SPECTRAL TECHNIQUES FOR PARTITIONING PLANTED RANDOM GRAPHS

YUFEI ZHAO

1. Introduction

This my final project paper for the class 18.S996: Algorithmic Aspects of Machine Learning, taught in Fall 2013 by Prof. Ankur Moitra. In this note, I will explain the spectral method for partitioning a graph as used in the following two papers:

[1] N. Alon, M. Krivelevich, and B. Sudakov, *Finding a large hidden clique in a random graph*.

[4] F. McSherry, *Spectral partitioning of random graphs*.

In [1], the problem is to recover a planted clique of size $c\sqrt{n}$ in a random graph $G(n, 1/2)$. That is, we first generate a random graph $G(n, 1/2)$, and then randomly put in a clique of size $c\sqrt{n}$, and the algorithmic goal is to recover this clique.

In [4], variations and extensions of this planted clique problem are considered, including planted bisection and k-coloring. In each case, the graph is generated by first partitioning the vertices into parts of prescribed size (the algorithm doesn't know which vertices belong to which partition) and then edges are places between parts and within parts using certain prescribed probabilities. The goal is to recover the original partition given the resulting random graph. In [4], McSherry considers probabilities that depend on n, the size of graph, whereas in [1], only the constant $1/2$ probability is considered.

These problems are all variations of classical NP-hard optimization problems which are even hard to approximate. However, the planted versions turn out to be more amenable, at least in certain parameter regimes. Note that our goal is to recover the planted object/partition, and not necessarily to solve the optimization problem (e.g., finding the largest clique), although these two goals can coincide (as they do in the case of planted clique, since the largest clique in $G(n, 1/2)$ has size $(2 + o(1)) \log_2 n$).

I will give only a very rough sketch of the proof ideas, without giving much details or calculations.

2. Eigenvalues of a random graph

We first review some facts about eigenvalues of a random graph, which are due to Füredi and Komlós [3].

Theorem 2.1. *Let A be an $n \times n$ random symmetric matrix with bounded independent entries (other than the constraint $A_{ij} = A_{ji}$). Suppose all the non-diagonal entries have mean zero and variance σ^2. Let $\lambda_1(A) \geq \cdots \geq \lambda_n(A)$ be the eigenvalues of A. Then w.h.p. (i.e., with probability tending to 1)*

$$\max_{1 \leq i \leq n} |\lambda_i(A)| = 2\sigma\sqrt{n} + O(n^{1/3} \log n).$$

Date: November 22, 2013.

We sketch the idea of the proof. Following the techniques in the proof of Wigner's famous semicircle law for the distribution of eigenvalues of a random matrix [5], we first compute the moments of the spectrum of A:

$$\sum_{i=1}^{n} \lambda_i(A)^k = \operatorname{tr} A^k$$

The quantity $\operatorname{tr} A^k$ is the sum of all weighted closed walks on n vertices with edge weights given by A. We find that

$$\mathbb{E} \operatorname{tr} A^k = \mathbb{E} \sum_{i_0, i_1, \ldots, i_k \in [n]} A_{i_0 i_1} A_{i_1 i_2} \cdots A_{i_{k-1} i_k}. \tag{1}$$

Since A has independent mean-zero nondiagonal entries, any term where some A_{ij} (also counting A_{ji}) appears exactly once has zero expectation. One then enumerates over all possible walks where every edge is traversed at least twice, and it turns out, when k is even, the dominant contributions to (1) come from walks of length k on a tree where every edge is traversed exactly twice. This corresponds to terms in the sum on the RHS of (1) where the set $\{i_0, i_1, \ldots, i_k\}$ has size exactly $k/2+1$. The contribution of these terms to the expectation is exactly

$$\frac{1}{(k/2)+1} \binom{k}{k/2} n(n-1) \cdots \left(n - \frac{k}{2}\right) \sigma^k.$$

Here $\frac{1}{(k/2)+1}\binom{k}{k/2}$ is the $(k/2)$-th Catalan number, which counts (among many other things) the number of possible tree walks up to isomorphism, and the factor $n(n-1)\cdots(n-(k/2))$ is the number of ways of assigning vertices of the graph to the vertices of the tree walk (i.e., injectively mapping $[n]$ to $\{i_0, i_1, \ldots, i_k\}$), and last factor σ^k comes from $\mathbb{E} A_{ij}^2 = \sigma^2$ when $i \neq j$.

