Detecting Network Cliques with Radon Basis Pursuit

Xiaoye Jiang
Stanford University

Yuan Yao
Peking University

Han Liu
Johns Hopkins University

Leonidas Guibas
Stanford University

Abstract

In this paper, we propose a novel formulation of the network clique detection problem by introducing a general network data representation framework. We show connections between our formulation with a new algebraic tool, namely *Radon basis pursuit in homogeneous spaces*. Such a connection allows us to identify rigorous recovery conditions for clique detection problems. Practical approximation algorithms are also developed for solving empirical problems and their usefulness is demonstrated on real-world datasets. Our work connects two seemingly different areas: *network data analysis* and *compressed sensing*, which helps to bridge the gap between the research of network data and the classical theory of statistical learning and signal processing.

1 Introduction

In the past decade, the research of network data has increased dramatically. Examples include scientific studies involving web data or hyper text documents connected via hyperlinks, social networks or user profiles connected via friend links, co-authorship and citation network connected by collaboration or citation relationships, gene or protein networks connected by regulatory relationships, and much more. Due to the increasing importance of network data, principled analytical and modeling tools are crucially needed.

Towards this goal, researchers from the *network modeling* community have proposed many models to explore and predict the network data. These models roughly fall into two categories: *static* (there is only one single snapshot of the network) and *dynamic* models (there are many snapshots of the network indexed by different time points). Examples include the Erdös-Rényi-Gilbert random graph model [10, 11], latent space model [14], stochastic blockmodel [2, 27], the preferential attachment model [1], and dynamic latent space model [24]. A comprehensive review of these models is provided in [12].

In network data analysis, the problem of identifying communities [17] or cliques[1] based on partial information arises frequently in a variety of applications, including identity management [13], statistical ranking [9, 15], and in particular, social networks [19]. In these applications we are typically given a network with the nodes representing players, items, or characters, and edge weights summarizing the observed pairwise interactions. *The basic problem is to determine communities or cliques within the network by observing the frequencies of low order interactions,* since in reality such low order interactions are often governed by a considerably smaller number of high order communities or cliques. In this sense we could formulate our problem as an *inverse problem* in networks, where one tries to infer a sparse signal over communities by sensing low order interactions. In particular, we cast our problem as a *compressed sensing* problem. Compressed sensing, also known as compressive sensing and compressive sampling, is a technique for finding sparse solutions to underdetermined linear systems. In statistical machine learning, it is related to reconstructing a signal which has a sparse representation in a large dictionary. The field of compressed sensing has existed for decades, but recently it has exploded due to the important contributions of [4, 5, 6, 26]. Before rigorously formulating the problem, we provide a concrete illustrative example.

Motivating Example: Detecting communities in social networks is of extraordinary importance. It can be used to understand the organization or collaboration structure of a social network. However, we do not have direct mechanisms to sense social communities. Instead, we have partial, low order interaction information. For example, imagine we have a small

Appearing in Proceedings of the 15^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2012, La Palma, Canary Islands. Volume 22 of JMLR: W&CP 22.

[1] A clique means a complete subgraph of the network.

social network of four people — Alice, Bob, Cathy, and David. Suppose Alice, Bob, Cathy took the same class, and they co-appear in the same classrooms three times per week; Bob, Cathy, and David are gymnastic buddies, they hang out to gym twice per week. Thus, we will observe that pairwise co-appearances of Alice and Bob are three times per week, while pairwise co-appearances of Bob and Cathy are five times per week, etc. With all such pairwise co-appearance data among people available, we hope to detect the two social communities in the network. [2]

Such a network community detection problem has been intensively studied in the social network literature. However, we note that there is no consistent definition of a "community" across different literature. Most methods detect community based on network node partitioning. Among them, the most famous one is based on the modularity of a partition of the nodes in a group [22] . A shortcoming of partition-based methods is that they do not allow overlapping communities, which occur frequently in practice. Recently there have been growing interests in studying overlapping community structures [18]. Moreover, we note that many real-world applications also try to support the feature of overlapping communities, such as Google+, where an user can group his friends into different overlapping communities (friends, family, acquaintances, and so on). The relevance of cliques to overlapping communities was probably first addressed in the clique percolation method [23]. They model communities as maximal connected components of cliques in a graph where two k-cliques are said to be connected if they share $k-1$ nodes.

In this paper, we use the same definition as in [23] but are more interested in identifying cliques. We pursue an alternative approach on exploring networks based on clique information which potentially sheds light on multiple aspects of community structures. Roughly speaking, we assume that each low order subset is associated with a frequency. As shown in our motivating example, the pair "Alice and Bob" co-appear "three times per week", the pair "Bob and Cathy" co-appear "five times per week", and etc. We also assume that there are latent frequencies associated with high order subsets which we hope to infer. For example, the community "Alice, Bob, and Cathy" have classes "three times per week", while the community "Bob, Cathy, and David" go to gyms "twice per week". Clearly, the interaction frequency of a particular low order subset should be the sum of frequencies of high order subsets which it belongs to. Hence we consider a *generative mechanism* in which there exists a linear mapping from frequencies on high order subsets (usually sparsely distributed) to low order subsets. One typically can collect data on low order subsets while the task is to find those few dominant high order subsets.

[2]Low order interaction data can be accessed easily, and they can be maintained with moderate storage. In some cases, data do appear only in the form of low order interactions, e.g., who visited whose personal page, and etc.

2 Main Idea

In this section, we introduce a general network data representation framework, which facilitate the formulation of the clique detection problem.

