Scribed by Neng Huang

Lecture 11

In which we use the SDP relaxation of the infinity-to-one norm and Grothendieck inequality to give an approximation reconstruction of the stochastic block model.

1 A Brief Review of the Model

First, let's briefly review the model. We have a random graph $G = (V, E)$ with an unknown partition of the vertices into two equal parts V_1 and V_2. Edges across the partition are generated independently with probability q, and edges inside the partition are generated independently with probability p. To abbreviate the notation, we let $a = pn/2$, which is the average internal degree, and $b = qn/2$, which is the average external degree. Intuitively, the closer are a and b, the more difficult it is to reconstruct the partition. We assume $a > b$, although there are also similar results in the complementary model where b is larger than a. We also assume $b > 1$ so that the graph is not almost empty.

We will prove the following two results, the first of which will be proved using Grothendieck inequality.

1. For every $\epsilon > 0$, there exists a constant c_ϵ such that if $a - b > c_\epsilon \sqrt{a+b}$, then we can reconstruct the partition up to less than ϵn misclassified vertices.
2. There exists a constant c such that if $a - b \geq c\sqrt{\log n}\sqrt{a+b}$, then we can do exact reconstruct.

We note that the first result is essentially tight in the sense that for every $\epsilon > 0$, there also exists a constant c'_ϵ such that if $a - b < c'_\epsilon \sqrt{a+b}$, then it will be impossible to reconstruct the partition even if an ϵ fraction of misclassified vertices is allowed. Also, the constant c_ϵ will go to infinity as ϵ goes to 0, so if we want more and more accuracy, $a - b$ needs to be a bigger and bigger constant times $\sqrt{a+b}$. When the constant becomes $O(\sqrt{\log n})$, we will get an exact reconstruction as stated in the second result.

2 The Algorithm

Our algorithm will be based on semi-definite programming. Intuitively, the problem of reconstructing the partition is essentially the same as min-bisection problem, which is to find a balanced cut with the fewest edges. This is because the balanced cut with the fewest expected edges is exactly our hidden cut. Unfortunately, the min-bisection problem is **NP**-hard, so we will use semi-definite programming. The min-bisection problem can be stated as the following program:

$$\begin{aligned} \text{minimize} \quad & \sum_{(u,v)\in E} \frac{1}{4}(x_u - x_v)^2 \\ \text{subject to} \quad & x_v^2 = 1, \forall v \in V \\ & \sum_{v\in V} x_v = 0. \end{aligned}$$

Its semi-definite programming relaxation will be

$$\begin{aligned} \text{minimize} \quad & \sum_{(u,v)\in E} \frac{1}{4}\|\mathbf{x}_u - \mathbf{x}_v\|^2 \\ \text{subject to} \quad & \|\mathbf{x}_v\|^2 = 1, \forall v \in V \\ & \|\sum_{v\in V} \mathbf{x}_v\|^2 = 0. \end{aligned} \tag{1}$$

Our algorithm will be as follows.

- Solve the semi-definite programming above.
- Let $\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*$ be the optimal solution and $X = (X_{ij})$ such that $X_{ij} = \langle \mathbf{x}_i^*, \mathbf{x}_j^* \rangle$.
- Find $\mathbf{z} = (z_1, \ldots, z_n)$, which is the eigenvector corresponding to the largest eigenvalue of X.
- Let $S = \{i : z_i > 0\}$, $V - S = \{i : z_i \leq 0\}$.
- Output $(S, V - S)$ as our partition.

Ideally, we want half of the $\mathbf{x}_i^*$'s pointing to one direction, and the other half pointing to the opposite direction. In this ideal case we will have

$$X = \left(\begin{array}{c|c} \mathbf{1} & \mathbf{-1} \\ \hline \mathbf{-1} & \mathbf{1} \end{array}\right) = \begin{pmatrix} 1 \\ \vdots \\ 1 \\ -1 \\ \vdots \\ -1 \end{pmatrix} (1, \ldots, 1, -1, \ldots, -1).$$

Then X will be a rank-one matrix and $(1, \ldots, 1, -1, \ldots, -1)^T$, which is the indicator vector of the hidden cut, will be its eigenvector with eigenvalue n. The remaining eigenvalues of X will be all zeros. So finding the largest eigenvector of X will reveal the hidden cut. In reality, if $a-b > c_\epsilon\sqrt{a+b}$, then our solution will be almost the same as that in the ideal case, so the cut we get will be almost the same as the hidden cut. Furthermore, if $a-b \geq c\sqrt{\log n}\sqrt{a+b}$, then the unique optimal solution of the SDP will be the combinatorial solution of min-bisection problem, that is, in the vector language, the one-dimensional solution.[1]

