frac{C_1(p',q')k\log n}{\rho}, \frac{C_2(p',q')\sqrt{n\log n}}{\sqrt{\rho}}\right\}, \tag{4}$$

then $U_1, \dots, U_r$ are actual ground truth clusters, namely, there exists an injection $\phi : [r] \mapsto [k]$ such that $U_i = V_{\phi(i)}$ for all $i \in [r]$.

The proof is given in the supplemental material. Using the same reasoning as before, we derive the following:

Theorem 10. *Let $g = b_3(p',q')/b_4(p',q')\log^2 n$ (with $b_3(p',q'), b_4(p',q')$ defined in Corollary 8). There exists a constant $C_4(p',q')$ such that the following holds. Assume the number of clusters k is bounded by some known number $k_0 \leq C_4(p',q')(\log n)/(\log\log n)$. Let $\rho_0 = \frac{b_3(p',q')^2 k_0^2 \log^4 n}{n}$. Then there exists ρ in the set $\{\rho_0, \rho_0 g, \dots, \rho_0 g^{k_0}\}$ for which, if A is obtained with observation rate ρ (zeroing ?'s), then with probability at least $1-n^{-2}$, any optimal solution (K, B) to (CP1) with c_1, c_2 from Corollary 8 satisfies (4).*

(Note that the upper bound on k_0 ensures that ρg^{k_0} is a probability.) The theorem is proven using the pigeonhole principle, noting that one of the intervals $(\ell_\flat(\rho), \ell_\sharp(\rho))$ must be disjoint from the set of cluster sizes, and there is at least one cluster of size at least n/k_0. The theorem, together with Corollary 8 and Theorem 9 ensures the following. On one end of the spectrum, if k_0 is constant (and n is large), then with high probability we can recover at least one large cluster (of size at least n/k_0) after querying no more than

$$O\left(nk_0^2\left(\frac{b_3(p',q')}{b_4(p',q')}\log^2 n\right)^{2k_0}\log^4 n\right) \tag{5}$$

values of $A(i,j)$. On the other end of the spectrum, if $k_0 \leq \delta(\log n)/(\log\log n)$ and n is large enough (exponential in $1/\delta$), then we can recover at least one large cluster after querying no more than $n^{1+O(\delta)}$ values of $A(i,j)$. (We omit the details of the last fact from this version.) This is summarized in the following:

Theorem 11. *Assume an upper bound k_0 on the number of clusters k. As long as n is larger than some function of k_0, p', q', Algorithm 4 will recover, with probability at least $1-n^{-1}$, at least one cluster of size at least*

n/k_0, regardless of the size of other (small) clusters. Moreover, if k_0 is a constant, then clusters covering all but a constant number of elements will be recovered with probability at least $1 - n^{-1}$, and the total number of observation queries is (5), hence almost linear.

Unlike previous results, our recovery guarantee imposes no lower bounds on the size of the smallest cluster. Note that the underlying algorithm is an *active learning* one, as more observations fall in smaller clusters which survive deeper in the recursion of Alg. 4.

Algorithm 3 RecoverBigPartialObs(V, k_0)
(Assume p', q' known, fixed)

require: ground set V, oracle access to $A \in \mathbb{R}^{V \times V}$, upper bound k_0 on number of clusters
$n \leftarrow |V|$
$\rho_0 \leftarrow \frac{b_3(p',q')^2 k_0^2 \log^4 n}{n}$
$g \leftarrow b_3(p', q')/b_4(p', q') \log^2 n$
for $s \in \{0, \dots, k_0\}$ **do**
  $\rho \leftarrow \rho_0 g^s$
  obtain matrix $A \in \{0, 1, ?\}^{V \times V}$ by sampling oracle at rate ρ, then zero ? values in A
  // (can reuse observations from prev. iterations)
  $c_1(p', q'), c_2(p', q') \leftarrow$ as in Corollary 8
  $(K, B) \leftarrow$ an optimal solution to (CP1)
  if K partial clustering matrix satisfying (4) **then**
    return induced clusters $\{U_1, \dots, U_r\}$
  end if
end for
return $\emptyset$

