JMLR: Workshop and Conference Proceedings vol 40:1–22, 2015

Correlation Clustering with Noisy Partial Information

Konstantin Makarychev KOMAKARY@MICROSOFT.COM
Microsoft Research

Yury Makarychev YURY@TTIC.EDU
Toyota Technological Institute at Chicago[*]

Aravindan Vijayaraghavan ARAVINDV@NORTHWESTERN.EDU
Courant Institute, NYU[†]

Abstract

In this paper, we propose and study a semi-random model for the Correlation Clustering problem on arbitrary graphs G. We give two approximation algorithms for Correlation Clustering instances from this model. The first algorithm finds a solution of value $(1+\delta)\,\text{opt-cost} + O_\delta(n\log^3 n)$ with high probability, where opt-cost is the value of the optimal solution (for every $\delta > 0$). The second algorithm finds the ground truth clustering with an arbitrarily small classification error η (under some additional assumptions on the instance).

Keywords: Correlation clustering, semi-random model, polynomial-time approximation scheme

1. Introduction

One of the most commonly used algorithmic tools in data analysis and machine learning is clustering – partitioning a corpus of data into groups based on similarity. The data observed in several application domains – e.g., protein-protein interaction data, links between web pages, and social ties on social networks – carry relational information between pairs of nodes, which can be represented using a graph. Clustering based on relational information can reveal important structural information such as functional groups of proteins (Bader and Hogue, 2003; Girvan and Newman, 2002), communities on web and social networks (Fortunato, 2010; Karrer and Newman, 2011), and can be used for predictive tasks such as link prediction (Taskar et al., 2004).

Correlation clustering tackles this problem of clustering objects when we are given qualitative information about the similarity or dissimilarity between some pairs of these objects. This qualitative information is represented in the form of a graph $G(V, E, c)$ in which edges E are labeled with signs $\{+, -\}$; we denote the set of '$+$' edges by E_+ and the set of '$-$' edges by E_-. Each edge (u, v) in E_+ indicates that u and v are similar, and each edge $(u, v) \in E_-$ indicates that u and v are dissimilar; the cost $c(u, v)$ of the edge shows the amount of similarity or dissimilarity between u and v.[1] In the ideal case, this qualitative information is consistent with the intended ("ground truth") clustering. However, the qualitative information may be noisy due to errors in the observations. Hence, the goal is to find a partition $\mathcal{P}$ of G that minimizes the cost of inconsistent

[†] Supported by NSF CAREER award CCF-1150062 and NSF award IIS-1302662.

[†] Supported by the Simons Collaboration on Algorithms and Geometry.

[1] One can also think of the instance as a graph $G(V, E, c)$ with the edge costs $c : E \to [-1, 1]$. If $c(u, v) > 0$ then $(u, v) \in E_+$, and If $c(u, v) < 0$ then $(u, v) \in E_-$.

edges:

$$\min_{\mathcal{P}} \sum_{(u,v)\in E_+ : \mathcal{P}(u)\neq\mathcal{P}(v)} c(u,v) + \sum_{(u,v)\in E_- : \mathcal{P}(u)=\mathcal{P}(v)} c(u,v),$$

where $\mathcal{P}(u)$ denotes the cluster that contains the vertex u. The objective captures the cost of inconsistent edges – cut edges in E_+ and uncut edges in E_-. (For a partition $\mathcal{P}$, we say that an edge $(u,v) \in E$ is consistent with $\mathcal{P}$ if either $(u,v) \in E_+$ and $\mathcal{P}(u) = \mathcal{P}(v)$ or $(u,v) \in E_-$ and $\mathcal{P}(u) \neq \mathcal{P}(v)$.)

Note that the underlying graph $G(V,E)$ can be reasonably sparse; this is desirable since collecting pairwise information can be expensive. One important feature of correlation clustering is that it, unlike most other clustering problems, allows us not to specify the number of clusters. Hence, it is particularly useful when we have no prior knowledge of the number of clusters that the data divides into.

Correlation clustering also comes up naturally in MAP inference in graphical models and structured prediction tasks for such tasks as image segmentation, parts-of-speech tagging and dependency parsing in natural language processing (Nowozin and Lampert, 2010; Smith, 2011). In structured prediction, we are given some observations as input (e.g., image data, sentences), and the goal is to predict a labeling $\mathbf{x} \in \mathcal{X}$ that encodes the high-level information that we would like to infer. For instance, in image segmentation, the variables $x \in \{0,1\}^n$ indicate whether each pixel is in the foreground or background. This is naturally modeled as a Correlation Clustering instance on the set of pixels (with 2 clusters), where edges connect adjacent pixels, and the costs (with signs) are set based on the similarity or dissimilarity of the corresponding pixels in the given image. The clusters in these inference problems then consist of the sets of variables that receive the same assignment in the MAP solution. Correlation clustering is also used in the context of consensus clustering and agnostic learning.

Correlation clustering was introduced in (Bansal et al., 2004), and implicitly in (Ben-Dor et al., 1999) as 'Cluster Editing'. The problem is APX-hard even on complete graphs[2] (when we are given the similarity information for every pair of objects) (Charikar et al., 2005). The state-of-the-art approximation algorithm (Charikar et al., 2005; Demaine et al., 2006) achieves an $O(\log n)$ approximation for minimizing disagreements in the worst-case. Furthermore, there is a gap-preserving reduction from the classic Minimum Multicut problem (Charikar et al., 2005; Demaine et al., 2006), for which the current state-of-the-art algorithm gives a $\Theta(\log n)$ factor approximation (Garg et al., 1993). The complementary objective of maximizing agreements is easier from the approximability standpoint, and a 0.766 factor approximation is known (Charikar et al., 2005; Swamy, 2004). For the special case of complete graphs (with unit costs on edges), small constant factor approximations have been obtained in a series of works (Bansal et al., 2004; Ailon et al., 2008; Chawla et al., 2014). Instances of Correlation Clustering on complete graphs that satisfy the notion of approximation stability were considered in (Balcan and Braverman, 2009). To summarize, despite our best efforts, we only know logarithmic factor approximation algorithms for Correlation Clustering; moreover, we cannot get a constant factor approximation for worst-case instances if the Unique Games Conjecture is true.

However, our primary interest in solving Correlation Clustering comes from its numerous applications, and the instances that we encounter in these applications are not worst-case instances.

[2] This rules out $(1+\epsilon)$ factor approximations for some small constant $\epsilon > 0$.

This motivates the study of the average-case complexity of the problem and raises the following question:

> Can we design algorithms with better provable guarantees for realistic average-case models of Correlation Clustering?

Several natural average-case models of Correlation Clustering have been studied previously. Ben-Dor et al. (1999) consider a model in which we start with a ground-truth clustering – an arbitrary partitioning of the vertices – of a complete graph. Initially, edges inside clusters of the ground truth solution are labeled '+' and edges between clusters are labeled '-'. We flip the label of each edge (change '$+$' to '$-$' and '$-$' to '$+$') with probability ε independently at random and obtain a Correlation Clustering instance (the flipped edges model the noisy observations) . In fact, this average-case model was also studied in the work (Bansal et al., 2004) that introduced the problem of Correlation Clustering. Mathieu and Schudy consider a generalization of this model where there is an adversary: for each edge, we keep the initial label with probability $(1-\varepsilon)$, and we let the adversary decide whether to flip the edge label or not with probability ε. The major drawback of these models is that they only consider the case of complete graphs, i.e. they require that the Correlation Clustering instance contains similarity information for *every* pair of nodes. Chen et al. extended the model of (Ben-Dor et al., 1999) from complete graphs to sparser Erdos–Renyi random graphs. In their model, the underlying unlabeled graph $G(V,E)$ comes from an Erdös–Renyi random graph (of edge probability p), and as in (Ben-Dor et al., 1999), the label of each edge is set (independently) to be consistent with the ground truth clustering with probability $1-\varepsilon$ and inconsistent with probability ε.

While these average-case models are natural, they are unrealistic in practice since most real-world graphs are neither dense nor captured by Erdös–Renyi distributions. For instance, real-world graphs in community detection have many structural properties (presence of large cliques, large clustering coefficients, heavy-tailed degree distribution) that are not exhibited by graphs that are generated by Erdös–Renyi models (Newman et al., 2006; Kumar et al., 1999). Graphs that come up in computer vision applications are sparse with grid-like structure (Yarkony et al., 2012). Further, these models assume that every pair of vertices have the same amount of similarity or dissimilarity (all costs are unit). Our semi-random model tries to address these issues by assuming very little about the observations – the underlying unlabeled graph $G(V,E)$ – and allowing non-uniform costs.

1.1. Our Semi-random Model

In this paper, we propose and study a new semi-random model for generating general instances of Correlation Clustering, which we believe captures many properties of real world instances. It generalizes the model of Mathieu and Schudy (2010) to arbitrary graphs $G(V,E,c)$ with costs. A semi-random instance $\{G(V,E,c),(E_+,E_-)\}$ is generated as follows:

1. The adversary chooses an undirected graph $G(V,E,c)$ and a partition $\mathcal{P}^*$ of the vertex set V (referred to as the planted clustering or ground truth clustering).

2. Every edge is E is included in set E_R independently with probability ε.

3. Every edge $(u,v)\in E\setminus E_R$ with u and v in the same cluster of $\mathcal{P}^*$ is included in E_+, and every edge $(u,v)\in E\setminus E_R$, with u and v in different clusters of $\mathcal{P}^*$ is included in E_-.

4. The adversary adds every edge from E_R either to E_+ or to E_- (but not to both sets).

This model can be further generalized to an adaptive semi-random model as described in Section 3.1.

1.2. Our Results

We develop two algorithms for semi-random instances of Correlation Clustering. The first algorithm gives a polynomial-time approximation scheme (PTAS) for instances from our semi-random model. The second algorithm recovers the planted partition with a small classification error η.