It is then shown that the other terms contribute negligibly to (1), at least when k grows not too quickly with n:

$$\mathbb{E} \operatorname{tr} A^k = \left(1 + O\left(\frac{k^6}{n}\right)\right) \frac{1}{(k/2)+1} \binom{k}{k/2} n(n-1) \cdots \left(n - \frac{k}{2}\right) \sigma^k, \qquad k \text{ even}.$$

(One difference between the proof here and Wigner's proof is that Wigner considers fixed k and lets $n \to \infty$, where as here we need to choose a k that grows with n.) To obtain a concentration result, we use Markov's inequality. Using an even integer $k = O(n^{1/6})$ and $v = O(n^{1/3} \log n)$ we get

$$\begin{aligned}
\mathbb{P}(\max |\lambda_i(A)| > 2\sigma\sqrt{n} + v) &= \mathbb{P}(\max |\lambda_i(A)|^k > (2\sigma\sqrt{n} + v)^k) \\
&\leq (2\sigma\sqrt{n} + v)^{-k} \mathbb{E} \max |\lambda_i(A)|^k \\
&\leq (2\sigma\sqrt{n} + v)^{-k} \mathbb{E} \operatorname{tr} A^k \\
&\leq (2\sigma\sqrt{n} + v)^{-k} n^{(k/2)+1} 2^k \sigma^k \\
&= n \left(1 + \frac{v}{2\sigma\sqrt{n}}\right)^{-k} \\
&\leq n \exp\left(-\frac{kv}{2\sigma\sqrt{n}}\right)
\end{aligned}$$

So

$$\max_{1\le i\le n} |\lambda_i(A)| \le 2\sigma\sqrt{n} + O(n^{1/3}\log n)$$

with probability at least $1 - n^{-10}$. One can also deduce a similar lower bound to $\max|\lambda_i|$, thereby yielding the theorem.

Using Theorem 2.1 we can deduce a bound on the second eigenvalue of a random graph.

Corollary 2.2. *Let $\lambda_1 \ge \cdots \ge \lambda_n$ be the eigenvalues of the adjacency matrix of the random graph $G(n, 1/2)$. Then w.h.p. we have*

$$\max_{i\ge 2} |\lambda_i| \le \sqrt{n} + O(n^{1/3}\log n).$$

This Corollary follows from Theorem 2.1 as an immediate consequence of the following lemma which allows us to zero-out the mean of the matrix entries.

Lemma 2.3. *Let A be a real symmetric matrix, and $B = A - tJ$ (where J is the matrix with all 1's and $t \in \mathbb{R}$). Then $\lambda_2(A) \le \lambda_1(B)$.*

Proof. By the Courant-Fisher theorem

$$\lambda_2(A) = \min_{\mathbf{v}} \max_{\substack{\langle \mathbf{x},\mathbf{v}\rangle = 0 \\ \|\mathbf{x}\|=1}} \mathbf{x}^T A\mathbf{x} \le \max_{\substack{\langle \mathbf{x},\mathbf{1}\rangle = 0 \\ \|\mathbf{x}\|=1}} \mathbf{x}^T A\mathbf{x} = \max_{\substack{\langle \mathbf{x},\mathbf{1}\rangle = 0 \\ \|\mathbf{x}\|=1}} \mathbf{x}^T (B+tJ)\mathbf{x} = \max_{\substack{\langle \mathbf{x},\mathbf{1}\rangle = 0 \\ \|\mathbf{x}\|=1}} \mathbf{x}^T B\mathbf{x} \le \lambda_1(B). \quad \blacksquare$$

3. Planted clique

Now we describe the Alon-Krivelevich-Sudakov [1] (abbreviated as AKS) algorithm for finding a planted clique of size $\Omega(\sqrt{n})$ in $G(n, 1/2)$. We deal with the case when the planted clique has size at least $10\sqrt{n}$. This is the most interesting case and it uses spectral methods.

Once we know how to deal with planted cliques of size at least $10\sqrt{n}$, we can bootstrap to obtain an algorithm to find a planted clique of size $c\sqrt{n}$ for any $c > 0$: one checks over all subsets of vertices of size $O_c(1)$ and then run the previous algorithm on their common neighborhood. The point is that sampling the common neighborhood boosts the density of the planted clique as a subset of the ambient graph.

Note that if the planted clique has size $k = \Omega(\sqrt{n\log n})$ then with high probability the k vertices of the highest degree must necessarily be the clique. So this case is easy.

From now on assume that the planted clique has size $k \ge 10\sqrt{n}$. The idea is look at the top eigenvectors of the adjacency matrix of G. While the top eigenvector is very close to the all 1's vector (this is true for $G(n, 1/2)$, and adding a clique won't perturb it by much), the second eigenvector encodes the planted clique. Indeed, the planted clique of size $10\sqrt{n}$ causes the second eigenvalue to be much larger than all but the top eigenvalue, and the clique roughly corresponds to the coordinates with large entries in the second eigenvector.

AKS algorithm for finding a planted clique

Input: Graph G generated from $G(n, 1/2)$ with a randomly planted clique of size $k \ge 10\sqrt{n}$.