Algorithm 4 RecoverPartialObs(V, k_0)
(Assume p', q' known, fixed)

require: ground set V, oracle access to $A \in \mathbb{R}^{V \times V}$, upper bound k_0 on number of clusters
$\{U_1, \dots, U_r\} \leftarrow$ RecoverBigPartialObs(V, k_0)
$V' \leftarrow [n] \setminus (U_1 \cup \dots \cup U_r)$
if $r = 0$ **then**
  return $\emptyset$
else
  return RecoverPartialObs$(V', k_0 - r) \cup \{U_1, \dots, U_r\}$
end if

4. Experiments

We experimented with simplified versions of our algorithms. Here we did not make an effort to compute the precise values of various constants defining the algorithms in this work, creating a difficulty in exact implementation. Instead, we assume p and q is known in (CP1), and set c_1, c_2 according to (2) with $t = \frac{1}{4}p + \frac{3}{4}q$ and $b_1 = 2$. For Algorithm 1, we start with $\kappa = 1$ and multiply it by 1.1 in each iteration until a partial clustering matrix is found. In Algorithm 3, ρ is increased by an additive factor of 0.025. Still, it is obvious that our experiments support our theoretical findings. A practical "user's guide" for this method with actual constants is subject to future work.

We use the Augmented Lagrangian Multiplier (ALM) method described in (Chen et al., 2012) to solve (CP1). In the sequel, whenever we say that "clusters $\{V_{i_1}, V_{i_2}, \dots\}$ were recovered", we mean that (CP1) resulted in an optimal solution $(\hat{K}, \hat{B})$ with $\hat{K}$ being a partial clustering matrix induced by $\{V_{i_1}, V_{i_2}, \dots\}$.

Experiment 1 (Full Observation) Consider $n = 1100$ nodes partitioned into 4 clusters $V_1, \dots, V_4$, of sizes $800, 200, 80, 20$, respectively. The graph is generated according to the planted partition model with $p = 0.7$, $q = 0.3$, and we assume full observation. We apply our algorithm and check if it successfully recovers all the clusters. We repeat for 20 times and observe 90% success. Table 1 shows one of the 20 execution; the algorithm terminates in 2 iterations and the recovered clusters at each iteration are shown.

Experiment 2 (Partial Observation - Fixed Sampling Rate) We have $n = 1100$ with clusters $V_1, \dots, V_4$ of sizes $800, 200, 50, 50$. The graph is generated with $p' = 0.7$, $q' = 0.1$, and observation rate $\rho = 0.3$. Out of 20 instances, our algorithm succeeds for 70% of the time. One such instance is shown in Table 1. In the other instances, only V_1 and V_2 are recovered, probably because the remaining graph is too small for exact recovery under random noise.

Experiment 3 (Partial Observation - Incremental Sampling Rate) We test Algorithm 4. We have $n = 1100$ with clusters $V_1, \dots, V_4$ of sizes $800, 200, 50, 50$. The observed graph is generated with $p' = 0.7$, $q' = 0.3$, and an observation rate ρ which we now specify. We start with $\rho = 0$ and increase it by 0.025 incrementally until we recover (and then remove) at least one cluster, then repeat. In all 20 instances, our algorithm recovers all the clusters when it terminates. Table 1 show one typical instance.

Experiment 3A We repeat the last experiment with a larger graph: $n = 4500$ with clusters $V_1, \dots, V_6$ of sizes $3200, 800, 200, 200, 50, 50$, and $p' = 0.8, q' = 0.2$. One execution is shown in Table 1. Note that we recover the smallest clusters, whose size is below $\sqrt{n}$.