Theorem 1 *For every $\delta > 0$, there is a polynomial-time algorithm that given a semi-random instance $\{G(V,E,c),(E_+,E_-)\}$ of Correlation Clustering (with noise probability $\varepsilon < 1/4$), finds a clustering that has disagreement cost $(1+\delta)\,\text{opt-cost} + O((1-2\varepsilon)^{-4}\delta^{-3} n \log^3 n)$ w.h.p. over the randomness in the instance, where opt-cost is the cost of disagreements of the optimal solution for the instance.*

The approximation additive term is much smaller than the cost of the planted solution if the average degree $\Delta \gg \varepsilon^{-1}\text{polylog}\, n$. Note that we compare the performance of our algorithm with the cost of the *optimal* solution. Further, these guarantees hold even in a more general adaptive semi-random model that is described in Section 3.1.

The above result gives a good approximation guarantee with respect to the objective. *But what about recovering the ground truth clustering?* Our semi-random model is too general to allow recovery. For instance, there could be large disconnected pieces inside some clusters of G, or there could be no edges between some clusters — in both cases, recovery is statistically impossible. Hence, we need some additional conditions for approximate recovery in our model, that guarantee at the very least that the ground truth clustering is uniquely optimal (in a robust sense).

Our first assumption is that there is mild *expansion inside clusters* — this connectivity assumption prevents large pieces inside clusters that are almost disconnected, which might get separated in an almost optimal clustering. The second and third assumptions are that there are enough edges from vertices in one cluster to other clusters, to prevent these clusters (or parts of them) from coalescing in near-optimal clusterings. Finally, we assume approximate regularity in degrees inside clusters (the degrees of all vertices are approximately equal up to a factor of $\alpha \geq 1$), since it is hard to correctly classify vertices with very few edges incident on them. These assumptions are described formally in Section 5. We refer to them as Assumptions 10. We now informally describe the algorithmic guarantees for approximate recovery:

Theorem 2 *There exists a polynomial-time algorithm that given a semi-random instance $\mathcal{I} = \{G=(V,E,c),(E_+,E_-)\}$ satisfying mild expansion inside clusters, regularity and inter-cluster density conditions with parameters α, β and λ_{gap} (see Assumptions 10 for details) finds a partition $\mathcal{P}$ with classification error at most 4η w.h.p. over the randomness in the instance, where*

$$\eta = \frac{C_2}{(1-2\varepsilon)}\left(\frac{n\log n}{c(E)}\right)^{1/12} \cdot \left(\frac{\alpha^3}{\beta\lambda_{gap}}\right)^{1/2}. \tag{1}$$

Our algorithm outputs a clustering such that only $O(\eta n)$ vertices are misclassified (up to a renaming of the clusters). We note that the expansion and regularity assumptions are satisfied by Erdös–Renyi graphs: for instance, such random graphs have strong expansion both inside and between clusters ($\lambda_{\text{gap}} = 1 - o(1)$) and have strong concentration of degrees. Our assumptions for recovery are soft: if there is bad expansion inside clusters (λ_{gap} is small), or if there are not sufficient edges between vertices in different clusters, we just need more observations (edges) to approximately recover the clusters. For instance, when α is polylogarithmic in n, β and λ_{gap} are inverse polylogarithmic in n, Theorem 2 only requires that the average degree of the graph (w.r.t. edge costs $c(e)$) is polylogarithmic in n.

1.3. Related Work on Semi-random Models

Over the last two decades, there has been extensive research on average-case complexity of many important combinatorial optimization problems. Semi-random instances typically allow 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. Semi-random models have also been studied for graph partitioning problems (Feige and Kilian, 1998; Chen et al., 2012; Makarychev et al., 2012, 2014), Independent Set (Feige and Kilian, 1998), Maximum Clique (Feige and Krauthgamer, 2000), Unique Games (Kolla et al., 2011), and other problems. Most related to our work, both in the nature of the model and in the techniques used, is a recent result of (Makarychev et al., 2013) on semi-random instances of Minimum Feedback Arc Set. While the techniques used in both papers are conceptually similar, the semidefinite (SDP) relaxation for Correlation Clustering that we use in this paper is very different from the SDP relaxation for Minimum Feedback Arc Set used in (Makarychev et al., 2013). Further, we get a true $1+\delta$ approximation scheme (with an extra additive approximation term). This is in contrast to previous semi-random model results (Makarychev et al., 2012, 2013), which compare the cost of the solution that the algorithm finds to the cost of the planted solution. Moreover, this work gives not only a PTAS for the problem, but also a simple algorithm for recovery the ground truth solution.

Mathieu and Schudy recently considered a semi-random model for Correlation Clustering on complete graphs with unit edge costs. Later, Elsner and Schudy conducted an empirical evaluation of algorithms for the complete graph setting. Chen et al. (2014) extended the average-case model of Correlation Clustering to sparser Erdös–Renyi graphs. Very recently, Globerson et al. (2014) considered a semi-random model for Correlation Clustering for recovery in grid graphs and planar graphs, and gave conditions for approximate recovery in terms of an expansion-related condition.

Comparison of Results. The two works that are most similar in the nature of guarantees are (Mathieu and Schudy, 2010) and (Chen et al., 2014). Mathieu and Schudy designed an algorithm based on semidefinite programming (SDP relaxations with ℓ_2^2-triangle inequality constraints) for their semi-random model on complete graphs. It finds a clustering of cost at most $1 + O(n^{-1/6})$ times the cost of the optimal clustering (as long as $\varepsilon \leq 1/2 - O(n^{-1/3})$) and manages to approximately recover the ground truth solution (when the clusters have size at least $\sqrt{n}$). However, this algorithm only works on complete graphs and assumes unit edge costs. Chen et al. studied the problem on sparser graphs from the Erdös–Renyi distribution, and using weaker convex relaxations gave an algorithm that recovers the ground-truth when $p \geq k^2 \log^{O(1)} n/n$. In the case of Erdös–Renyi graphs, our algorithms obtain similar guarantees for smaller values of k (the implicit dependence on k is a worse polynomial than in (Chen et al., 2014), however). The main advantage of our algorithms is that they work for more general graphs G: the first algorithm requires only that the average degree of G is some poly-log of n, while the second algorithm requires additionally that the graph has a mild expansion and regularity; its performance depends softly on the expansion and regularity parameters of the graph.

Empirical Results. We describe our empirical results in Appendix A.

2. Overview of the Algorithms and Structural Insights

SDP relaxation. We use a simple SDP relaxation for the problem (Swamy, 2004). For every vertex u, we have a unit vector $\bar{u}$. For two vertices u and v, we interpret the inner product $\langle \bar{u}, \bar{v} \rangle \in [0,1]$ as the indicator of the event: u and v lie in the same partition. The SDP is given below:

$$\min_{\mathcal{P}} \sum_{(u,v)\in E_+} c(u,v)(1 - \langle \bar{u}, \bar{v} \rangle) + \sum_{(u,v)\in E_-} c(u,v)\langle \bar{u}, \bar{v} \rangle.$$

subject to: for all $u, v \in V$,

$$\begin{aligned} \langle \bar{u}, \bar{v} \rangle &\in [0,1]; \\ \|\bar{u}\|^2 &= 1. \end{aligned}$$

The intended vector (SDP) solution has one co-ordinate for every cluster of the clustering $\mathcal{P}$: the vector $\bar{u}$ for vertex u has 1 in the co-ordinate corresponding to $\mathcal{P}(u)$ and 0 otherwise. Hence this SDP is a valid relaxation. We note that this relaxation is weaker than the SDP used in (Mathieu and Schudy, 2010) because it does not have ℓ_2^2-triangle inequalities constraints. Hence, this semidefinite program is more scalable, and it is efficiently solvable for instances with a few thousand nodes.

Approximation Algorithm (PTAS). We now describe the algorithm that gives a PTAS. Fix a parameter $\delta = o(1) \in (0, 1/2)$. To simplify the notation, denote by $f(u,v)$ (for $(u,v) \in E$) the SDP value of the edge (without cost):

$$f(u,v) = 1 - \langle \bar{u}, \bar{v} \rangle \text{ if } (u,v) \in E_+, \text{ and } f(u,v) = \langle \bar{u}, \bar{v} \rangle, \text{ otherwise.} \tag{2}$$

Our PTAS is based on a surprising structural result about near-integrality of the SDP relaxation on the edges of the graph (see Theorem 3 for a formal statement).

Informal Structural Theorem. *In any feasible SDP solution of cost at most OPT, the SDP value of edge $f(u,v) \geq 1-\delta$ for a $1-o_\delta(1/\log n)$ fraction of the inconsistent edges $(u,v) \in E(G)$.*

Hence, the structural result suggests that by removing all edges that contribute at least $(1-\delta)$ to the objective, the remaining instance has a solution of very small cost. We then run the $O(\log n)$ worst-case approximation algorithm of (Charikar et al., 2005) or (Demaine et al., 2006) on the remaining graph to obtain a PTAS overall.

Recovery. The algorithm outlined above finds a solution of near optimal cost. Under additional assumptions, we show that we can in fact design a very simple greedy rounding scheme that can also efficiently recover the ground truth clustering approximately.

The structural theorem above shows that the SDP vectors are highly correlated for pairs of adjacent vertices. Under the additional conditions, we show that the vectors are in fact globally clustered according to the ground truth clustering:

Informal Structural Theorem. *When the semi-random instance $\{G = (V, E, c), E_+, E_-\}$ satisfies Assumption 10, we have w.h.p. that: for a $(1 - O(\eta))$ fraction of the clusters P_i^* we can choose centers $u_i \in P_i^*$ and define cores $\mathrm{core}(P_i^*) = \{v \in P_i^* : \|\bar{v} - \bar{u}_i\| \leq 1/10\} \subseteq P_i^*$ (balls of radius $1/10$ around centers $\bar{u}_i$) such that $\mathrm{core}(P_i^*) \geq (1-\eta)|P_i^*|$ (the core of P_i^* contains all but an η fraction of vertices of P_i^*) and centers u_i are mutually separated by a distance of at least $4/5$.*

The recovery algorithm is a greedy algorithm that finds heavy regions – sets of vectors that are clumped together – and puts them into clusters.

