Lecture 7

In which we analyze a nearly-linear time algorithm for finding an approximate eigen-vector for the second eigenvalue of a graph adjacency matrix, to be used in the spectral partitioning algorithm.

In past lectures, we showed that, if $G = (V, E)$ is a d-regular graph, and M is its normalized adjacency matrix with eigenvalues $1 = \lambda_1 \geq \lambda_2 \ldots \geq \lambda_n$, given an eigenvector of λ_2, the algorithm SpectralPartition finds, in nearly-linear time $O(|E| + |V| \log |V|)$, a cut $(S, V - S)$ such that $h(S) \leq 2\sqrt{h(G)}$.

More generally, if, instead of being given an eigenvector $\mathbf{x}$ such that $M\mathbf{x} = \lambda_2\mathbf{x}$, we are given a vector $\mathbf{x} \perp \mathbf{1}$ such that $\mathbf{x}^T M \mathbf{x} \geq (\lambda_2 - \epsilon)\mathbf{x}^T\mathbf{x}$, then the algorithm finds a cut such that $h(S) \leq \sqrt{4h(G) + 2\epsilon}$. In this lecture we describe and analyze an algorithm that computes such a vector using $O((|V| + |E|) \cdot \frac{1}{\epsilon} \cdot \log \frac{|V|}{\epsilon})$ arithmetic operations.

A symmetric matrix is *positive semi-definite* (abbreviated PSD) if all its eigenvalues are nonnegative. We begin by describing an algorithm that approximates the largest eigenvalue of a given symmetric PSD matrix. This might not seem to help very much because the adjacency matrix of a graph is not PSD, and because we want to compute the second largest, not the largest, eigenvalue. We will see, however, that the algorithm is easily modified to approximate the second eigenvalue of a PSD matrix (if an eigenvector of the first eigenvalue is known), and that the adjacency matrix of a graph can easily be modified to be PSD.

1 The Power Method to Approximate the Largest Eigenvalue

The algorithm works as follows

Algorithm *Power*

- Input: PSD symmetric matrix $M \in \mathbb{R}^{n \times n}$, positive integer t

- Pick uniformly at random $\mathbf{x}_0 \sim \{-1, 1\}^n$

- for $i := 1$ to t

    - $\mathbf{x}_i := M\mathbf{x}_{i-1}$

- return $\mathbf{x}_t$

That is, the algorithm simply picks uniformly at random a vector $\mathbf{x}$ with ± 1 coordinates, and outputs $M^t\mathbf{x}$.

Note that the algorithm performs $O(t \cdot (n + m))$ arithmetic operations, where m is the number of non-zero entries of the matrix M.

Theorem 1 *For every PSD matrix M, positive integer t and parameter $\epsilon > 0$, with probability $\geq 3/16$ over the choice of $\mathbf{x}_0$, the algorithm Power outputs a vector $\mathbf{x}_t$ such that*

$$\frac{\mathbf{x}_t^T M \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{x}_t} \geq \lambda_1 \cdot (1 - \epsilon) \cdot \frac{1}{1 + 4n(1 - \epsilon)^{2t}}$$

where λ_1 is the largest eigenvalue of M.

Note that, in particular, we can have $t = O(\log n / \epsilon)$ and $\frac{\mathbf{x}_t^T M \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{x}_t} \geq (1 - O(\epsilon)) \cdot \lambda_1$.

Let $\lambda_1 \geq \cdots \lambda_n$ be the eigenvalues of M, with multiplicities, and $\mathbf{v}_1, \ldots, \mathbf{v}_n$ be a system of orthonormal eigenvectors such that $M\mathbf{v}_i = \lambda_i \mathbf{v}_i$. Theorem 1 is implied by the following two lemmas

Lemma 2 *Let $\mathbf{v} \in \mathbb{R}^n$ be a vector such that $||\mathbf{v}|| = 1$. Sample uniformly $\mathbf{x} \sim \{-1, 1\}^n$. Then*

$$\mathbb{P}\left[|\langle \mathbf{x}, \mathbf{v} \rangle| \geq \frac{1}{2}\right] \geq \frac{3}{16}$$

Lemma 3 *Let $\mathbf{x} \in \mathbb{R}^n$ be a vector such that $|\langle \mathbf{x}, \mathbf{v}_1 \rangle| \geq \frac{1}{2}$. Then, for every positive integer t and positive $\epsilon > 0$, if we define $\mathbf{y} := M^t\mathbf{x}$, we have*

$$\frac{\mathbf{y}^T M \mathbf{y}}{\mathbf{y}^T \mathbf{y}} \geq \lambda_1 \cdot (1 - \epsilon) \cdot \frac{1}{1 + 4||\mathbf{x}||^2(1 - \epsilon)^{2t}}$$

2

It remains to prove the two lemmas.