Experiment 4 (Mid-Size Clusters) Our current theoretical results do not say anything about the

Experiment 1:

Iter.	κ	# nodes left	Clusters recovered
1	1	1100	V_1, V_2, V_3
2	2.41	20	V_4

Experiment 2:

Iter.	κ	# nodes left	Clusters recovered
1	1	1100	V_1,V_2
2	1	100	V_3, V_4

Experiment 3:

Iter.	ρ	# nodes left	Clusters recovered
1	0.2	1100	V_1
2	0.4	300	V_2
3	0.95	100	V_3, V_4

Experiment 3A:

Iter.	ρ	# nodes left	Clusters recovered
1	0.15	4500	V_1
2	0.175	1300	V_2
3	0.2	500	V_3, V_4
4	0.475	100	V_5, V_6

Table 1. Experiment Results

mid-size clusters – those with sizes between $\ell_\flat$ and $\ell_\sharp$. It is interesting to study the behavior of (CP1) in the presence of mid-size clusters. We generated an instance with $n = 750$, $\{|V_1|, |V_2|, |V_3|, |V_4|\} = \{500, 150, 70, 30\}$, $p = 0.8, q = 0.2$, and $\rho = 0.12$. We then solved (CP1) with a fixed $\kappa = 1$. The low-rank part $\hat{K}$ of the solution is shown in Fig. 2. The large cluster V_1 is completely recovered in $\hat{K}$, while the small clusters V_3 and V_4 are entirely ignored. The mid-size cluster V_2, however, exhibits a pattern we find difficult to characterize. This shows that the gap between $\ell_\sharp$ and $\ell_\flat$ in our theorems is a real phenomenon and not an artifact of our proof technique. Nevertheless, the large cluster appears clean, and might allow recovery using simple procedures. If this is true in general, it might not be necessary to search for a gap free of cluster sizes. Perhaps for any κ, (CP1) identifies all large clusters above $\ell_\sharp$ after a possible simple mid-size cleanup procedure. Understanding this phenomenon and its algorithmic implications is of much interest.

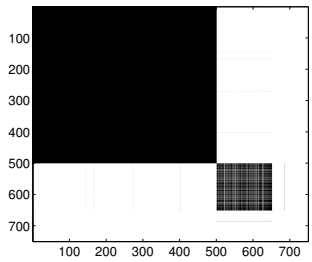


Figure 2. The solution to (CP1) with mid-size clusters.

5. Discussion

An immediate goal is to better understand the "mid-size crisis". Our current results say nothing about clusters that fall in the interval $(\ell_\flat, \ell_\sharp)$. Our numerical experiments confirm that the mid-size phenomenon is real: they are neither completely recovered nor entirely ignored by the optimal $\hat{K}$. obvious pattern.

Our study was mainly theoretical, focusing on the planted partition model. Our experiments focused on confirming the theoretical findings with data generated exactly according to the distribution we could provide provable guarantees for. It is interesting to apply our methods to real applications, particularly big datasets merged from web application and social networks.

Another interesting direction is extending the "peeling strategy" to other high-dimensional learning problems. One intuitive explanation of the small cluster barrier encountered in previous work is *ambiguity* – when viewing from the whole graph, a small cluster is both a low-rank matrix and a sparse one. Only when "zooming in" (after removing big clusters), small clusters patterns emerge. There are other formulations with a similar property. For example, in Xu et al. (2012), the authors propose to decompose a matrix into the sum of a low rank one and a column sparse one to solve an outlier-resistant PCA task. Notice that a column sparse matrix is also low rank. We hope the "peeling strategy" may also help with that problem.

Acknowledgements

The authors would like to thank the anonymous reviewers for helpful comments. N. Ailon acknowledges the support of a Marie Curie International Reintegration Grant PIRG07-GA-2010-268403, and a grant from Technion-Cornell Innovation Institute (TCII). Y. Chen is supported by NSF Grant EECS-1056028 and DTRA grant HDTRA 1-08-0029. H. Xu is supported by the Ministry of Education of Singapore through AcRF Tier-two grant R-265-000-443-112 and NUS startup grant R-265-000-384-133.

References

Ailon, N., Charikar, M., and Newman, A. Aggregating inconsistent information: Ranking and clustering. *J. ACM*, 55(5):23:1–23:27, 2008.