Recovery Algorithm

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

Output: partition $P_1, \dots, P_t$ of V (for some t).

```
$i = 1$, $\rho_{\text{core}} = 0.1$
Define an auxiliary graph $G_{aux} = (V, E_{aux})$ with $E_{aux} = \{(u,v) : \|\bar{u} - \bar{v}\| \leq \rho_{\text{core}}\}$
while $V \setminus (P_1 \cup \dots P_{i-1}) \neq \varnothing$
  Let $u$ be the vertex of maximum degree in $G_{aux}[V \setminus (P_1 \cup \dots P_{i-1})]$.
  Let $P_i = \{v \notin P_1 \cup \dots \cup P_{i-1} : (u,v) \in E_{aux}\}$    // note that $P_i$ contains $u$
  $i = i + 1$
return clusters $P_1, \dots, P_{i-1}$.
```

This structural result about the global clustering and near integrality of the SDP vectors is consistent with empirical evidence. While our algorithm succeeds when the SDP is tight (as in (Chen et al., 2014)), the analysis of our algorithm also shows how to deal with nearly integral solutions, in which most inner products $\langle \bar{u}, \bar{v} \rangle$ are only close to 0 or 1 (but may not be tight). We believe that many instances arising in practice have SDP solutions that are nearly integral, but not integral. Hence, we believe that in practice, our algorithm will work better than previously known algorithms.

3. Polynomial-time Approximation Scheme

In this section, we present the analysis of our polynomial-time approximation scheme for correlation clustering, which we presented in Section 2. The PTAS works in a very general Adaptive Model, which we describe first.

3.1. Adaptive Model

We study a more general "adaptive" semi-random model. A semi-random instance is generated as follows. We start with a graph $G_0(V, \varnothing)$ on n vertices with no edges and a partition $\mathcal{P}^*$ of V into disjoint sets, which we call the planted partition. The adversary adds edges one by one. We denote the edge chosen at step t by e_t and its cost $c(e_t) \in [0,1]$. After the adversary adds an edge e_t to the set of edges, the nature flips a coin and with probability ε adds e to the set of random edges E_R. The next edge e_{t+1} chosen by the adversary may depend on whether e_t belongs to E_R or not. The adversary stops the semi-random process at a stopping time T. Thus, we obtain a graph $G^*(V, \{e_1, \dots, e_T\}, c)$ and a set of random edges E_R. We denote the set of all edges by $E^* = \{e_1, \dots, e_T\}$. The adversary may remove some edges belonging to E_R from the set E^*. Denote the set of the remaining edges by E. Note that $E^* \setminus E_R \subset E \subset E^*$.

Once the graph $G(V, E)$ and the set E_R are generated, we perform steps 3 and 4 from the basic semi-random model for the graph $G(V, E)$ and random set of edges $E_R \cap E$ (as described in Section 1.1). We obtain a semi-random instance. This is the instance the algorithm gets. Of course, the algorithm does not get the set of random edges E_R. Note that the cost of the planted solution $\mathcal{P}^*$ is at most the cost of the edges $E_R \cap E$ i.e. $c\big(E_R \cap E\big)$, since all edges in $E \setminus E_R$ are consistent with $\mathcal{P}^*$.

This Adaptive Model is more general than the Basic Semi-random model we introduced earlier. The basic semi-random model corresponds to the case when the whole set of edges E^* is fixed in advance independent of the random choices made in E_R, and $E = E^*$. However, in the adaptive model the edge e_t can be chosen based on which of the edges $e_1, \dots e_{t-1}$ belong to E_R. For instance, the adversary can choose edge e_t from the portion of the graph where many of the previously chosen edges belong to E_R.

3.2. Analysis of the Algorithm

Now we analyze the algorithm presented in Section 2. We need to bound the number of edges removed at the first step (that is, edges (u,v) with $f(u,v) > 1-\delta$) and the number of edges cut by the $O(\log n)$ approximation algorithm at the second step. The SDP contribution of every edge (u,v) removed at the first step is at least $c(u,v)(1-\delta)$. Thus the cost of edges removed at the first step is bounded by $SDP/(1-\delta) \leq (1+2\delta)OPT$. To bound the cost of the solution produced by the approximation algorithm at the second step, we need to bound the cost of the optimal solution for the remaining instance i.e., the instance with the set of edges $\{(u,v) \in E : f(u,v) \leq 1-\delta\}$.

For any subset of edges $F \subset E$, let $c(F)$ represent the cost of the edges in F i.e. $c(F) = \sum_{e \in F} c(e)$. Denote E_+^* and E_-^*: $E_+^* = \{(u,v) : \mathcal{P}^*(u) = \mathcal{P}^*(v)\}$ and $E_-^* = \{(u,v) : \mathcal{P}^*(u) \neq \mathcal{P}^*(v)\}$. Now define a function $f^*(u,v)$, which slightly differs from $f(u,v)$. For all $(u,v) \in E$,

$$f^*(u,v) = \begin{cases} 1 - \langle \bar{u}, \bar{v} \rangle, & \text{if } \mathcal{P}^*(u) = \mathcal{P}^*(v); \\ \langle \bar{u}, \bar{v} \rangle, & \text{if } \mathcal{P}^*(u) \neq \mathcal{P}^*(v). \end{cases} \tag{3}$$

Here, $\mathcal{P}^*$ is the planted partition. Note that $\mathcal{P}^*$ and $f^*(u,v)$ are not known to the algorithm. Observe that $f(u,v) = f^*(u,v)$ if the edge (u,v) is consistent with the planted partition $\mathcal{P}^*$, and $f(u,v) = 1 - f^*(u,v)$ otherwise. Our goal is to show that the algorithm removes all but very few edges inconsistent with $\mathcal{P}^*$, i.e., edges (u,v) with $f(u,v) = 1 - f^*(u,v)$. We prove the following theorem in Section C. The proof relies on Theorem 9 presented in Section 4.

Theorem 3 *Let $\{G = (V,E,c), (E_+, E_-)\}$ be a semi-random instance of the correlation clustering problem. Let E_R be the set of random edges, and $\mathcal{P}^*$ be the planted partition. Denote by $Q \subset E_R$ the set of random edges not consistent with $\mathcal{P}^*$. Then, for some universal constant C and every $\delta, \gamma > 0$, and for $\Lambda = C(1-2\varepsilon)^{-2}\gamma^{-2}\delta^{-3} n \log n$,*

$$\Pr\left[\sum_{(u,v) \in Q : f(u,v) \leq 1-\delta} c(u,v) \geq \Lambda + \frac{6\gamma}{1-2\varepsilon} c(Q)\right] = o(1).$$

where f corresponds to any feasible SDP solution of cost at most OPT.

Remark 4 *In the statement of Theorem 3, $c(Q)$ is the value of the solution given by the planted solution $\mathcal{P}^*$. If $OPT = c(Q)$, then the planted solution $\mathcal{P}^*$ is indeed an optimal clustering. The function $f(u,v)$ in the theorem that corresponds to the SDP contribution of edge (u,v) could come from any (not necessarily optimal) SDP solution of cost at most OPT. This will be useful in Lemma 5.*

Let $D = O(\log n)$ be the approximation algorithms of Charikar et al. (2005) or Demaine et al. (2006). We apply Theorem 3 with $\gamma = \frac{\delta(1-2\varepsilon)}{6D}$. The cost of edges in $\{(u,v) \in Q : f(u,v) \leq 1-\delta\}$ is bounded by

$$\Lambda + \frac{6\gamma}{1-2\varepsilon} c(Q) \leq \Lambda + D^{-1}\delta\, c(Q), \tag{4}$$

w.h.p., where $\Lambda = O((1-2\varepsilon)^{-4}\delta^{-3}n\log^3 n)$. Thus, after removing edges with $f(u,v) \geq (1-\delta)$, the cost of the optimal solution is at most (4) w.h.p. The approximation algorithm finds a solution of cost at most D times (4). Thus, the total cost of the solution returned by the algorithm is at most

$$\begin{aligned}(1+2\delta)OPT + D \times (\Lambda + D^{-1}\delta \cdot c(Q)) &= (1+3\delta)c(Q) + D\Lambda \\ &= (1+3\delta)c(Q) + O((1-2\varepsilon)^{-4}\delta^{-3}n\log^3 n).\end{aligned}$$

The above argument shows that the solution has small cost compared to the cost of the planted solution $\mathcal{P}^*$. We can in fact use Theorem 3 to give a true approximation i.e., compared to the cost of the optimal solution OPT. This follows from the following lower bound on OPT in terms of $c(Q)$ for semi-random instances (which we prove in Section D).