1. Find the second eigenvector v_2 of the adjacency matrix of G.
2. Let W be the vertices corresponding to k largest coordinates (in absolute value) in v_2. Let Q be the vertices in G that have at least $3k/4$ neighbors in W.

Output: Q.

In order for this algorithm to work, we need to know that the eigenvalues of G are well separated. It is proved that w.h.p. the eigenvalues of the adjacency matrix satisfy

(1) $\lambda_1 \geq (1/2 + o(1))n$
(2) $4\sqrt{n} \leq \lambda_2 \leq 6\sqrt{n}$
(3) $|\lambda_i| \leq (1 + o(1))\sqrt{n}$ for all $i \geq 3$.

For (i), we know by the operator norm interpretation that λ_1 is at least the average degree. For (iii), we use Corollary 2.2. Indeed, let $G_2 = G(k, 1/2)$ be the random graph on Q, the vertices of the planted clique. Then $G_1 = G - G_2$ is distributed as a $G(n, 1/2)$. Let u_1 and u_2 be the principal eigenvectors of the adjacency matrices of G_1 and G_2 respectively. Then Corollary 2.2 tells us that the operator norm of the adjacency matrix of G, when restricted to the subspace orthogonal to both u_1 and u_2, is at most $(1+o(1))(\sqrt{n}+\sqrt{k}) = (1+o(1))\sqrt{n}$. This proves (iii).

The proof of (ii) is a bit more involved and we just sketch the idea here. Essentially one shows that the second eigenvector v_2 of G is very close to the vector z_2 whose coordinates are $n - k$ on the clique and k off the clique. From this one obtains that $(\lambda_2 - k/2)^2 \leq n/2$. So (ii) holds since $k \geq 10\sqrt{n}$. Additionally, the observation that $z_2 - v_2$ is small also shows that the k largest coordinates in v_2 correspond to the planted clique with only a small number of mistakes, which are corrected in the last step of the algorithm by counting neighbors.

4. McSherry's algorithm

McSherry [4] considers the following generalization of the planted clique problem.

Random graph model: partition the vertices into k classes. Let P be a $k \times k$ matrix with entries $P_{ij} \in [0, 1]$. Put in an edge between a vertex of class i and a vertex of class j with probability P_{ij}.

The input to the algorithm is the resulting random graph, and the goal is to recover the original vertex partition.

Here are some special cases of McSherry's results (we omit his more general result):

- In the planted multisection problem with k parts, where intra-part edges are included with probability q and inter-part edges are included with probability p, with $p < q$, the algorithm can recover the multisection if
$$\frac{q-p}{q} > \Omega\left(\frac{\log n}{qn}\right).$$
- In the planted k coloring problem, where the size of each color class is linear in n, and intra-part edges are included with probability p, the algorithm can recover the coloring if
$$p > \Omega\left(\frac{\log^3 n}{n}\right).$$
- In the planted clique problem where the clique has size s, and intra-part edges are included with probability p, the algorithm can recover the clique if
$$\frac{1-p}{p} > \Omega\left(\frac{n}{s^2} + \frac{\log n}{s}\right).$$

We refer to weighted graphs and symmetric matrices interchangeably. Following the notation used in [4], we use G to denote the probability matrix for the random graph model, i.e., the entries of G are the probabilities of generating an edge. The rank of G is at most k. Let $\widehat{G}$ be the random graph (i.e., $\{0, 1\}$-valued matrix) generated from this distribution.

Here is an informal description of the algorithm.

McSherry's spectral partitioning algorithm

Input: A graph $\widehat{G}$ generated from the random graph model G.

1. Randomly divide the vertices into two parts. Let this division split the columns of the matrix as $\widehat{G} = [\widehat{A} \mid \widehat{B}]$.
2. Let $\widehat{H} = [P_{\widehat{B}}(\widehat{A})|P_{\widehat{A}}(\widehat{B})]$, where $P_{\widehat{A}}$ is the projection onto the subspace[1] spanned by the most significant k eigenvectors of $\widehat{A}$ and similarly with $P_{\widehat{B}}$. We write $P_{\widehat{B}}(\widehat{A})$ to mean $P_{\widehat{B}}$ acting column-by-column on $\widehat{A}$.
3. Cluster the columns of $\widehat{H}$ greedily based on the distance between the columns (put two columns in the same block if their distance is at most some threshold τ).

Output: The partition of vertices generated in the last step of the algorithm.

We sketch an outline for the analysis of the algorithm. For each vertex u, let A_u denote the column of A corresponding to u, and similarly with $\widehat{A}$. We want to show that

$$\|A_u - P_{\widehat{B}}\widehat{A}_u\|$$

is small. If u and v belong to the same part of the vertex partition, then $A_u = A_v$, and furthermore, we assume in the model that if u and v belong to different parts, then A_u and A_v are appropriately separated. If $\|A_u - P_{\widehat{B}}\widehat{A}_u\|$ is always small, then the columns of $\widehat{H}$ are close to the columns of G and the clustering step works.