Ailon, N., Begleiter, R., and Ezra, E. Active learning using smooth relative regret approximations with applications. In *COLT*, 2012.

Ames, B. and Vavasis, S. Nuclear norm minimization for the planted clique and biclique problems. *Mathematical Programming*, 129(1):69–89, 2011.

Bansal, N., Blum, A., and Chawla, S. Correlation clustering. *Machine Learning*, 56:89–113, 2004.

Bollobás, B. and Scott, AD. Max cut for random graphs with a planted partition. *Combinatorics, Prob. and Comp.*, 13(4-5):451–474, 2004.

Candès, E., Li, X., Ma, Y., and Wright, J. Robust principal component analysis? *J. ACM*, 58:1–37, 2011.

Carson, T. and Impagliazzo, R. Hill-climbing finds random planted bisections. In *SODA*, 2001.

Chandrasekaran, V., Sanghavi, S., Parrilo, S., and Willsky, A. Rank-sparsity incoherence for matrix decomposition. *SIAM J. on Optimization*, 21(2): 572–596, 2011.

Charikar, Moses, Guruswami, Venkatesan, and Wirth, Anthony. Clustering with qualitative information. *J. Comput. Syst. Sci.*, 71(3):360–383, 2005.

Chaudhuri, K., Chung, F., and Tsiatas, A. Spectral clustering of graphs with general degrees in the extended planted partition model. *COLT*, 2012.

Chen, Y., Sanghavi, S., and Xu, H. Clustering sparse graphs. In *NIPS. Available on arXiv:1210.3335*, 2012.

Condon, A. and Karp, R.M. Algorithms for graph partitioning on the planted partition model. *Random Structures and Algorithms*, 2001.

Demaine, E., Emanuel, D., Fiat, A., and Immorlica, N. Correlation clustering in general weighted graphs. *Theoretical Comp. Sci.*, 2006.

Eriksson, B., Dasarathy, G., Singh, A., and Nowak, R. Active clustering: Robust and efficient hierarchical clustering using adaptively selected similarities. *arXiv:1102.3887*, 2011.

Giesen, J. and Mitsche, D. Reconstructing many partitions using spectral techniques. In *Fundamentals of Computation Theory*, pp. 433–444, 2005.

Giotis, Ioannis and Guruswami, Venkatesan. Correlation clustering with a fixed number of clusters. *Theory of Computing*, 2(1):249–266, 2006.

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

Jalali, A., Chen, Y., Sanghavi, S., and Xu, H. Clustering partially observed graphs via convex optimization. In *ICML. Available on arXiv:1104.4803*, 2011.

Jalali, Ali and Srebro, Nathan. Clustering using max-norm constrained optimization. In *ICML. Available on arXiv:1202.5598*, 2012.

Krishnamurthy, A., Balakrishnan, S., Xu, M., and Singh, A. Efficient active algorithms for hierarchical clustering. *arXiv:1206.4672*, 2012.

Mathieu, C. and Schudy, W. Correlation clustering with noisy input. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 712–728. SIAM, 2010.

McSherry, F. Spectral partitioning of random graphs. In *FOCS*, pp. 529–537, 2001.

Oymak, S. and Hassibi, B. Finding dense clusters via low rank + sparse decomposition. arXiv:1104.5186v1, 2011.

Rohe, K., Chatterjee, S., and Yu, B. Spectral clustering and the high-dimensional stochastic block model. *Ann. of Stat.*, 39:1878–1915, 2011.

Shamir, O. and Tishby, N. Spectral Clustering on a Budget. In *AISTATS*, 2011.

Shamir, R. and Tsur, D. Improved algorithms for the random cluster graph model. *Random Struct. & Alg.*, 31(4):418–449, 2007.

Voevodski, K., Balcan, M., Röglin, H., Teng, S., and Xia, Y. Active clustering of biological sequences. *JMLR*, 13:203–225, 2012.

Xu, H., Caramanis, C., and Sanghavi, S. Robust PCA via outlier pursuit. *IEEE Transactions on Information Theory*, 58(5):3047–3064, 2012.