Lemma 5 *In the notation of Theorem 3, with probability* $1 - o(1)$,

$$c(Q) \leq (1+2\delta)OPT + O\left((1-2\varepsilon)^{-4}\delta^{-3}n\log^3 n\right).$$

Proof [Proof of Theorem 1] From Theorem 3, we get the total cost of the solution is bounded by

$$\begin{aligned}(1+2\delta)OPT + D \times (\Lambda + D^{-1}\delta \cdot c(Q)) &= (1+2\delta)OPT + D \times \Lambda + \frac{\delta}{1-\delta/D} \cdot (OPT + \Lambda) \\ &\leq (1+4\delta)OPT + 2D\Lambda \\ &= (1+4\delta)OPT + O((1-2\varepsilon)^{-4}\delta^{-3}n\log^3 n).\end{aligned}$$

This finishes the analysis of the algorithm. ■

4. Betting with Stakes Depending on the Outcome

We first informally describe the theorem we prove in this section. Consider the following game. Assume that we are given a set of vectors $\mathcal{W} \subset [0,1]^m$. At every step t, the player (adversary) picks an arbitrary not yet chosen coordinate $e_t \in \{1,\dots,m\}$, and the casino (nature) flips a coin such that with probability $\varepsilon < 1/2$, the player wins, and with probability $(1-\varepsilon) > 1/2$, the player looses. In the former case, we set $X_t = 1$; and in the latter case we set $X_t = -1$. At some point $T \leq m$ the player stops the game. At that point, he picks a vector $w \in \mathcal{W}$ and declares that at time t his stake was $w(e_t)$ dollars. We stress that the vector w may depend on the outcomes X_t. Then, the player's payoff equals

$$\sum_{t=1}^{T} X_t w(e_t).$$

If the player could pick an arbitrary w after the outcomes X_t are revealed, then clearly he could get a significant payoff by letting $w(e_t) = 1$, for $X_t = 1$, and $w(e_t) = 0$, otherwise. However, we assume that the set $\mathcal{W}$ of possible bets is relatively small. Then, we show that with high probability the payoff is negative unless the total amount of bets $\sum_t w(e_t)$ is very small. The precise statement of the theorem (see below) is slightly more technical.

The main idea of the proof is that for any $w \in \mathcal{W}$ fixed in advance, the player is expected to loose with high probability, since the coin is not fair ($\varepsilon < 1/2$), and thus the casino has an advantage. In fact, the probability that the player wins is exponentially small if the coordinates of w are sufficiently large. Now we union bound over all w's in $\mathcal{W}$ and conclude that with high probability for every $w \in \mathcal{W}$, the player's payoff is negative.

When we apply this theorem to a semi-random instance of Correlation Clustering (with unit costs i.e. $c(e_t) = 1$), the stakes are defined by the solution of the SDP: for an edge $e_t = (u, v)$, $w(e_t) = f^*(u, v)$. Loosely speaking, we show that since the SDP value is at most OPT, the game is profitable for the adversary. This implies that most stakes $f^*(u, v)$ are close to 0. Now, if an edge (u, v) is consistent with the planted partition $\mathcal{P}^*$, then $f(u, v) = f^*(u, v) \approx 0$, and hence we do not remove this edge. On the other hand, if the edge is not consistent with the planted partition, then $f(u, v) = 1 - f^*(u, v) \approx 1$, hence we remove the edge.

Lemma 6 *Let $\mathcal{W} \subset [0,1]^m$ be a set of vectors. Consider a stochastic process $(e_1, X_1, c_1), \dots, (e_T, X_T, c_T)$. Each $e_t \in \{1, \dots, m\} \setminus \{e_1, \dots, e_{t-1}\}$, $X_t \in \{\pm 1\}$, $c_t \in [0, 1]$. Let $\mathcal{F}_t$ be the filtration generated by the random variables $(e_1, X_1, c_1), \dots, (e_t, X_t, c_t)$, and $\mathcal{F}'_t$ be the filtration generated by the random variables $(e_1, X_1, c_1), \dots, (e_t, X_t, c_t)$ and (e_{t+1}, c_{t+1}). The random variable $T \in \{1, \dots, m\}$ is a stopping time w.r.t. $\mathcal{F}_t$. Each X_t is a Bernoulli random variable independent of $\mathcal{F}'_{t-1}$.*

$$X_t = \begin{cases} 1, & \text{with probability } \varepsilon; \\ -1, & \text{with probability } 1 - \varepsilon; \end{cases}$$

where $\varepsilon < 1/2$. Then, for all $\Lambda > 3(1 - 2\varepsilon)^{-2}$,

$$\Pr\Big(\exists w \in \mathcal{W} \text{ s.t. } \sum_{t=1}^{T} X_t w(e_t) c_t + \frac{1 - 2\varepsilon}{2} \sum_{t=1}^{T} w(e_t) c_t \geq 0 \text{ and } \sum_{t=1}^{T} w(e_t) c_t \geq \Lambda\Big) \leq$$
$$\leq 2|\mathcal{W}| e^{-1/5 (1-2\varepsilon)^2 \Lambda}. \quad (5)$$

We prove this lemma in Section D. We now slightly generalize this theorem. In our application, the set of all possible stakes can be infinite, however, we know that there is a relatively small epsilon net for it.

Definition 7 *We say that a set $\mathcal{W} \subset \mathbb{R}^m$ is a γ–net for a set $\mathcal{Z} \subset \mathbb{R}^m$ in the ℓ_∞ norm, if for every $z \in \mathcal{Z}$, there exists $w \in \mathcal{W}$ such that $\|z - w\|_\infty \equiv \max_i\{|z(i) - w(i)|\} \leq \gamma$.*

Remark 8 *If $\mathcal{W}$ is a γ–net for $\mathcal{Z} \subset [0,1]^m$, then there exists $\mathcal{W}' \subset [0,1]^m$ of the same size as $\mathcal{W}$ ($|\mathcal{W}'| = |\mathcal{W}|$), such that for every $z \in \mathcal{Z}$, there exists $w' \in \mathcal{W}'$ satisfying $w'(i) \leq z(i) \leq w'(i) + 2\gamma$ for all i. To obtain $\mathcal{W}'$ we simply subtract $\min(\gamma, w(i))$ from each coordinate of w and then truncate each $w'(i)$ at the threshold of 1.*

We prove the following theorem in Section D.

Theorem 9 *Consider a stochastic process $(e_1, X_1, c_1), \dots, (e_T, X_T, c_T)$ such that each $e_t \in \{1, \dots, m\} \setminus \{e_1, \dots, e_{t-1}\}$, $X_t \in \{\pm 1\}$ and $c_t \in [0, 1]$. Let $\mathcal{F}_t$ be the filtration generated by the random variables $(e_1, X_1, c_1), \dots, (e_t, X_t, c_t)$, and $\mathcal{F}'_t$ be the filtration generated by the random variables $(e_1, X_1, c_1), \dots, (e_t, X_t, c_t)$ and (e_{t+1}, c_{t+1}). The random variable $T \in \{1, \dots, m\}$ is a stopping time w.r.t. $\mathcal{F}_t$. Each X_t is a Bernoulli random variable independent of $\mathcal{F}'_{t-1}$.*

$$X_t = \begin{cases} 1, & \text{with probability } \varepsilon; \\ -1, & \text{with probability } 1 - \varepsilon; \end{cases}$$

where $\varepsilon < 1/2$. Let $\mathcal{Z} \subset [0,1]^m$ be a set of vectors having a γ–net in the L_∞ norm of size N. Define two random sets depending on $\{X_t\}$:

$$Q_+ = \{t : X_t = 1\} \text{ and } Q_- = \{t : X_t = -1\}.$$

Then, for all $\Lambda > 3(1-2\varepsilon)^2$, we have

$$\Pr\Big(\exists z \in \mathcal{Z},\ Q_\oplus \subset Q_+ \text{ s.t.} \sum_{t \in Q_\oplus \cup Q_-} X_t z(e_t) c_t \geq 0 \\ \text{and} \sum_{t \in Q_\oplus} z(e_t) c_t \geq \Lambda + \frac{6\gamma}{1-2\varepsilon} \sum_{t \in Q_\oplus} c_t\Big) \leq 2Ne^{-1/5(1-2\varepsilon)^2\Lambda}. \quad (6)$$

5. Recovery Algorithm

In this section, we prove Theorem 2 that shows that under some additional assumptions on the graph G and partition $\mathcal{P}^*$, we can recover the planted partition $\mathcal{P}^*$ with an arbitrarily small classification error η. The recovery algorithm is a very fast and very simple greedy algorithm (presented in Section 2).

Assumptions 10 *Consider a semi-random instance $\mathcal{I} = \{G = (V,E,c), (E_+, E_-)\}$. Let $\mathcal{P}^*$ be the planted partition. Denote the clusters of G w.r.t clustering $\mathcal{P}^*$ by $P_1^*, \dots, P_k^*$. Let $\beta = c(E_+^*)/c(E)$ (note that E_+^* is the set of edges that lie within clusters), $\beta_{ij} = c(\{(u,v) : u \in P_i^*, v \in P_j^*\})/c(E)$ (here, $\{(u,v) : u \in P_i^*, v \in P_j^*\}$ is the set of edges between clusters P_i^* and P_j^*). Assume that the instance $\mathcal{I}$ satisfies the following conditions with a parameter $\alpha \geq 1$.*

- ***Cluster Expansion.*** *All induced graphs $G[P_i^*]$ are spectral expanders with spectral expansion at least λ_{gap}; that is, the second smallest eigenvalue of the normalized Laplacian of $G[P_i^*]$ is at least λ_{gap}.*
- ***Intercluster Density.*** *For some sufficiently large constant C_1, $\beta_{ij} > \frac{C_1}{(1-2\varepsilon)^2}\left(\frac{n \log n}{c(E)}\right)^{1/6}$.*
- ***Approximate Intercluster Regularity.*** *The graph formed by the edges between every two clusters P_i^* and P_j^* is approximately regular with respect to the cost function c: for every $u', u'' \in P_i^*$ we have $c(\{(u',v) : v \in P_j^*\}) \leq \alpha c(\{(u'',v) : v \in P_j^*\})$.*
- ***Approximate Cluster Regularity.*** *All induced graphs $G[P_i^*]$ are approximately regular with the same degree w.r.t to the cost function c. That is, for some number c_0, every cluster P_i^*, and every vertex $u \in P_i^*$, $c_0 \leq c(\{(u,v) \in E : v \in P_i^*\}) \leq \alpha c_0$.*

Definition 11 *Let $\mathcal{I} = \{G = (V,E,c), (E_+, E_-)\}$ be a semi-random instance of correlation clustering, $\mathcal{P}^*$ be the planted partition, and $P_1^*, \dots, P_k^*$ be the planted clusters. We say that a partition $\mathcal{P}$ of V into clusters $P_1, \dots, P_t$ has an η classification error if there is a partial matching between clusters $P_1^*, \dots, P_k^*$ and clusters $P_1, \dots, P_t$ such that*

$$\sum_{P_i^* \text{ is matched with } P_j} |P_i^* \cap P_j| \geq (1-\eta)|V|.$$

Theorem 2 relies on the following theorem that describes the structure of optimal SDP solutions to semi-random instances of correlation clustering that satisfy conditions in Assumption 10.