We represent a network as a graph $G = (V, E)$, where $V = \{1, \ldots, n\}$ is the set of nodes and $E \subset V \times V$ is the set of edges. Let $B \in \mathbb{R}^{n\times n}$ be the adjacency matrix of the observed network whose element $B(i, j) \in \mathbb{R}$ represents a quantity associated with nodes i and j. Entries in the matrix B may indicate the co-appearance frequency of the pair i and j. As a bivariate function $B : V \times V \to \mathbb{R}$, we consider the following representation

$$B(i, j) = \sum_k c_k \phi_k(i, j) + z. \tag{1}$$

In our framework, we assume that (1) has a sparse representation with respect to a dictionary $A = [\phi_1, \ldots, \phi_N]$ where each $\phi_k : V \times V \to \mathbb{R}$ is a basis function, i.e., there exists a subset $S \subset \{1, \ldots, N\}$ with cardinality $|S| \ll N$, such that $c_k = 0$ for $k \notin S$. Here, z represents noise, and N may be infinitely large. We can view ϕ_k as evaluating a possibly infinite-dimensional function on a discrete set $V \times V$, thus the model (1) is intrinsically nonparametric and can model any static networks.

To address the clique detection problem, we focus on a specific design of the basis dictionary A in this paper. In such a dictionary, each basis can be interpreted as a clique, which is a complete subgraph of G and with a set of nodes $K \subset V$. We let ϕ_K be the adjacency matrix of a clique with nodes K, i.e. $\phi_K(i, j) = 1$ if $i, j \in K, i \neq j$ and 0 otherwise. Such a basis function leads to the clique detection problem studied in this paper.

In the sequel, without loss of generality, we assume that B is symmetric: $B = B^T$ and $\mathrm{diag}(B) = 0$. With these assumptions, to model B we only need to model its upper-triangle. For notational simplicity, we squeeze B into a vector $b \in \mathbb{R}^M$ where $M = n(n-1)/2$ is the number of upper-triangle elements in B. In this case each basis function becomes a vector $\phi_k \in \mathbb{R}^M$ and A becomes a M-by-N matrix. We denote by A_{pq} the element on the p-th row and q-th column of A. Here p indexes a pair of different nodes and q indexes a basis ϕ_q.

Given the dictionary A, we can recover the sparse representation in (1), by reconstructing $x = (x_1, \ldots, x_N)^T$ from the following problem

$$(\mathrm{P}_0) \qquad \min \|x\|_0 \quad \text{s.t. } \|b - Ax\|_z \leq \delta \qquad (2)$$

where $\|\cdot\|_z$ is a vector norm constructed using the knowledge of z. The problem in (2) is non-convex. In the sparse learning literature, a convex relaxation of (2) can be written as

$$(\mathrm{P}_1) \qquad \min \|x\|_1 \quad \text{s.t. } \|b - Ax\|_z \leq \delta. \qquad (3)$$

In both optimization problems, entries in x represent unknown frequencies associated with high order subsets, while entries in b represent observed frequencies associated with low order subsets. The matrix A can thus be interpreted as an operator that can linearly map frequencies on high order subsets to low order subsets.

Now, we see that the network clique detection problem fit nicely into the general framework (1). In such a generative model where the observed adjacency matrix is assumed to have a sparse representation in a large dictionary where each basis corresponds to a clique, we connect our framework with a new algebraic tool, namely *Radon basis pursuit in homogeneous spaces*. Our problem can be regarded as an extension of the work in [15] which studies sparse recovery of *functions on permutation groups*, while we reconstruct *functions on k-sets* (cliques), which are often called the *homogeneous spaces* associated with permutation groups in the literature [9]. It turns out that the discrete Radon basis becomes the natural choice instead of the Fourier basis considered in [15]. Unfortunately, the greedy algorithm for exact recovery in [15] cannot be applied to noisy settings, and in general the Radon basis does not satisfy the Restricted Isometry Property (RIP) [4] which is crucial for the universal recovery in compressed sensing. All of these leave us new challenges on addressing the noiseless exact recovery and stable recovery with noise. In this paper, we develop new theories and algorithms which guarantee exact, sparse, and stable recovery under the choice of Radon basis. These theories have deep roots in Basis Pursuit [7] and its extensions with uniformly bounded noise. We also provide practical algorithms on the clique recovery problem to illustrate the usefulness of our framework.

3 Clique Detection with Radon Basis Pursuit

Under the general framework in (1), we formulate the clique detection problem into a compressed sensing problem (3) named *Radon Basis Pursuit*. For this, we construct the dictionary A so that each column of A corresponds to one clique. The intuition of such a construction is that we assume there are several hidden cliques within the network, which are perhaps of different sizes and may have overlaps. Every clique has certain weights and the observed adjacency matrix B (or equivalently, the vectorized upper-triangle part of b) is a linear combination of many clique basis contaminated by the noise vector z.

For simplicity, we first restrict ourselves to the case that all the cliques are of the same size $k < n$. The case with mixed sizes will be discussed later. Let $C_1, C_2, \ldots, C_N$ be all the cliques of size k and each $C_j \subset V$. We have $N = \binom{n}{k}$. For each $q \in \{1, \ldots, N\}$, we construct the dictionary A as the following:

$$A_{pq} = \begin{cases} 1 & \text{if the } p\text{-th pair of nodes both lie in } C_q \\ 0 & \text{otherwise.} \end{cases}$$

The matrix A constructed here is related to discrete Radon transforms on homogeneous space. In fact, up to a constant and column scaling, the transpose matrix A^* is called the discrete Radon transform for two suitably defined homogeneous spaces [9]. Our usage here is to exploit the transpose matrix of the Radon transform to construct an over-complete dictionary, so that the observation b has a sparse representation with respect to it. Due to the limited space of this paper, we leave out the technical discussions of the Radon transformations.