So it remains to show that $\|A_u - P_{\widehat{B}}\widehat{A}_u\|$ is small. By the triangle inequality,

$$\|A_u - P_{\widehat{B}}\widehat{A}_u\| \le \|A_u - P_{\widehat{B}}A_u\| + \|P_{\widehat{B}}(A_u - \widehat{A}_u)\|. \tag{2}$$

Here is a useful lemma.

Lemma 4.1. *For any matrix $\widehat{B}$ and rank k matrix B, one has*

$$\|(P_{\widehat{B}} - I)B\| \le 2\|\widehat{B} - B\|.$$

Here $\|\cdot\|$ denotes the operator norm (equivalently the spectral norm) of a matrix.

Proof. Write

$$(P_{\widehat{B}} - I)B = (P_{\widehat{B}} - I)\widehat{B} + (P_{\widehat{B}} - I)(B - \widehat{B}).$$

For the first term, as $P_{\widehat{B}}\widehat{B}$ is the optimal rank k approximation of $\widehat{B}$, we have

$$\|(P_{\widehat{B}} - I)\widehat{B}\| = \min_{\mathrm{rk}\, M \le k} \|M - \widehat{B}\| \le \|B - \widehat{B}\|.$$

For the second term, since $P_{\widehat{B}}$ is a projection, so is $I - P_{\widehat{B}}$, so

$$\|(P_{\widehat{B}} - I)(B - \widehat{B})\| \le \|B - \widehat{B}\|.$$

Combining the two inequality and we obtain the desired result. ■

Furthermore, Theorem 2.1 essentially tells us that $\|B - \widehat{B}\|$ should be small. It should be at most $O(\sigma\sqrt{n})$. In fact, there is also a concentration result for $\|B - \widehat{B}\|$ proved in [2] using Talagrand concentration inequality.

To bound the first term on the RHS of (2), note that

$$A_u - P_{\widehat{B}}A_u = (I - P_{\widehat{B}})BB^+A_u,$$

[1]McSherry calls this a *spectral projection*, and actually, he needs a variation that calls a *combinatorial projection*. I present the spectral projection version here since it's easier and conceptually simpler. McSherry proves that the combinatorial projection works and he conjectures that the spectral projection also works.

where A^+ is the pseudoinverse. By Lemma 4.1 and the observation in the previous paragraph, $(I - P_{\widehat{B}})B$ is small. We can also bound $\|B^+ A_u\|$. It then follows that $\|A_u - P_{\widehat{B}} A_u\|$ must be small.

For the second term on the RHS of (2), since we split up the columns in Step 1 in the algorithm, $\widehat{A}$ and $\widehat{B}$ are sampled independently. We can then apply a concentration inequality[2] to bound the size of $\|P_{\widehat{B}}(A_u - \widehat{A}_u)\|$.

It then follows that (2) is small, which validates McSherry's algorithm.

References

[1] N. Alon, M. Krivelevich, and B. Sudakov, *Finding a large hidden clique in a random graph*, Random Structures Algorithms **13** (1998), 457–466.

[2] N. Alon, M. Krivelevich, V. H. Vu, *On the concentration of eigenvalues of random symmetric matrices*, Israel J. Math. **131** (2002), 259-267.

[3] Z .Füredi, J. Komlś, *The eigenvalues of random symmetric matrices*, Combinatorica **1** (1981), 233–241.

[4] F. McSherry, *Spectral partitioning of random graphs*, IEEE Symposium on Foundations of Computer Science, 2001.

[5] E. P. Wigner, *On the distribution of the roots of certain symmetric matrices,* Ann. of Math. (2) **67** (1958), 325–327.

[2]When the probabilities are of constant order (i.e., not decreasing with n), one can use Azuma's inequality to bound $\|P_{\widehat{B}}(A_u - \widehat{A}_u)\|$, as

$$\|P_{\widehat{B}}(A_u - \widehat{A}_u)\|^2 = \sum_{j=1}^{k} \langle (A_u - \widehat{A}_u), q_j \rangle^2$$

where $q_1, \dots, q_k$ are the first k left-singular vectors of $\widehat{A}$. When the probabilities in the random graph model decrease with n, McSherry conjectures that $|P_{\widehat{B}}(A_u - \widehat{A}_u)|$ should still be small (at an order commensurate with the probabilities), but he does not have a proof. Azuma's concentration inequality doesn't give good enough bounds, so we want to apply a Chernoff bound instead. Chernoff bounds can only be applied to a sum of independent summands, and this is why McSherry considers a variation of the spectral projection algorithm, namely he gives a combinatorial projection algorithm that projects the columns of $\widehat{G}$ to subspaces spanned by characteristic vectors. He then uses Chernoff to bound the size of this combinatorial projection.