Theorem 12 *Assume that a semi-random instance $\mathcal{I} = \{G = (V, E, c), (E_+, E_-)\}$ satisfies Assumptions 10. Let $\{\bar{u}\}$ be the optimal SDP solution to $\mathcal{I}$. With probability $1 - o(1)$, there exist a subset of clusters $\mathcal{C} \subset \{P_1^*, \dots, P_k^*\}$ and a vertex u_i in each cluster P_i^* satisfying the following properties. Let $\rho_{\text{core}} = 1/10$ and $\rho_{\text{inter}} = 4/5$. Let $\operatorname{core}(P_i^*) = \{v \in P_i^* : \|\bar{v} - \bar{u}_i\| \leq \rho_{\text{core}}\}$ for $P_i^* \in \mathcal{C}$, then*

1. $|\cup_{P_i^* \in \mathcal{C}} P_i^*| \geq (1 - \eta)|V|$.
2. $|\operatorname{core}(P_i^*)| \geq (1 - \eta)|P_i^*|$.
3. *In particular,* $\sum_{P_i^* \in \mathcal{C}} |P_i^*| \geq (1 - \eta)^2 |V|$.
4. $\|\bar{u}_i - \bar{u}_j\| \geq \rho_{\text{inter}}$ *for every two distinct clusters* $P_i^*, P_j^* \in \mathcal{C}$.

We prove Theorem 12 in Section E.

Proof [Proof of Theorem 2] Let $P_1, \dots, P_t$ be the clusters found by the recovery algorithm (presented in Section 2). Consider a cluster P_i. Let u be the vertex we choose at iteration i of the while–loop, during the execution of the recovery algorithm. If P_i intersects a core $\operatorname{core}(P_j^*)$ of a cluster P_j^* then $\|u - u_j\| \leq 2\rho_{\text{core}}$. Note that P_i cannot intersect cores $\operatorname{core}(P_{j'}^*)$ and $\operatorname{core}(P_{j''}^*)$ of two distinct clusters $P_{j'}^*$ and $P_{j''}^*$, since $\|u - u_{j'}\| + \|u - u_{j''}\| \geq \|u_{j'} - u_{j''}\| \geq \rho_{\text{inter}} > 4\rho_{\text{core}}$. Thus each cluster P_i intersects the core of at most one cluster P_j^*.

We match every cluster $P_j^* \in \mathcal{C}$ to the first cluster P_i that intersects $\operatorname{core}(P_j^*)$. Consider a cluster $P_j^* \in \mathcal{C}$ and the matching cluster P_i. We have, $\operatorname{core}(P_j^*) \cap (P_1 \cup \dots \cup P_{i-1}) = \varnothing$ and, in particular, $u_j \notin P_1 \cup \dots \cup P_{i-1}$. We have that u_j is connected to all vertices in $\operatorname{core}(P_j^*)$ in the graph $G_{aux}[V \setminus (P_1 \cup \dots P_{i-1})]$. Thus u_j has degree at least $|\operatorname{core}(P_j^*)|$ in $G_{aux}[V \setminus (P_1 \cup \dots P_{i-1})]$. Therefore, vertex u that we choose at iteration i has degree at least $|\operatorname{core}(P_j^*)|$ and $|P_i| \geq |\operatorname{core}(P_j^*)|$; in particular, $|P_i \setminus \operatorname{core}(P_j^*)| \geq |\operatorname{core}(P_j^*) \setminus P_i|$. We have,

$$|P_j^* \cap P_i| \geq |\operatorname{core}(P_j^*) \cap P_i| = |\operatorname{core}(P_j^*)| - |\operatorname{core}(P_j^*) \setminus P_i| \geq |\operatorname{core}(P_j^*)| - |P_i \setminus \operatorname{core}(P_j^*)|.$$

Observe that $P_i \setminus \operatorname{core}(P_j^*)$ does not contain any vertices from $\operatorname{core}(P_k^*)$ (for every k): since P_i is disjoint from $\operatorname{core}(P_k^*)$ for $k \neq j$, $P_i \setminus \operatorname{core}(P_j^*)$ is disjoint from $\operatorname{core}(P_k^*)$ for $k \neq j$; $P_i \setminus \operatorname{core}(P_j^*)$ is clearly disjoint from $\operatorname{core}(P_j^*)$. We have, $\sum_{P_i \text{ is matched with } P_j^*} |P_i \setminus \operatorname{core}(P_j^*)| \leq |V| - |\bigcup_{P_j^* \in \mathcal{C}} \operatorname{core}(P_j^*)|$. From Theorem 12 (item 3), we get $|V| - |\bigcup_{P_j^* \in \mathcal{C}} \operatorname{core}(P_j^*)| \leq n - (1-\eta)^2 n \leq 2\eta n$. Therefore,

$$\sum_{P_i \text{ is matched with } P_j^*} |P_i \cap P_j^*| \geq \Big(\sum_{P_j^* \in \mathcal{C}} |\operatorname{core}(P_j^*)|\Big) - 2\eta n \geq (1 - \eta)^2 n - 2\eta n \geq (1 - 4\eta)n.$$

We proved that the algorithm finds a clustering with classification error at most 4η. ■

References

Nir Ailon, Moses Charikar, and Alantha Newman. Aggregating inconsistent information: Ranking and clustering. *J. ACM*, 55(5):23:1–23:27, November 2008.

G D Bader and C W Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, 4(1):2, 2003.

Maria-Florina Balcan and Mark Braverman. Finding low error clusterings. In *Conference on Learning Theory (COLT)*, 2009.

Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine Learning*, 56 (1-3):89–113, 2004.

Amir Ben-Dor, Ron Shamir, and Zohar Yakhini. Clustering gene expression patterns. *Journal of Computational Biology*, 6(3/4):281–297, 1999.

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

Steven Kay Butler. *Eigenvalues and structures of graphs*. PhD thesis, UC San Diego, 2008.

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

Shuchi Chawla, Konstantin Makarychev, Tselil Schramm, and Grigory Yaroslavtsev. Near optimal LP rounding algorithm for correlation clustering on complete and complete k-partite graphs. *CoRR*, abs/1412.0681, 2014.

Yudong Chen, Sujay Sanghavi, and Huan Xu. Clustering sparse graphs. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 2204–2212. Curran Associates, Inc., 2012.

Yudong Chen, Ali Jalali, Sujay Sanghavi, and Huan Xu. Clustering partially observed graphs via convex optimization. *Journal of Machine Learning Research*, 15:2213–2238, 2014.

Erik D. Demaine, Dotan Emanuel, Amos Fiat, and Nicole Immorlica. Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361:172–187, 2006. Approximation and Online Algorithms.

Micha Elsner and Warren Schudy. Bounding and comparing methods for correlation clustering beyond ilp. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, ILP '09, pages 19–27, 2009.

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.

Uriel Feige and Robert Krauthgamer. Finding and certifying a large hidden clique in a semirandom graph. *Random Struct. Algorithms*, 16:195–208, March 2000.

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

David A Freedman. On tail probabilities for martingales. *The Annals of Probability*, pages 100–118, 1975.

Naveen Garg, Vijay V. Vazirani, and Mihalis Yannakakis. Approximate max-flow min-(multi)cut theorems and their applications. In *Proceedings of Symposium on Theory of Computing*, pages 698–707, 1993.

M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

Amir Globerson, Tim Roughgarden, David Sontag, and Cafer Yildirim. Tight error bounds for structured prediction. *CoRR*, abs/1409.5834, 2014.

Brian Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *Phys. Rev. E*, 83:016107, Jan 2011.

Alexandra Kolla, Konstantin Makarychev, and Yury Makarychev. How to play unique games against a semi-random adversary: Study of semi-random models of unique games. In *Proceeding of Symposium on Foundations of Computer Science*, pages 443–452, 2011.

Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew Tomkins. Trawling the web for emerging cyber-communities. In *Computer Networks*, pages 1481–1493, 1999.

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. Sorting noisy data with partial information. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, pages 515–528, 2013.

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.

Claire Mathieu and Warren Schudy. Correlation clustering with noisy input. In *Proceedings of Symposium on Discrete Algorithms*, pages 712–728, 2010.

Mark Newman, Albert-Laszlo Barabasi, and Duncan J. Watts. *The Structure and Dynamics of Networks: (Princeton Studies in Complexity)*. Princeton University Press, Princeton, NJ, USA, 2006.

Sebastian Nowozin and Christoph H. Lampert. Structured learning and prediction in computer vision. *Foundations and Trends in Computer Graphics and Vision*, 6(34):185–365, 2010.

Noah A. Smith. *Linguistic Structure Prediction*. Synthesis Lectures on Human Language Technologies. Morgan and Claypool, May 2011.

Chaitanya Swamy. Correlation clustering: Maximizing agreements via semidefinite programming. In *Proceedings of Symposium on Discrete Algorithms*, pages 526–527, 2004.

Ben Taskar, Ming-fai Wong, Pieter Abbeel, and Daphne Koller. Link prediction in relational data. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems (NIPS) 16*. Cambridge, MA: MIT Press, 2004.