The above formulation can be generalized to the case where b is a vector of length $\binom{n}{j}$ ($j \geq 2$) with the p'th entry in b characterizing a quantity associated with a j-set. The dictionary A will then be changed to a binary matrix $R^{j,k}$ with entries indicating whether a j-subset is a subset of a k-clique, i.e.,

$$R^{j,k}_{pq} = \begin{cases} 1 & \text{if the } p\text{-th subset of nodes all lie in } C_q \\ 0 & \text{otherwise.} \end{cases}$$

Therefore, the case where b is the vector of length $\binom{n}{2}$ corresponds to a special case where $A = R^{2,k}$. Our algorithms and theory hold for general $R^{j,k}$ with $j < k$.

Now we provide two concrete reconstruction programs for the clique identification problems:

$$(\mathcal{P}_1) \qquad \min \|x\|_1 \ \text{ s.t. } b = Ax$$
$$(\mathcal{P}_{1,\delta}) \qquad \min \|x\|_1 \ \text{ s.t. } \|Ax - b\|_\infty \leq \delta.$$

$\mathcal{P}_1$ is known as Basis Pursuit [7] where we consider an ideal case that the noise level is zero. For robust reconstruction against noise, we consider the relaxed program $\mathcal{P}_{1,\delta}$. The program in $\mathcal{P}_{1,\delta}$ differs from the Dantzig selector [6] which uses the constraint that $\|A^*(Ax - b)\|_\infty \leq \delta$. The reason for our choice of $\mathcal{P}_{1,\delta}$

lies in the fact that a more natural noise model for network data is bounded noise rather than Gaussian noise. Moreover, our linear programming formulation of $\mathcal{P}_{1,\delta}$ enables practical computation for large scale problems.

4 Mathematical Theory

One advantage of our new framework to represent network data is that it enables rigorous theoretical analysis of the corresponding convex programs.

4.1 Failure of Universal Recovery

Recently it was shown by [5] and [4] that $\mathcal{P}_1$ has a unique sparse solution x_0, if the matrix A satisfies the *Restricted Isometry Property* (RIP), i.e. for every subset of columns $T \subset \{1, \ldots, N\}$ with $|T| \leq s$, there exists a certain universal constant $\delta_s \in [0, \sqrt{2}-1)$ such that

$$(1-\delta_s)\|x\|_2^2 \leq \|A_T x\|_2^2 \leq (1+\delta_s)\|x\|_2^2, \quad \forall x \in \mathbb{R}^{|T|},$$

where A_T is the sub-matrix of A with columns indexed by T. Then exact recovery holds for all s-sparse signals x_0, whence called the *universal recovery*.

Unfortunately, in our construction of the basis matrix A, RIP is not satisfied unless for very small s. We have the following theorem regarding to the failure of universal recovery in our case.

Theorem 1. *Let $A = R^{j,k}$ with $j < k$. Unless $s < \binom{k+j+1}{k}$, there does not exist a $\delta_s < 1$ such that the inequalities*

$$(1-\delta_s)\|x\|_2^2 \leq \|A_T x\|_2^2 \leq (1+\delta_s)\|x\|_2^2, \quad \forall x \in \mathbb{R}^{|T|}$$

hold universally for every $T \subset \{1, \ldots, N\}$ with $|T| \leq s$.

Note that $\binom{k+j+1}{k}$ does not depend on the network size n, which will be problematic. We can only recover a constant number of cliques no matter how large the network is! The main problem for such a negative result is that the RIP tries to guarantee exact recovery for *arbitrary* signals with a sparse representation in A. Instead of studying such "universal" conditions, In this paper we seek conditions that secure exact recovery of a collection of sparse signals x_0, whose sparsity pattern satisfies certain conditions more appropriate to our setting. Such conditions could be more natural in reality, which will be shown in the sequel as simply requiring bounded overlaps between cliques.

4.2 Exact Recovery Conditions

Here we present our exact recovery conditions for x_0 from the observed data b by solving the linear program $\mathcal{P}_1$. Suppose A is an M-by-N matrix and x_0 is a sparse signal. Let $T = \mathrm{supp}(x_0)$, T^c be the complement of T, and A_T (or A_{T^c}) be the submatrix of A where we only extract column set T (or T^c, respectively). The following proposition from [5] characterizes the conditions that $\mathcal{P}_1$ has a unique condition.

Proposition 1. *Let $x_0 = (x_{01}, \ldots, x_{0N})^T$, we assume that $A_T^* A_T$ is invertible and there exists a vector $w \in R^M$ such that:*

1. $\langle A_i, w\rangle = \mathrm{sign}(x_{0i}), \forall i \in T$;
2. $|\langle A_j, w\rangle| < 1, \forall j \in T^c$.

Then x_0 is the unique solution for $\mathcal{P}_1$.

In other words, the theorem simply points out the necessary and sufficient condition that in the noise-free case $\mathcal{P}_1$ exactly recover the sparse signal x_0. The necessity comes from the KKT condition in convex optimization theory [5]. However this condition is difficult to check due to the presence of w. If we further assume that w lies in the column span of A_T, the condition in Proposition 1 reduces to the following condition.