PROOF: [Of Lemma 2] Let $\mathbf{v} = (v_1, \ldots, v_n)$. The inner product $\langle \mathbf{x}, \mathbf{v} \rangle$ is the random variable

$$S := \sum_i x_i v_i$$

Let us compute the first, second, and fourth moment of S.

$$\mathbb{E}\, S = 0$$

$$\mathbb{E}\, S^2 = \sum_i v_i^2 = 1$$

$$\mathbb{E}\, S^4 = 3 \left(\sum_i v_i^2 \right) - 2 \sum_i v_i^4 \leq 3$$

Recall that the Paley-Zygmund inequality states that if Z is a non-negative random variable with finite variance, then, for every $0 \leq \delta \leq 1$, we have

$$\mathbb{P}[Z \geq \delta \,\mathbb{E}\, Z] \geq (1-\delta)^2 \cdot \frac{(\mathbb{E}\, Z)^2}{\mathbb{E}\, Z^2} \tag{1}$$

which follows by noting that

$$\mathbb{E}\, Z = \mathbb{E}[Z \cdot 1_{Z < \delta \,\mathbb{E}\, Z}] + \mathbb{E}[Z \cdot 1_{Z \geq \delta \,\mathbb{E}\, Z}]\ ,$$

that

$$\mathbb{E}[Z \cdot 1_{Z < \delta \,\mathbb{E}\, Z}] \leq \delta \,\mathbb{E}\, Z\ ,$$

and that

$$\mathbb{E}[Z \cdot 1_{Z \geq \delta \,\mathbb{E}\, Z}] \leq \sqrt{\mathbb{E}\, Z^2} \cdot \sqrt{\mathbb{E}\, 1_{Z \geq \delta \,\mathbb{E}\, Z}}$$
$$= \sqrt{\mathbb{E}\, Z^2}\sqrt{\mathbb{P}[Z \geq \delta \,\mathbb{E}\, Z]}$$

We apply the Paley-Zygmund inequality to the case $Z = S^2$ and $\delta = 1/4$, and we derive

$$\mathbb{P}\left[S^2 \geq \frac{1}{4}\right] \geq \left(\frac{3}{4}\right)^2 \cdot \frac{1}{3} = \frac{3}{16}$$

$\square$

Remark 4 *The proof of Lemma 2 works even if* $\mathbf{x} \sim \{-1, 1\}^n$ *is selected according to a 4-wise independent distribution. This means that the algorithm can be derandomized in polynomial time.*

PROOF:[Of Lemma 3] Let us write $\mathbf{x}$ as a linear combination of the eigenvectors

$$\mathbf{x} = a_1 \mathbf{v}_1 + \cdots + a_n \mathbf{v}_n$$

where the coefficients can be computed as $a_i = \langle \mathbf{x}, \mathbf{v}_i \rangle$. Note that, by assumption, $|a_1| \geq .5$, and that, by orthonormality of the eigenvectors, $||\mathbf{x}||^2 = \sum_i a_i^2$.

We have

$$\mathbf{y} = a_1 \lambda_1^t \mathbf{v}_1 + \cdots + a_n \lambda_n^t \mathbf{v}_n$$

and so

$$\mathbf{y}^T M \mathbf{y} = \sum_i a_i^2 \lambda_i^{2t+1}$$

and

$$\mathbf{y}^T \mathbf{y} = \sum_i a_i^2 \lambda_i^{2t}$$

We need to prove a lower bound to the ratio of the above two quantities. We will compute a lower bound to the numerator and an upper bound to the denominator in terms of the same parameter.