Julian Yarkony, Alexander T. Ihler, and Charless C. Fowlkes. Fast planar correlation clustering for image segmentation. In *12th European Conference on Computer Vision (ECCV)*, pages 568–581, 2012.

Xinyuan Zhao, Defeng Sun, and Kim-Chuan Toh. A newton-cg augmented lagrangian method for semidefinite programming. *SIAM J. Optimization*, 20:1737–1765, 2010.

Appendix A. Empirical Results

This paper focuses on designing an algorithm with provable theoretical guarantees for correlation clustering in a natural semi-random model. We have tested our algorithm to confirm that it is easily implementable and scalable. We used the SDPNAL MATLAB library to solve the semidefinite programming (SDP) relaxation for the problem (Zhao et al., 2010). We implemented the recovery algorithm from Section 2 in C++, and also used a simple cleanup step that merges small clusters with the larger clusters based on their average inner products (this extra step can only improve our theoretical guarantees). We note that we could solve the SDP relaxation for instances with thousands of vertices since we used a very basic SDP relaxation without ℓ_2^2-triangle inequality constraints.

We tested the algorithm on random $G(n, p)$ graphs with 4 planted clusters of size $n/4$ each, with the error rate (the probability of flipping the label) $\varepsilon = 0.2$. We used the same values of n as were used in (Chen et al., 2014); we chose values of p smaller than or close to the minimal values for which the algorithm of (Chen et al., 2014) works (Chen et al. do not report the exact values of probabilities p; we took approximate values from Figure 2 in their paper). We summarize our results in Table 1.

		run number					
n	p	1	2	3	4	avg.	%
200	0.25	0	0	2	2	1	0.50%
400	0.19	6	6	4	4	5	1.25%
1000	0.15	0	0	0	0	0	0.00%
2000	0.13	0	0	0	0	0	0.00%

Table 1: The table summarizes results of our experiments. The first and second columns list the values of n and p, respectively. The next four columns list the number of misclassified vertices in 4 runs of the program; column 7 lists the average number of misclassified vertices; column 8 shows this number as the percent of the total number of vertices.

Appendix B. Epsilon Net for SDP Solutions

In order to use Theorem 9, we need to prove that the set of all SDP solutions to our problem has a small epsilon net. We use the following lemma from Makarychev et al. (2013).

Lemma 13 (ITCS, Lemma 2.7) *For every graph $G = (V, E)$ on n vertices ($V = \{1, \dots, n\}$) with the average degree $\Delta = 2|E|/|V|$, real $M \geq 1$, and $\gamma \in (0, 1)$, there exists a set of matrices $\mathcal{W}$ of size at most $|\mathcal{W}| \leq \exp(O(\frac{nM^4 \log \Delta}{2\gamma^2} + n \log n))$ such that: for every collection of vectors $L(1), \dots, L(n)$, $R(1), \dots R(n)$ with $\|L(u)\| = M$, $\|R(v)\| = M$ and $\langle L(u), R(v)\rangle \in [0, 1]$, there exists $W \in \mathcal{W}$ satisfying for every $(u, v) \in E$:*

$$w_{uv} \leq \langle L(u), R(v)\rangle \leq w_{uv} + \gamma;$$

$$w_{uv} \in [0, 1].$$

By letting G be the complete graph, $M = 1$, $L(u) = R(u) = f(u)$, we get the following corollary.

Corollary 14 *For every $\gamma \in (0, 1)$, there exists a set of matrices $\mathcal{W}$ of size at most $|\mathcal{W}| \leq \exp\left(O(n\gamma^{-2} \log n)\right)$ such that: For every collection of vectors $\{f(u)\}$, there exists $W \in \mathcal{W}$ satisfying for every (u, v):*

$$|w_{uv} - \langle f(u), f(v)\rangle| \leq \gamma.$$

Appendix C. Structural Theorem – Proof of Theorem 3

Define f and f^* as in (2) and (3). Recall, that the algorithm removes all edges $(u, v) \in E$ with $f(u, v) \geq (1 - \gamma)$. We show that the number of edges inconsistent with the planted partition $\mathcal{P}^*$ that are remain in the graph after the fist step of the algorithm is small with high probability.

Proof [Proof of Theorem 3] For $(u, v) \in E$, let

$$X_{(u,v)} = \begin{cases} 1, & \text{if } (u, v) \in E_R; \\ -1, & \text{otherwise.} \end{cases}$$

Let $Q_+ = E_R$ and $Q_- = E^* \setminus E_R$. Then, $Q \subset Q_+$. Observe, that $f(u, v) = f^*(u, v)$ if $(u, v) \in E \setminus Q = Q_-$ and $f(u, v) = 1 - f^*(u, v)$ if $(u, v) \in Q \subset Q_+$. The SDP value is upper bounded by the optimal value OPT, which in turn is at most $c(Q)$. Write,

$$SDP = \sum_{(u,v)\in E} c(u, v)f(u, v) = \sum_{(u,v)\in E\setminus Q} c(u, v)f^*(u, v) + \sum_{(u,v)\in Q} c(u, v)(1 - f^*(u, v)) \leq c(Q).$$

Therefore,

$$\sum_{(u,v)\in E\setminus Q} c(u, v)f^*(u, v) \leq c(Q) - \sum_{(u,v)\in Q} c(u, v)(1 - f^*(u, v)) = \sum_{(u,v)\in Q} c(u, v)f^*(u, v).$$

We rewrite this expression as follows,

$$\sum_{(u,v)\in Q\cup Q_-} X_{(u,v)}c(u, v)f^*(u, v) \geq 0. \tag{7}$$

Suppose that

$$\sum_{(u,v)\in Q: f(u,v)\leq 1-\delta} c(u,v) \geq \Lambda + \frac{6\gamma}{1-2\varepsilon} c(Q).$$

For $(u,v) \in Q$, $f(u,v) = 1 - f^*(u,v)$. Thus, $\{(u,v) \in Q : f(u,v) \leq 1-\delta\} = \{(u,v) \in Q : f^*(u,v) \geq \delta\}$, and

$$\sum_{(u,v)\in Q} c(u,v) f^*(u,v) \geq \delta\Lambda + \frac{6\delta\gamma}{1-2\varepsilon} c(Q). \tag{8}$$

By Theorem 9 and Corollary 14, the probability that inequalities (7) and (8) hold is at most

$$2\exp\big(O(n\gamma^{-2}\delta^{-2}\log n)\big)\exp\big(-{}^1\!/\!{}_5(1-2\varepsilon)^2\delta\Lambda\big) = o(1),$$

for an appropriate choice of the constant C in the bound on Λ. ■

Appendix D. Proof of Lemmas 5, 6 and Theorem 9

Proof [Proof of Lemma 5] Let f_{OPT} correspond to the "integral" SDP solution corresponding to the optimal solution OPT. In this solution, $f_{OPT}(u,v) = 1$ for positive edges (u,v) which are across different clusters and negative edges (u,v) which are in the same cluster. This SDP solution has cost OPT and satisfies the conditions of Theorem 3. Hence, w.h.p., $c\left(Q \setminus (Q \cap OPT)\right) \leq \frac{\delta}{D} \cdot c(Q) + \Lambda$. Hence,

$$c(Q) - OPT \leq \frac{\delta}{D} c(Q) + \Lambda \qquad \text{and} \qquad OPT \geq (1 - \frac{\delta}{D}) \cdot c(Q) - \Lambda.$$

■

Proof [Proof of Lemma 6] To prove the desired upper bound (5), we estimate the probability that $\sum_{t=1}^T X_t w(e_t) c_t + \frac{1-2\varepsilon}{2} \sum_{t=1}^T w(e_t) c_t \geq 0$ and $\sum_{t=1}^T w(e_t) c_t \in [\Lambda', 2\Lambda']$ for a fixed $w \in \mathcal{W}$ and $\Lambda' \geq \Lambda$. Then we apply the union bound for all $w \in \mathcal{W}$, and Λ' of the form $2^i\Lambda$.

Fix a $w \in \mathcal{W}$ and $\Lambda' = 2^i$. Each X_{t+1} is independent of $\mathcal{F}'_t$, hence $\mathbb{E}[X_{t+1} w(e_{t+1}) c_{t+1} \mid \mathcal{F}'_t] = \mathbb{E}[X_{t+1}] w(e_{t+1}) c_{t+1} = (2\varepsilon - 1) w(e_{t+1}) c_{t+1}$. Thus,

$$S_\tau \equiv \sum_{t=1}^{\tau} (X_t + 1 - 2\varepsilon) w(e_t) c_t$$

is a martingale. Note that $|S_{t+1} - S_t| \leq w(e_{t+1}) c_{t+1} \leq c_{t+1}$ and

$$\mathrm{Var}[X_{t+1} w(e_{t+1} c_{t+1}) \mid \mathcal{F}'_t] = 4\varepsilon(1-\varepsilon) w(e_{t+1})^2 c_{t+1}^2 \leq 4\varepsilon(1-\varepsilon) w(e_{t+1}) c_{t+1}.$$

If $\sum_{t=1}^T X_t w(e_t) c_t + \frac{1-2\varepsilon}{2} \sum_{t=1}^T w(e_t) c_t \geq 0$ and $\sum_{t=1}^T w(e_t) c_t \in [\Lambda', 2\Lambda']$, then

$$S_T = \left[\sum_{t=1}^T X_t w(e_t) c_t + \frac{(1-2\varepsilon)}{2} \sum_{t=1}^T w(e_t) c_t\right] + \frac{(1-2\varepsilon)}{2} \sum_{t=1}^T w(e_t) c_t \geq \frac{(1-2\varepsilon)}{2} \Lambda',$$

and

$$\sum_{t=1}^{T} \mathrm{Var}[X_t w(e_t) c_t \mid \mathcal{F}'_{t-1}] = 4(\varepsilon - \varepsilon^2) \sum_{t=1}^{T} w(e_t) c_t \le 8\varepsilon(1-\varepsilon)\Lambda'.$$