Irrepresentable Condition (IRR) The matrix A satisfies the IRR condition with respect to $T = \mathrm{supp}(x_0)$, if $A_T^* A_T$ is invertible and $\|A_{T^c}^* A_T (A_T^* A_T)^{-1}\|_\infty < 1$, where $\|\cdot\|_\infty$ stands for the matrix ∞-norm, i.e., $\|A\|_\infty := \max_j \sum_i |A_{ij}|$.

Intuitively, the IRR condition requires that, for the true sparsity pattern x_0, the relevant bases A_T is not highly correlated with irrelevant bases A_T^c. Note that this condition only depends on A and x_0, which is easier to check. The assumption that w lies in the column span of A_T is mild; it is actually a necessary condition so that x_0 can be reconstructed by Lasso [25] or Dantzig selector [6], even under Gaussian-like noise assumptions [29, 30].

4.3 Detecting Cliques of Equal Size

In this section, we present sufficient conditions of IRR which can be easily verified. We consider the case that $A = R^{j,k}$ with $j < k$. Given data b about all j-subsets, we want to infer important k-cliques. Suppose x_0 is a sparse signal on all k-cliques. We have the following worst-case theorem, which follows from Lemma 1.

Theorem 2. *Let $T = \mathrm{supp}(x_0)$, if we enforce the overlaps among k-cliques in T to be no larger than r, then $r \leq j-2$ guarantees the IRR condition.*

Lemma 1. *Let $T = \mathrm{supp}(x_0)$ and $j \geq 2$. Suppose for any $\sigma_1, \sigma_2 \in T$, the two cliques corresponding to σ_1 and σ_2 have overlaps no larger than r, we have*

1. *If $r = j-2$, then $\|A_{T^c}^* A_T (A_T^* A_T)^{-1}\|_\infty < 1$;*

2. If $r = j-1$, then $\|A_{T^c}^* A_T (A_T^* A_T)^{-1}\|_\infty \le 1$ where equality holds with certain examples;

3. If $r = j$, there are counter examples such that $\|A_{T^c}^* A_T (A_T^* A_T)^{-1}\|_\infty > 1$.

The proof of Lemma 1 is based on combinatorial arguments. Theorem 2 provides a sufficient condition on allowed clique overlaps which guarantees the IRR Condition. Clique overlaps no larger than $j-2$ is sufficient to guarantee the exact sparse recovery by $\mathcal{P}_1$, while larger overlaps may violate the IRR Condition. This theorem is based on a worst-case analysis. In the following, we construct explicitly conditions which allow large overlaps while the IRR still holds, as long as such heavy overlaps do not occur too often among the cliques in T. The existance of a partition of T in the next theorem is a reasonable assumption in the network settings where network hierarchies exist.

Theorem 3. *Assume $(k+1)/2 \le j < k$, let $T = \mathrm{supp}(x_0)$. Suppose there exists a partition $T = T_1 \cup T_2 \cup \cdots \cup T_m$ with each T_i satisfies $|T_i| \le K$, such that for any σ_i, σ_j belong to the same partition, $|\sigma_i \cap \sigma_j| \le r$; for any σ_i, σ_j belong to different partitions, $|\sigma_i \cap \sigma_j| \le 2j-k-1$, If K satisfies $(K-1)\binom{r}{j}/\binom{k}{j} < 1/4$ and $\left(\binom{k-1}{j} + (K-1)\binom{(k+r)/2}{j}\right)/\binom{k}{j} \le 3/4$, then IRR holds.*

The basis matrix $A = R^{j,k}$ have $\binom{n}{k}$ bases, which is not polynomial with respect to k. As we will see from later sections, a practical implementation of the Radon basis pursuit for the clique detection problem works on a subset of bases among all $\binom{n}{k}$ bases. In that case, we are actually solving $\mathcal{P}_1$ and $\mathcal{P}_{1,\delta}$ with the basis matrix $\bar{A}$, which is only a submatrix of A with a subset of column bases extracted. We have the following theorem regarding this scenario.

Theorem 4. *Denote the set of all cliques for columns in $\bar{A}$ by S. Assume any two k-cliques in $S = T \cup T^c$ have intersections at most r, i.e. $\forall \sigma_i, \sigma_j \in S$, $|\sigma_i \cap \sigma_j| \le r$, where $T = \mathrm{supp}(x_0) \subset S$, and T^c is the complement of T with respect to S. Then IRR holds if*

$$r \le k / \left(|T|(1+\sqrt{|T|})\right)^{1/j}.$$

In summary, IRR is sufficient and almost necessary to guarantee exact recovery. Generally, the intuition behind the IRR is that *overlaps among cliques must be small*. In some cases, we can have large overlaps among the cliques, provided that they do not occur too often. In the next subsection, we show that IRR is also sufficient to guarantee stable recovery with noises.

4.4 Stable Recovery Theorems

In applications, one always encounters examples with noise such that exact sparse recovery is impossible. In this setting, $\mathcal{P}_{1,\delta}$ will be a good replacement of $\mathcal{P}_1$ as a robust reconstruction program. Here we present stable recovery theorem of $\mathcal{P}_{1,\delta}$ with bounded noise.