3 Analysis of the Algorithm

First, we rearrange the SDP to make it slightly simpler. We have the following SDP:

$$\begin{aligned} \text{maximize} \quad & \sum_{u,v} A_{uv}\langle \mathbf{x}_u, \mathbf{x}_v\rangle \\ \text{subject to} \quad & \|\mathbf{x}_v\|^2 = 1, \forall v \in V \\ & \|\sum_{v\in V} \mathbf{x}_v\|^2 = 0. \end{aligned} \tag{2}$$

We note that SDP?? and SDP?? have the same optimal solution, because the cost function of SDP?? is

$$\begin{aligned} & \sum_{(u,v)\in E} \frac{1}{4}\|\mathbf{x}_u - \mathbf{x}_v\|^2 \\ =& \sum_{u,v} \frac{1}{4} A_{uv}\|\mathbf{x}_u - \mathbf{x}_v\|^2 \\ =& \sum_{u,v} \frac{1}{4} A_{uv}(2 - \langle \mathbf{x}_u, \mathbf{x}_v\rangle) \\ =& (\sum_{u,v} \frac{1}{2} A_{uv}) - \frac{1}{4}(\sum_{u,v} A_{uv}\langle \mathbf{x}_u, \mathbf{x}_v\rangle). \end{aligned}$$

The first term is a constant and the second is the cost function of SDP?? with a factor of -1/4.

Now, consider the cost of SDP?? of $\chi = (\chi_v)_{v\in V}$ where

$$\chi_v = \begin{cases} +1 & \text{if } v \in V_1, \\ -1 & \text{if } v \in V_2. \end{cases}$$

The expected cost will be

$$\mathop{\mathbb{E}}_{\text{choice of graph}} (\text{cost of } \chi) = \frac{n(n-1)}{2}p - \frac{n^2}{2}q = n(a-b) - a.$$

[1] "A miracle", said Luca.

Since each edge is chosen independently, with high probability our cost will be at least $n(a-b) - a - \sqrt{n(a+b)} \geq n(a-b) - O(n)$, which implies that the optimal solution of SDP?? will be at least $n(a-b) - O(n)$. Let $\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*$ be the optimal solution of the SDP, then we have

$$\begin{aligned} n(a-b) - O(n) &\leq cost(\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*) \\ &= \sum_{u,v} A_{uv} \langle \mathbf{x}_u^*, \mathbf{x}_v^* \rangle \\ &= \sum_{u,v} \left(A_{uv} - \frac{a+b}{n} \right) \langle \mathbf{x}_u^*, \mathbf{x}_v^* \rangle \end{aligned} \tag{3}$$

In the last equality we used the fact that $\sum_{u,v} \langle \mathbf{x}_u^*, \mathbf{x}_v^* \rangle = \| \sum_u \mathbf{x}_u^* \|^2 = 0$

When we used the spectral method last week, we said that the largest eigenvalue of $A - \frac{d}{n} J$ is large, where d is the average degree. This is because the hidden cut will give us a vector with large Rayleigh quotient. But $A - \mathbb{E}\, A$ has a relatively small spectral norm, so everything should come from $\mathbb{E}\, A - \frac{d}{n} J$, which when simplified will be 1 for entries representing vertices on the same side and -1 for entries representing vertices on different sides. We will redo this argument with SDP norm in place of spectral norm and every step appropriately adjusted.

Recall that the SDP norm of a matrix M is defined to be

$$\|M\|_{SDP} := \max_{\substack{|\mathbf{x}_1|=\cdots=|\mathbf{x}_n|=1 \\ |\mathbf{y}_1|=\cdots=|\mathbf{y}_n|=1}} \sum_{u,v} M_{uv} \langle \mathbf{x}_u, \mathbf{y}_v \rangle.$$

Let $R = \left(\begin{array}{c|c} \mathbf{p} & \mathbf{q} \\ \hline \mathbf{q} & \mathbf{p} \end{array} \right)$, then by Grothendieck inequality we have

$$\|A - R\|_{SDP} \leq c \|A - R\|_{\infty \to 1}.$$

We proved in the previous lecture that $\|A - R\|_{\infty \to 1} \leq O(n\sqrt{a+b})$ with high probability, so we know that the SDP norm $\|A - R\|_{SDP} \leq O(n\sqrt{a+b})$ with high probability as well. By definition, this means

$$\sum_{u,v} (A_{uv} - R_{uv}) \langle \mathbf{x}_u^*, \mathbf{x}_y^* \rangle \leq \|A - R\|_{SDP} \leq O(n\sqrt{a+b}). \tag{4}$$