Now, by Freedman's inequality (see Freedman (1975)),

$$\begin{aligned}\Pr\left(S_T \ge (1-2\varepsilon)\Lambda' \text{ and } \sum_{t=1}^{T} \mathrm{Var}[X_t w(e_t) c_t \mid \mathcal{F}_{t-1}] \le 8\varepsilon(1-\varepsilon)\Lambda'\right) &\le e^{-\frac{(1-2\varepsilon)^2\Lambda'^2}{2((1-2\varepsilon)\Lambda' + 8\varepsilon(1-\varepsilon)\Lambda')}} \\ &= e^{-\frac{(1-2\varepsilon)^2\Lambda'}{5}},\end{aligned}$$

and

$$\begin{aligned}\Pr\left(\sum_{t=1}^{T} X_t w(e_t) c_t \ge 0 \text{ and } \sum_{t=1}^{T} w(e_t) c_t \in [\Lambda', 2\Lambda']\right) &\le \Pr\left(S_T \ge (1-2\varepsilon)\Lambda' \text{ and } \sum_{t=1}^{T} w(e_t)^2 c_t^2 \le 2\Lambda'\right) \\ &\le e^{-1/5\,(1-2\varepsilon)^2\Lambda'} = \left(e^{-1/5\,(1-2\varepsilon)^2\Lambda}\right)^{2^i}.\end{aligned}$$

Summing up this upper bound over all $w \in \mathcal{W}$ and $\Lambda' = 2^i\Lambda$, we get (5). ■

Proof [Proof of Theorem 9] Let $\mathcal{W}$ be a γ–net for $\mathcal{Z}$. For simplicity of exposition we subtract $\min(\gamma, w(i))$ from all coordinates of vectors $w \in \mathcal{W}$. Thus, we assume that for all $z \in \mathcal{Z}$, there exists $w \in \mathcal{W}$ such that $w(i) \le z(i) \le w(i) + 2\gamma$ and $w(i) \ge 0$ for all i (see Remark 8).

Suppose that for some $z \in \mathcal{Z}$ and $Q_\oplus \subset Q_+$, the inequalities

$$\sum_{t \in Q_\oplus \cup Q_-} X_t z(e_t) c_t \ge 0 \tag{9}$$

and

$$\sum_{t \in Q_\oplus} z(e_t) c_t \ge \Lambda + \frac{6\gamma}{1-2\varepsilon} \sum_{t \in Q_\oplus} c_t \tag{10}$$

hold. Pick a $w \in \mathcal{W}$, such that $w(i) \le z(i) \le w(i) + 2\gamma$ for all i. We replace $z(e_t)$ with $w(e_t)$ in (10):

$$\sum_{t \in Q_\oplus} w(e_t) c_t \ge \sum_{t \in Q_\oplus} (z(e_t) - 2\gamma) c_t \ge \Lambda + \frac{4\gamma}{(1-2\varepsilon)} \cdot \sum_{t \in Q_\oplus} c_t. \tag{11}$$

Then,

$$\begin{aligned}\sum_{t=1}^{T} X_t w(e_t) c_t + \frac{1-2\varepsilon}{2} \sum_{t=1}^{T} w(e_t) c_t &\ge \sum_{t \in Q_\oplus \cup Q_-} X_t w(e_t) c_t + \frac{1-2\varepsilon}{2} \sum_{t \in Q_\oplus} w(e_t) c_t \qquad (12) \\ &\ge \left[\sum_{t \in Q_\oplus} (z(e_t) - 2\gamma) c_t - \sum_{Q_-} z(e_t) c_t\right] + 2\gamma \sum_{t \in Q_\oplus} c_t \\ &= \sum_{t \in Q_\oplus \cup Q_-} X_t z(e_t) c_t \ge 0.\end{aligned}$$

By Lemma 6, there exists a $w \in \mathcal{W}$ satisfying (11) and (12) with probability at most $2Ne^{-1/5(1-2\varepsilon)^2\Lambda}$. This concludes the proof. ■

Appendix E. Proof of Theorem 12

Proof [Proof of Theorem 12] Let $\delta = \gamma = (n \log n / c(E))^{1/6}$. Let Λ and Q be as in Theorem 3. Let $\sigma = 6\delta/(1-2\varepsilon)$. Note that $\Lambda = O(\delta c(E)^{5/6}/(1-2\varepsilon)^2)$.

Define f as in (2):

$$f(u,v) = \begin{cases} 1 - \langle \bar{u}, \bar{v} \rangle, & \text{if } (u,v) \in E_+; \\ \langle \bar{u}, \bar{v} \rangle, & \text{if } (u,v) \in E_-. \end{cases} \tag{13}$$

Consider the set of edges $E_{\text{flip}} = \{(u,v) \in E : f(u,v) > 1 - \delta\}$. Change the sign of each edge in E_{flip} and obtain a new partitioning of E into positive and negative edges, $\hat{E}_+$ and $\hat{E}_-$:

$$\begin{aligned} \hat{E}_+ &= E_+ \triangle E_{\text{flip}} = \{(u,v) \in E_+ : f(u,v) \le 1 - \delta\} \cup \{(u,v) \in E_- : f(u,v) > 1 - \delta\}, \\ \hat{E}_- &= E_- \triangle E_{\text{flip}} = \{(u,v) \in E_- : f(u,v) \le 1 - \delta\} \cup \{(u,v) \in E_+ : f(u,v) > 1 - \delta\}. \end{aligned}$$

Let us now consider the corresponding instance $\hat{\mathcal{I}} = \left\{ G = (V, E, c), (\hat{E}_+, \hat{E}_-) \right\}$. Let $\hat{f}$ be the analog of function f for $\hat{\mathcal{I}}$:

$$\hat{f}(u,v) = \begin{cases} 1 - \langle \bar{u}, \bar{v} \rangle, & \text{if } (u,v) \in \hat{E}_+ \\ \langle \bar{u}, \bar{v} \rangle, & \text{if } (u,v) \in \hat{E}_- \end{cases} = \begin{cases} f(u,v), & \text{if } (u,v) \notin E_{\text{flip}}; \\ 1 - f(u,v), & \text{if } (u,v) \in E_{\text{flip}}. \end{cases} \tag{14}$$

Similarly, let $\widehat{SDP} = \sum_{(u,v)\in E} c(u,v)\hat{f}(u,v)$ be the cost of the SDP solution $\{\bar{u}\}$ for $\hat{\mathcal{I}}$.

Lemma 15 *With probability $1 - o(1)$, the following properties hold.*

1. $c(Q \setminus E_{\text{flip}}) \le \sigma c(Q) + \Lambda$.
2. $c(E_{\text{flip}} \setminus Q) \le (2\delta + \sigma)c(Q) + \Lambda$.
3. *Then* $\widehat{SDP} \le (2\delta + \sigma)c(Q) + \Lambda$.

Proof 1. From Theorem 3, we get that $c(Q \setminus E_{\text{flip}}) \le \sigma c(Q) + \Lambda$ with probability $1 - o(1)$.
2. Write $c(E_{\text{flip}} \setminus Q) = c(E_{\text{flip}}) - c(Q \cap E_{\text{flip}})$. Now we bound $c(E_{\text{flip}})$ and $c(Q \cap E_{\text{flip}})$. Note that

$$SDP = \sum_{(u,v)\in E} c(u,v) f(u,v) \ge \sum_{(u,v)\in E_{\text{flip}}} c(u,v)(1-\delta) = (1-\delta)c(E_{\text{flip}}).$$

Hence,

$$c(E_{\text{flip}}) \le SDP/(1-\delta) \le c(Q)/(1-\delta) \le (1+2\delta)c(Q),$$

here, we used that $\{\bar{u}\}$ is an optimal SDP solution and therefore $SDP \le c(Q)$.

By item 1, $c(Q \cap E_{\text{flip}}) = c(Q) - c(Q \setminus E_{\text{flip}}) \ge (1-\sigma)c(Q) - \Lambda$. We get that

$$c(E_{\text{flip}} \setminus Q) \le (1+2\delta)c(Q) - (1-\sigma)c(Q) - \Lambda = (2\delta + \sigma)c(Q) + \Lambda.$$

3. From the second formula for $\hat{f}(u,v)$ in (14), we get that $f(u,v) - \hat{f}(u,v) = 2f(u,v) - 1 \ge 1 - 2\delta$ for $(u,v) \in E_{\text{flip}}$, and $f(u,v) - \hat{f}(u,v) = 0$ for $(u,v) \notin E_{\text{flip}}$. Therefore,

$$\begin{aligned} c(Q) - \widehat{SDP} \ge SDP - \widehat{SDP} &= \sum_{(u,v)\in E} c(u,v)(f(u,v) - \hat{f}(u,v)) \\ &= \sum_{(u,v)\in E_{\text{flip}}} c(u,v)(f(u,v) - \hat{f}(u,v)) \ge (1-2\delta)c(E_{\text{flip}}) \ge (1-2\delta)c(Q \cap E_{\text{flip}}) \\ &\ge (1-2\delta)((1-\sigma)c(Q) - \Lambda) \ge (1 - 2\delta - \sigma)c(Q) - \Lambda. \end{aligned}$$

Therefore, $\widehat{SDP} \leq (2\delta + \sigma)c(Q) + \Lambda$. ■

We now bound the total squared Euclidean length of all edges in E_+^*.