Theorem 5. *Under the general framework (1), we assume that $\|z\|_\infty \le \epsilon$, $|T| = s$, and the IRR holds with $\|A_{T^c}^* A_T (A_T^* A_T)^{-1}\|_\infty \le \alpha \le 1/s$. Then the following error bound holds for any solution $\widehat{x}_\delta$ of $\mathcal{P}_{1,\delta}$,*

$$\|\widehat{x}_\delta - x_0\|_1 \le \frac{2s(\epsilon+\delta)}{(1-\alpha s)} \|A_T (A_T^* A_T)^{-1}\|_1. \qquad (4)$$

In the special case where $k = j+1$, we have:

Corollary 1. *Let $k = j+1, |T| = s$, and for any $\sigma_1, \sigma_2 \in T$, the two cliques corresponding to σ_1 and σ_2 have overlaps no larger than $j-2$. Then we have $\|A_{T^c}^* A_T (A_T^* A_T)^{-1}\|_\infty \le 1/(j+1)$, and thus the following error bound for solution $\widehat{x}_\delta$ of $\mathcal{P}_{1,\delta}$ holds:*

$$\|\widehat{x}_\delta - x_0\|_1 \le \frac{2s(\epsilon+\delta)}{(j+1-s)} (j+1)^{3/2}, \qquad s < j+1.$$

4.5 Identifying Cliques with Mixed Sizes

In general settings, we need to identify high order cliques of mixed sizes, i.e., cliques of sizes $k_1, k_2, \cdots, k_l$ ($k_1 < k_2 < \cdots < k_l$), based on the observed data b on all j-subsets. One way to construct the basis matrix A is by concatenating $R^{j,k}$ with different k's satisfying $k > j$. We can then solve $\mathcal{P}_1$ and $\mathcal{P}_{1,\delta}$ for exact recovery and stable recovery with this newly concatenated basis matrix A. We have the following theorem:

Theorem 6. *Suppose x_0 is a sparse signal on cliques of sizes $k_1, k_2, \cdots, k_\ell (j \le k_1 < k_2 < \cdots < k_\ell \le k)$ and $b = Ax_0$. Let $T = \mathrm{supp}(x_0)$.*

1. If the cliques in T have no overlaps, then they can be identified by $\mathcal{P}_1$.

2. If the data $b = Ax_0 + z$ is contaminated by the noise z, $\mathcal{P}_{1,\delta}$ provides an estimate of x_0 for which the inequality in (4) still holds.

The above theorem provides us a sufficient condition to guarantee exact sparse recovery with concatenated bases and the stable recovery theory is also established.

5 Computational Algorithm

In practical applications, we often have pairwise interaction data in a network with n nodes and we wish to infer high order cliques up to size k. Directly constructing A by concatenating Radon basis matrices

$R^{j,j}, R^{j,j+1} \ldots, R^{j,k}$ and solving $\mathcal{P}_{1,\delta}$ would incur exponential complexity since A has exponentially many columns with respect to k. This would be intractable for inferring high order cliques in large networks. In this section, we describe a polynomial time (with respect to both n and k) approximation algorithm for solving $\mathcal{P}_{1,\delta}$. Recall that the primal and dual programs $\mathcal{P}_{1,\delta}$ and $\mathcal{D}_{1,\delta}$ are[3]:

$$(\mathcal{P}_{1,\delta}) \qquad \min \|x\|_1 \ \text{ s.t. } \|Ax - b\|_\infty \leq \delta$$
$$(\mathcal{D}_{1,\delta}) \qquad \max -\delta\|\gamma\|_1 - b^*\gamma \ \text{ s.t. } \|A^*\gamma\|_\infty \leq 1.$$

The key of our algorithm is that we use a polynomial number variables and constraints to approximate both programs, yielding an approximate solution for $\mathcal{P}_{1,\delta}$. More precisely, we apply a sequential primal-dual interior point method to solve the relaxed programs:

$$(\mathcal{P}_{1,\delta,\mathrm{T}}) \qquad \min \|x\|_1 \ \text{ s.t. } \|A_T x - b\|_\infty \leq \delta$$
$$(\mathcal{D}_{1,\delta,\mathrm{T}}) \qquad \max -\delta\|\gamma\|_1 - b^*\gamma \ \text{ s.t. } \|A_T^*\gamma\|_\infty \leq 1.$$

Here A_T is a submatrix of A where we extract a subset of columns T. We approximate the solution to the original programs by solving the above relaxed programs where we use polynomially many columns indexed by T. In particular, we want to find an interior point γ for $\mathcal{D}_{1,\delta,T}$ which is also feasible for $\mathcal{D}_{1,\delta}$. With this γ available, we can use duality gaps to check convergence because the current dual objective provides a lower bound for $\mathcal{D}_{1,\delta}$ and any interior point for $\mathcal{P}_{1,\delta,T}$ provides an upper bound for $\mathcal{P}_{1,\delta}$.

Let A_i be the i-th column of A. We need to sequentially update the column set T. When we have a solution γ (which is called the approximate analytic center) for the relaxed program $\mathcal{D}_{1,\delta,T}$, we need to find a new column A_i ($i \in T^c$) which is not feasible in $\mathcal{D}_{1,\delta,T}$. By incorporate A_i into T, the feasible region of $\mathcal{D}_{1,\delta,T}$ is reduced to better approximate that of $\mathcal{D}_{1,\delta}$. When the current solution γ has no violated constraint, i.e., γ is feasible for $\mathcal{D}_{1,\delta}$, we use interior point methods to find a series of interior points which converge to the solution of $\mathcal{D}_{1,\delta,T}$. However, we may obtain a new interior point γ which is not feasible for $\mathcal{D}_{1,\delta}$. We then need to go back and add violated constraints. A formal description is provided in Algorithm 1.