Substracting ?? from ??, we obtain

$$\sum_{u,v} \left(A_{uv} - \frac{a+b}{n} - A_{uv} + R_{uv} \right) \langle \mathbf{x}_u^*, \mathbf{x}_y^* \rangle \geq n(a-b) - O(n\sqrt{a+b}). \tag{5}$$

Observe that

$$R = \left(\begin{array}{c|c} \mathbf{p} & \mathbf{q} \\ \hline \mathbf{q} & \mathbf{p} \end{array} \right) = \frac{p+q}{2} J + \frac{p-q}{2} C = \frac{a+b}{n} J + \frac{a-b}{n} C, \tag{6}$$

where J is the all-one matrix and $C = \left(\begin{array}{c|c} \mathbf{1} & \mathbf{-1} \\ \hline \mathbf{-1} & \mathbf{1} \end{array}\right)$. Plugging **??** into **??**, we get

$$\sum_{u,v} \frac{a-b}{n} C_{uv} \langle \mathbf{x}_u^*, \mathbf{x}_y^* \rangle \geq n(a-b) - O(n\sqrt{a+b}),$$

which can be simplified to

$$\sum_{u,v} C_{uv} \langle \mathbf{x}_u^*, \mathbf{x}_y^* \rangle \geq n^2 \left(1 - \frac{O(\sqrt{a+b})}{a-b}\right).$$

For simplicity, in the following analysis the term $\frac{O(\sqrt{a+b})}{a-b}$ will be called $1/c$. Notice that C is a matrix with 1 for nodes from the same side of the cut and -1 for nodes from different sides of the cut, and $\langle \mathbf{x}_u^*, \mathbf{x}_y^* \rangle$ is an inner product of two unit vectors. If $1/c$ is very close to zero, then the sum will be very close to n^2. This means that $C_{uv} \langle \mathbf{x}_u^*, \mathbf{x}_y^* \rangle$ should be 1 for almost every pair of (u, v), which shows that $X = \langle \mathbf{x}_u^*, \mathbf{x}_y^* \rangle$ is actually very close to C. Now, we will make this argument robust. To achieve this, we introduce the Frobenius norm of a matrix.

Definition 1 (Frobenius norm) *Let $M = (M_{ij})$ be a matrix. The Frobenius norm of M is*

$$\|M\|_F := \sqrt{\sum_{i,j} M_{ij}^2}.$$

The following fact is a good exercise.

Fact 2 *Let $M = (M_{ij})$ be a matrix. Then*

$$\|M\| \leq \|M\|_F,$$

where $\|\cdot\|$ denotes the spectral norm.

To see how close are C and X, we calculate the Frobenius norm of $C - X$, which will be

$$\begin{aligned} \|C - X\|_F^2 &= \sum_{u,v} C_{uv}^2 + \sum_{u,v} X_{uv}^2 - 2 \sum_{u,v} C_{uv} X_{uv} \\ &\leq 2n^2 - 2n^2 \left(1 - \frac{1}{c}\right) = \frac{2}{c} n^2. \end{aligned}$$

This gives us a bound on the spectral norm of $C - X$, namely

$$\|C - X\| \leq \|C - X\|_F \leq \sqrt{\frac{2}{c}} n.$$

Let $\mathbf{z} = (z_1, \ldots, z_n)$ be the unit eigenvector of X corresponding to its largest eigenvalue, then by Davis-Kahan theorem we have[2]

$$\left\| \mathbf{z} - \frac{1}{\sqrt{n}}\chi \right\| \leq \sqrt{2} \cdot \frac{\sqrt{\frac{2}{c}n}}{n - \sqrt{\frac{2}{c}n}}.$$

For any $\epsilon > 0$, if c is a large enough constant then we will have $\left\| \mathbf{z} - \frac{1}{\sqrt{n}}\chi \right\| \leq \sqrt{\epsilon}$. Now we have the following standard argument:

$$\begin{aligned} \epsilon n &\geq \|\sqrt{n}\mathbf{z} - \chi\|^2 \\ &= \sum_i (\sqrt{n}z_i - \chi_i)^2 \\ &\geq \#\{i : \mathrm{sign}(\sqrt{n}z_i) \neq \chi_i\}. \end{aligned}$$

The last inequality is because every i with $\mathrm{sign}(\sqrt{n}z_i) \neq \chi_i$ will contribute at least 1 in the sum $\sum_i (\sqrt{n}z_i - \chi_i)^2$. This shows that our algorithm will misclassify at most ϵn vertices.

[2]When we apply Davis-Kahan theorem, what we get is actually an upper bound on $\min\{\|\mathbf{z} - \chi/\sqrt{n}\|, \|-\mathbf{z} - \chi/\sqrt{n}\|\}$. We have assumed here that the bound holds for $\|\mathbf{z} - \chi/\sqrt{n}\|$, but the exact same proof will also work in the other case.