Lemma 16 *With probability* $1 - o(1)$, *we have*

$$\frac{1}{2} \sum_{(u,v) \in E_+^*} c(u,v) \|\bar{u} - \bar{v}\|^2 \leq (4\delta + 3\sigma)c(Q) + 3\Lambda$$

Proof Note that for $(u,v) \in \hat{E}_+$, $\frac{1}{2}\|\bar{u} - \bar{v}\|^2 = \hat{f}(u,v)$ and thus

$$\frac{1}{2} \sum_{(u,v) \in \hat{E}_+} c(u,v) \|\bar{u} - \bar{v}\|^2 \leq \widehat{SDP}.$$

Also, $E_+^* \cap \hat{E}_- \subset (Q \setminus E_{\text{flip}}) \cup (E_{\text{flip}} \setminus Q)$. Thus, by Lemma 15, $c(E_+^* \cap \hat{E}_-) \leq c(Q \setminus E_{\text{flip}}) + c(E_{\text{flip}} \setminus Q) \leq 2(\delta + \sigma)c(Q) + 2\Lambda$. We have,

$$\begin{aligned} \frac{1}{2} \sum_{(u,v) \in E_+^*} c(u,v) \|\bar{u} - \bar{v}\|^2 &\leq \frac{1}{2} \sum_{(u,v) \in E_+^* \cap \hat{E}_+} c(u,v) \|u - v\|^2 + \frac{1}{2} \sum_{(u,v) \in E_+^* \cap \hat{E}_-} c(u,v) \|u - v\|^2 \\ &\leq \widehat{SDP} + c(E_+^* \cap \hat{E}_-) = (4\delta + 3\sigma)c(Q) + 3\Lambda. \end{aligned}$$

■

We are ready to prove Theorem 12. Recall that

$$\eta = \frac{C_2}{(1 - 2\varepsilon)} \left(\frac{n \log n}{c(E)}\right)^{1/12} \cdot \left(\frac{\alpha^3}{\beta \lambda_{gap}}\right)^{1/2}.$$

We assume that $\eta < 1/4$ as otherwise the statement of the theorem is trivial. We let $\eta' = \eta/\alpha^2$. Let

$$\rho_{\text{avg}}^2 = \frac{1}{c(E_+^*)} \sum_{(u,v) \in E_+^*} c(u,v) \|\bar{u} - \bar{v}\|^2 \leq O(\sigma c(Q) + \Lambda)/c(E_+^*) \leq O(\sigma + \Lambda/c(E))/\beta.$$

Let $E_+^*(i) = \left\{(u,v) \in E_+^* : u, v \in P_i^*\right\}$ be the set of edges within cluster P_i^*. Write

$$\sum_{i=1}^{k} \sum_{(u,v) \in E_+^*(i)} c(u,v) \|\bar{u} - \bar{v}\|^2 = c(E_+^*) \rho_{\text{avg}}^2.$$

Let $\mathcal{C}$ be the set of clusters P_i^* such that

$$\sum_{(u,v) \in E_+^*(i)} c(u,v) \|\bar{u} - \bar{v}\|^2 \leq c(E_+^*(i)) \rho_{\text{avg}}^2 / (\alpha \eta').$$

By Markov's inequality, $\sum_{P_i^* \notin \mathcal{C}} c(E_+^*(i)) \leq \alpha \eta' c(E_+^*)$. By the Approximate Cluster Regularity condition in Assumptions 10, $c(E_+^*(i)) \geq \frac{1}{\alpha}(|P_i^*|/n)c(E_+^*)$. We get that $\sum_{P_i^* \notin \mathcal{C}} |P_i^*| \leq \alpha^2 \eta' n$ and thus $\sum_{P_i^* \in \mathcal{C}} |P_i^*| \geq (1 - \eta)n$. We get that item 1 in the statement of the theorem holds.

Recall that the normalized Laplacian of a graph H is a matrix $\mathcal{L}$ with unit diagonal and non-diagonal entries $L_{uv} = -1/\sqrt{\deg_H u \deg_H v}$. We use the following form of the the Poincaré inequality, which immediately follows from Theorem 4 and formula (1.4) in (Butler, 2008).

Theorem 17 (Poincaré Inequality) *Consider a graph $H = (V_H, E_H, c_H)$ and set of vectors $\{\bar{u}\}_{u \in V_H}$. Let λ be the second smallest eigenvalue of the normalized Laplacian of H. Suppose that for some α and every two vertices u and v, $\deg_H u \le \alpha \deg_H v$. Then we have*

$$\frac{1}{|V_H|^2} \sum_{u,v \in V} \|\bar{u} - \bar{v}\|^2 \le \frac{\alpha}{\lambda \cdot c_H(E_H)} \sum_{(u,v) \in E_H} c_H(u,v) \|\bar{u} - \bar{v}\|^2.$$

We apply the Poincaré inequality to the induced graph $G[P_i^*]$. We have for each cluster $P_i^* \in \mathcal{C}$,

$$\frac{1}{|P_i^*|^2} \sum_{u,v \in P_i^*} \|\bar{u} - \bar{v}\|^2 \le \frac{1}{\lambda_{gap}} \frac{\alpha}{c(E_+^*(i))} \sum_{(u,v) \in E_+^*(i)} c(u,v) \|\bar{u} - \bar{v}\|^2 \le \frac{\alpha \rho_{\text{avg}}^2}{\lambda_{gap}(\alpha \eta')} \le \frac{\rho_{\text{avg}}^2}{\lambda_{gap} \eta'}.$$

Therefore,

$$\min_{u \in P_i^*} \left(\frac{1}{|P_i^*|} \sum_{v \in P_i^*} \|\bar{u} - \bar{v}\|^2 \right) \le \frac{1}{|P_i^*|} \sum_{u \in P_i^*} \left(\frac{1}{|P_i^*|} \sum_{v \in P_i^*} \|\bar{u} - \bar{v}\|^2 \right) \le \frac{\rho_{\text{avg}}^2}{\lambda_{gap} \eta'}.$$

Thus we can choose u_i in each $P_i^* \in \mathcal{C}$ such that $\frac{1}{|P_i^*|} \sum_{v \in P_i^*} \|\bar{u}_i - \bar{v}\|^2 \le \frac{\rho_{\text{avg}}^2}{\lambda_{gap} \eta'}$. This choice of vertices u_i defines sets $\text{core}(P_i^*)$, as in the statement of the theorem.

Using again Markov's inequality, we get that for at least a $1 - \eta/a$ fraction of vertices v in P_i^*, $\|\bar{u}_i - \bar{v}\|^2 \le a\rho_{\text{avg}}^2/(\lambda_{gap}\eta\eta')$. From the bound $\rho_{\text{avg}}^2 = O(\sigma + \Lambda/c(E))/\beta$ and (1), we get $\rho_{\text{core}}^2 \ge a\rho_{\text{avg}}^2/(\lambda_{gap}\eta\eta')$ and

$$|\operatorname{core}(P_i^*)| \ge |\left\{v \in P_i^* : \|\bar{u}_i - \bar{v}\|^2 \le a\rho_{\text{avg}}^2/(\lambda_{gap}\eta\eta')\right\}| \ge (1 - \eta/a)|P_i^*|.$$

In particular, item 2 in the statement of the theorem holds. We get item 3 from items 1 and 2.

Finally, we show that $\|\bar{u}_i - \bar{u}_j\| \ge \rho_{\text{inter}}$ for every two distinct clusters $P_i^*, P_j^* \in \mathcal{C}$. To this end, we show that there are vertices $v' \in \text{core}(P_i^*)$ and $v'' \in \text{core}(P_j^*)$ such that $\|\bar{v}' - \bar{v}''\| \ge \rho_{\text{inter}} + 2\rho_{\text{core}}$, and thus $\|\bar{u}_i - \bar{u}_j\| \ge (\rho_{\text{inter}} + 2\rho_{\text{core}}) - \|u_i - v'\| - \|u_j - v''\| \ge \rho_{\text{inter}}$. Assume to the contrary that $\|\bar{v}' - \bar{v}''\| < \rho_{\text{inter}} + 2\rho_{\text{core}}$ for every $v' \in \text{core}(P_i^*)$ and $v'' \in \text{core}(P_j^*)$. Let $E_{ij} = \left\{(v', v'') \in E : v' \in \text{core}(P_i^*), v'' \in \text{core}(P_j^*)\right\}$.

Since $E_{ij} \subset E_-^*$, we have for every $(v', v'') \in E_{ij} \setminus (Q \triangle E_{\text{flip}})$,

$$\hat{f}(v', v'') = \langle \bar{v}', \bar{v}'' \rangle = 1 - \|\bar{v}' - \bar{v}''\|^2/2 \ge 1 - (\rho_{\text{inter}} + 2\rho_{\text{core}})^2/2 = 1/2.$$

Therefore,

$$\widehat{SDP} \ge \sum_{(v',v'') \in E_{ij} \setminus (Q \triangle E_{\text{flip}})} c(v', v'') f(v', v'') \ge c(E_{ij} \setminus (Q \triangle E_{\text{flip}}))/2.$$

From the Approximate Intercluster Regularity condition and bounds $|\operatorname{core}(P_i^*)| \ge (1 - \eta/a)|P_i^*|$ and $|\operatorname{core}(P_j^*)| \ge (1 - \eta/a)|P_j^*|$, we get $c(E_{ij}) \ge (1 - 2\eta)\beta_{ij}c(E)$. By Lemma 15,

$$c(Q \triangle E_{\text{flip}}) \le 2(\delta + \sigma)c(Q) + 2\Lambda \le 2(\delta + \sigma)c(E) + 2\Lambda.$$

By the Intercluster Density condition in Assumptions 10, our choice of δ and our assumption that $\eta \leq 1/4$, we have

$$c(E_{ij} \setminus (Q \triangle E_{\text{flip}})) \geq ((1-2\eta)\beta_{ij} - 2\delta - 2\sigma)c(E) - 2\Lambda \geq \beta_{ij}c(E)/3.$$

We get that

$$(2\delta + \sigma)c(E) + \Lambda \geq \widehat{SDP} \geq \beta_{ij}c(E)/6,$$

which contradicts to the Intercluster Density condition and our choice of δ. ■