In Algorithm 1, the first IF statement involves a problem of finding a violated dual constraint for the current relaxed program. In the special case where γ are dual variables associated with edges, the problem becomes the *maximum edge weight clique problem*, which is known to be NP-hard. We use a simple greedy heuristic algorithm, which iteratively adds new nodes in order to maximize summation of edge weights to solve this problem [20], which runs in $\mathcal{O}(nk^2)$ time and can return a 0.94-approximate solution in the average case. Note that, if γ is feasible for the dual relaxation problem with no additional violated constraints, then 0.94γ must be feasible for $\mathcal{D}_{1,\delta}$ whose objective is discounted by 0.94. Thus, we will terminate with an 0.94-approximate solution.

[3]the proof that $\mathcal{D}_{1,\delta}$ is the dual of $\mathcal{P}_{1,\delta}$ trivially follows from the KKT conditions

Algorithm 1 Cutting Plane Method for Solving $\mathcal{P}_{1,\delta}$

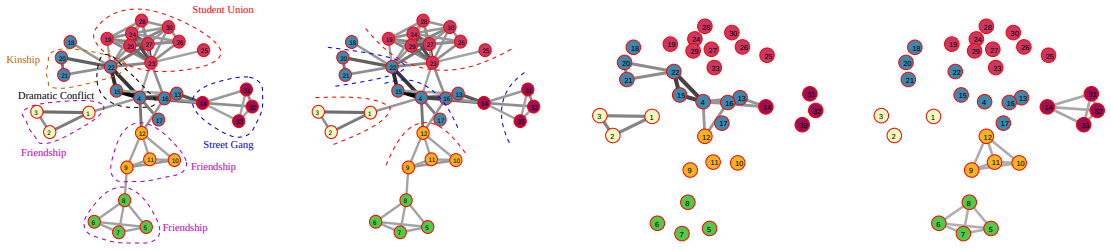
```
Initialize A = I, x = b, γ = (1, 1, ··· , 1)^t.
while TRUE do
  if ∃ |A_i^* γ| > 1 where i ∈ T^c then
    T ← T ∪ {i}, formulate new D_{1,δ,T} and P_{1,δ,T}.
    Find new interior points γ and x for D_{1,δ,T} and
    P_{1,δ,T} respectively.
  else if the duality gap is small then
    get the dual solution x̂ and stop.
  else
    find a new interior point γ for D_{1,δ,T}, which
    optimizes the dual objective.
  end if
end while
```

Let η be the threshold to check the duality gap. Algorithm 1 can also be understood as the column generation method [8], since adding a new inequality constraint in the dual program adds a variable to the primal program and thus adds a column to the basis matrix. For more details of the algorithm, see [21] and [28]. Theoretically, if one is able to find a violated constraint in constant time and uses interior point methods to locate approximate centers of the primal-dual feasible regions, then Algorithm 1 has computational complexity $\mathcal{O}(M/\eta^2)$, where M is the number of dual variables [21, 28]. In our case, $M \asymp \mathcal{O}(n^2)$ and find a violated constraint has complexity $\mathcal{O}(nk^2)$, thus algorithm 1 has complexity $\mathcal{O}(n^3k^2/\eta^2)$.

Finally, we note that other iterative algorithms, e.g., Bregman iterations, which have guaranteed convergence rates [3] can be used to find solutions of linear program relaxations in our algorithms. We also note that, in practice, we never need to explicitly construct the matrix A because there are many combinatorial structures within the basis matrix to exploit. For example, evaluating inner products between the bases can be efficiently estimated by directly comparing two sets.

6 Experimental Results

In this section, we demonstrate two examples of identifying communities in social networks. We compare our approach with the state-of-the-art clique percolation method. In these examples, we use the clique



(a) (b) (c) (d)

Cliques	Names of Characters	Relationships	Percolation	Radon Basis
$\{1, 2, 3\}$	{Myriel, Mlle Baptistine, Mme Magloire}	Friendship	N	N
$\{4, 13, 14\}$	{Valjean, Mme Thenardier, Thenardier}	Dramatic Conflicts	N	Y
$\{4, 15, 22\}$	{Valjean, Cosette, Marius}	Dramatic Conflicts	N	Y
$\{20, 21, 22\}$	{Gillenormand, Mlle Gillenormand, Marius}	Kinship	N	Y
$\{5, 6, 7, 8\}$	{Tholomyes, Listolier, Fameuil, Blacheville}	Friendship	Y	Y
$\{9, 10, 11, 12\}$	{Favourite, Dahlia, Zephine, Fantine}	Friendship	Y	Y
$\{14, 31, 32, 33\}$	{Thenardier, Gueulemer, Babet, Claquesous}	Street Gang	N	Y

Figure 1: *Les Misérables* social network. (a) Social network of characters in *Les Misérables*; (b) Spectral clustering result; (c) The identified 3-cliques; (d) The identified 4-cliques. The table summarizes the ground truth community of all the nodes.

volume and *conductance*, which arguably are the simplest evaluation criteria of clustering quality, to evaluate different algorithms. The clique volume is the sum of edge weights inside the clique, while the clique conductance is the ratio between the number of weights leaving the clique and the clique volume [19]. Let B be the adjacency matrix of a network. The *conductance* $\phi(S)$ of a set of nodes S is $\phi(S) = \frac{\sum_{\{(i,j):i\in S, j\notin S\}} B_{ij}}{\min(\mathrm{Vol}(S), \mathrm{Vol}(V\setminus S))}$ and *volume* is $\mathrm{Vol}(S) = \sum_{\{i,j\in S\}} B_{ij}$.

6.1 The Social Network of Les Misèrables

We consider the social network of 33 characters in Victor Hugo's novel *Les Misèrables* [16]. We represent this social network using a weighted graph (Figure 1-(a)). The edge weights are the co-appearance frequencies of the two corresponding characters. Figure 1-Table illustrates several social communities formed by relationships including *friendships, street gangs, kinships*, etc. The underlying social community, regarded as the ground truth for the data, is summarized in Figure 1-(a) where several social communities arise. Figure 1-(b) shows the spectral clustering result in which the first three red cuts are reasonable while the next three blue cuts destroyed a lot of community structures within the network.

We compare our method with the clique percolation method, 23 and 19 cliques were identified respectively where our approach can identify more meaningful cliques – see Figure 1-Table where we verified the ground truth from the novel. For example, our method can correctly identify two separate cliques $\{4, 15, 22\}$ and $\{20, 21, 22\}$, while the clique percolation method is treating $\{4, 15, 20, 21, 22\}$ as a single clique. The interaction frequencies among those characters, however, show that there are relatively smaller cross-community interactions, thus those two 3-cliques should be separated. Figure 1-(c) and (d) depict important 3 and 4 cliques identified by our algorithm. The sparsity patterns of those cliques satisfy the irrepresentable condition where overlaps between them are generally not large. However, they do not necessarily satisfy the condition in Lemma 1 which is based on a worst-case analysis. In Figure 2-(a-d), we also compare both methods in terms of clique conductances and volumes and see that cliques identified by Radon basis pursuit have slightly lower conductances and larger volumes, which demonstrates advantages of our approach.

In summary, our method obtains more abundant social structure information than the competing techniques. We also obtain social communities with overlaps which is impossible for clustering methods. We note that some simple schemes will not work well. For example, one may think of scoring each large clique by the mean scores of the included small cliques. In this example, since two or three key characters appear very frequently, we will end up with finding that the top high order cliques always contain them. In fact, among the top ten 3-cliques, seven of them contain node 4 and six of them contain node 15, which does not give us good results.

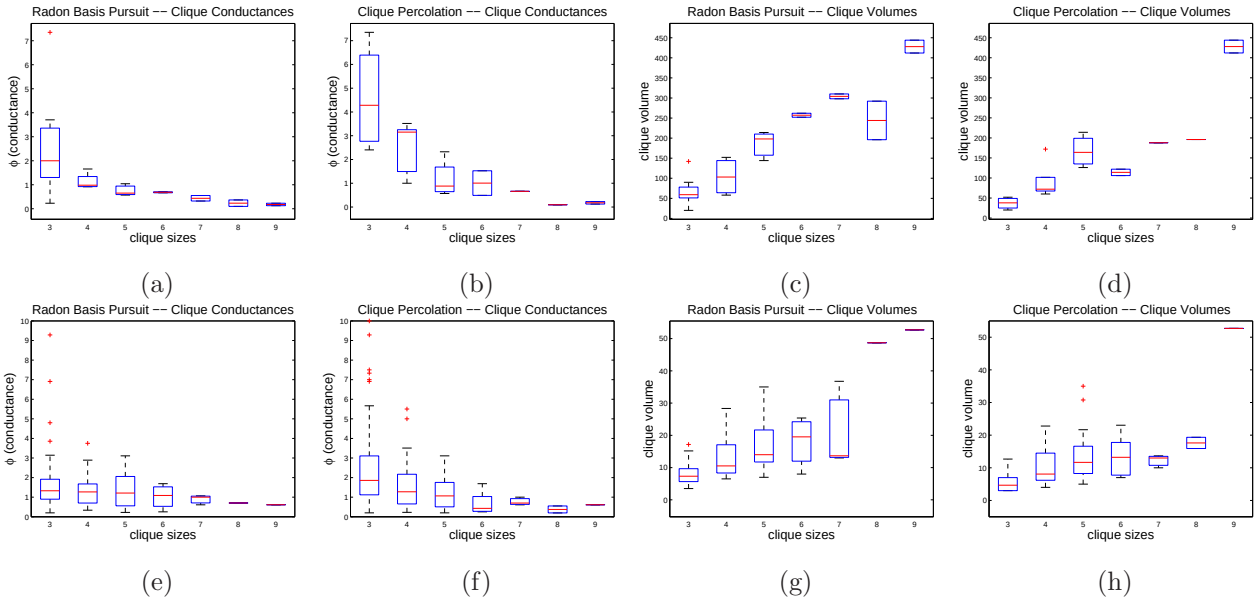


Figure 2: *Les Misérables* social network and coauthorship network: Box plot of clique conductances and volumes for our approach and the clique percolation method. Cliques identified by our approach have smaller conductances and larger volumes.

6.2 Coauthorships in Network Science

We also studied a larger coauthorship network where there is a total of 1589 scientists who come from a broad variety of fields. There are 136 and 166 cliques identified by our approach and the clique percolation method. In Figure 2-(e-h), we evaluate these two methods in terms of clique conductances and volumes. We see that the cliques identified by Radon basis pursuit have smaller conductances and comparable clique volumes than the clique percolation method. Our approach scales very well. In this example, it can identify the cliques up to size 9 in 564 seconds.

Finally, we note that clustering techniques, e.g., spectral clustering which does not allow overlaps, cannot provide abundant social community information. When we ran the bipartite spectral clustering on the data, many community structures were destroyed. Another alternative approach might be to simply score each large clique by the mean scores of the included small cliques. However, this approach is not robust with respect to super-nodes (those with many edges with large weights). Such nodes will incorrectly generate many high order cliques with large scores.

7 Conclusions

We studied the network clique detection problem in this paper by introducing a new network data representation framework. Such a novel framework allows us to explore and analyze network data guided by more rigorous theory coming from the compressed sensing literature. Instead of providing just another heuristic method, we aim at contributing at the foundational level to network data analysis. We hope that our work could build a bridge connecting the research communities of network modeling and compressed sensing, so that research results and tools from one area could be ported to another one to create more exciting results.

Acknowledgements

The authors would like to thank Zongming Ma, Minyu Peng, Michael Saunders, Yinyu Ye, and the anonymous reviewers for very helpful dicussions and comments. Thanks also to Kyle Heath, Qixing Huang, and Fan Wang for reading this paper and giving their valuable feedback. Xiaoye Jiang and Leonidas Guibas wish to acknowledge the support of ARO grants W911NF-10-1-0037 and W911NF-07-2-0027, as well as NSF grant CCF 1011228 and a gift from the Google Corporation. Yuan Yao acknowledges supports from the National Basic Research Program of China (973 Program 2011CB809105, 2012CB825501), NSFC (61071157), Microsoft Research Asia, and a professorship in the Hundred Talents Program at Peking University. Han Liu is thankful for the support of NSF grant IIS-1116730 and a faculty supporting package from Johns Hopkins University.

References

[1] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, October 1999.

[2] P. J. Bickel and A. Chen. A nonparametric view of network models and newmangirvan and other modularities. *Proceedings of National Academy of Sciences of the United States of America*, 106(50):21068–21073, December 2009.

[3] J. Cai, S. Osher, and Z. Shen. Linearized bregman iterations for compressed sensing. *Mathematics of Computation*, 78(267):1515–1536, 2009.

[4] E. J. Candès. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus de l'Académie des Sciences, Paris, Série I*, 346:589–592, 2008.

[5] E. J. Candès and T. Tao. Decoding by linear programming. *IEEE Transaction on Information Theory*, 51:4203–4215, 2005.

[6] E. J. Candès and T. Tao. The dantzig selector: statistical estimation when p is much larger than n. *Annals of Statistics*, 35(6):2313–2351, 2007.

[7] S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61, 1999.

[8] G. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations Research*, 8:101–111, 1960.

[9] P. Diaconis. *Group Representations in Probability and Statistics*. Institute of Mathematical Statistics, 1988.

[10] P. Erdös and A. Rényi. On random graphs, i. *Publicationes Mathematicae*, 6:290–297, 1959.

[11] P. Erdös and A. Rényi. On the evolution of random graphs. *Publication of the Mathematical Institue of the Hungrian Academy of Science*, 5:17–61, 1960.

[12] A. Goldenberg, A. X. Zheng, S. E. Fienberg, and E. M. Airoldi. A survey of statistical network models. *Foundations and Trends in Machine Learning*, 2, 2010.

[13] L. J. Guibas. The identity management problem — a short survey. In *International Conference on Information Fusion*, 2008.

[14] P. D. Hoff, A. E. Raftery, M. S. Handcock, and M. S. H. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97:1090–1098, 2001.

[15] S. Jagabathula and D. Shah. Inferring rankings under constrained sensing. In *Neural Information Processing Systems (NIPS)*, 2008.

[16] D. E. Knuth. *The Stanford GraphBase: A Platform for Combinatorial Computing*. Addison-Wesley, 1993.

[17] I. Kovács, R. Palotai, M. Szalay, and P. Csermely. Community landscapes: a novel, integrative approach for the determination of overlapping network modules. *PLoS ONE*, 5(9):e12528, September 2010.

[18] A. Lancichinetti and S. Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, 80(1):16118, 2009.

[19] J. Leskovec, K. Lang, and M. Mahoney. Empirical comparison of algorithms for network community detection. In *ACM WWW International Conference on World Wide Web (WWW)*, 2010.

[20] G. S. Lueker. Maximization problems on graphs with edge weights chosen from a normal distribution. In *ACM Symposium on Theory of Computing*, pages 13–18, 1978.

[21] J. E. Mitchell. Polynomial interior point cutting plane methods. *Optimization Methods and Software*, 18:2003, 2003.

[22] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of National Academy of Sciences*, 103(23):8577–8582, 2006.

[23] G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814, 2005.

[24] P. Sarkar and A. Moore. Dynamic social network analysis using latent space models. *SIGKDD Explorations: Special Edition on Link Mining*, 2005.

[25] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.

[26] Y. Tsaig and D. L. Donoho. Compressed sensing. *IEEE Transaction on Information Theory*, 52:1289–1306, 2006.

[27] S. Wasserman and C. Anderson. Stochastic a posterior blockmodels: Construction and assessment. *Social Networks*, 9(1):1–36, 1987.

[28] Y. Ye. *Interior Point Algorithms: Theory and Analysis*. Wiley, 1997.

[29] M. Yuan and Y. Lin. On the nonnegative garrote estimator. *Journal of the Royal Statistical Society. Series B*, 69(2):143–161, 2007.

[30] P. Zhao and B. Yu. On model selection consistency of lasso. *Journal of Machine Learning Reserach*, 7:2541–2563, 2006.