Let k be the number of eigenvalues larger than $\lambda_1 \cdot (1 - \epsilon)$. Then, recalling that the eigenvalues are sorted in non-increasing order, we have

$$\mathbf{y}^T M \mathbf{y} \geq \sum_{i=1}^k a_i^2 \lambda_i^{2t+1} \geq \lambda_1 (1 - \epsilon) \sum_{i=1}^k a_i^2 \lambda_i^{2t}$$

We also see that

$$\sum_{i=k+1}^n a_i^2 \lambda_i^{2t}$$

$$\leq \lambda_1^{2t} \cdot (1 - \epsilon)^{2t} \sum_{i=k+1}^n a_i^2$$

$$\leq \lambda_1^{2t} \cdot (1 - \epsilon)^{2t} \cdot ||\mathbf{x}||^2$$

4

$$\leq 4a_1^2 \lambda_1^{2t}(1-\epsilon)^{2t}||\mathbf{x}||^2$$

$$\leq 4||\mathbf{x}||^2(1-\epsilon)^{2t}\sum_{i=1}^{k} a_i^2 \lambda_i^{2t}$$

So we have

$$\mathbf{y}^T\mathbf{y} \leq (1 + 4||\mathbf{x}||^2(1-\epsilon)^{2t}) \cdot \sum_{i=1}^{k} a_i^2$$

giving

$$\frac{\mathbf{y}^T M \mathbf{y}}{\mathbf{y}^T \mathbf{y}} \geq \lambda_1 \cdot (1 - \epsilon) \cdot \frac{1}{1 + 4||\mathbf{x}||^2(1-\epsilon)^{2t}}$$

$\square$

Remark 5 *Where did we use the assumption that M is positive semidefinite? What happens if we apply this algorithm to the adjacency matrix of a bipartite graph?*

2 Approximating the Second Eigenvalue

If M is a PSD matrix, and if we know a unit-length eigenvector $\mathbf{v}_1$ of the largest eigenvalue of M, we can approximately find the second eigenvalue with the following adaption of the algorithm from the previous section.

Algorithm *Power2*

- Input: PSD symmetric matrix $M \in \mathbb{R}^{n \times n}$, positive integer t, vector $\mathbf{v}_1$

- Pick uniformly at random $\mathbf{x} \sim \{-1, 1\}^n$

- $\mathbf{x}_0 := \mathbf{x} - \langle \mathbf{v}_1, \mathbf{x} \rangle \cdot \mathbf{v}_1$

- for $i := 1$ to t

    $-$ $\mathbf{x}_i := M\mathbf{x}_{i-1}$

- return $\mathbf{x}_t$

If $\mathbf{v}_1, \ldots, \mathbf{v}_n$ is an orthonormal basis of eigenvectors for the eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n$ of M, then, at the beginning, we pick a random vector

$$\mathbf{x} = a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \cdots a_n\mathbf{v}_n$$

that, with probability at least $3/16$, satisfies $|a_2| \geq 1/2$. (Cf. Lemma 2.) Then we compute $\mathbf{x}_0$, which is the projection of $\mathbf{x}$ on the subspace orthogonal to $\mathbf{v}_1$, that is

$$\mathbf{x}_0 = a_2 \mathbf{v}_2 + \cdots a_n \mathbf{v}_n$$

Note that $||\mathbf{x}||^2 = n$ and that $||\mathbf{x}_0||^2 \leq n$.

The output is the vector $\mathbf{x}_t$

$$\mathbf{x}_t = a_2 \lambda_2^t \mathbf{v}^2 + \cdots a_n \lambda_n^t \mathbf{v}^n$$

If we apply Lemma 3 to subspace orthogonal to $\mathbf{v}_1$, we see that when $|a_2| \geq 1/2$ we have that, for every $0 < \epsilon < 1$,

$$\frac{\mathbf{x}_t^T M \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{x}_t} \geq \lambda_2 \cdot (1 - \epsilon) \cdot \frac{1}{4n(1 - \epsilon)^{2t}}$$

We have thus established the following analysis.

Theorem 6 *For every PSD matrix M, positive integer t and parameter $\epsilon > 0$, if $\mathbf{v}_1$ is a length-1 eigenvector of the largest eigenvalue of M, then with probability $\geq 3/16$ over the choice of $\mathbf{x}_0$, the algorithm Power2 outputs a vector $\mathbf{x}_t \perp \mathbf{v}_1$ such that*

$$\frac{\mathbf{x}_t^T M \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{x}_t} \geq \lambda_2 \cdot (1 - \epsilon) \cdot \frac{1}{1 + 4n(1 - \epsilon)^{2t}}$$

where λ_2 is the second largest eigenvalue of M, counting multiplicities.

Finally, we come to the case in which M is the normalized adjacency matrix of a regular graph.

We know that M has eigenvalues $1 = \lambda_1 \geq \cdots \lambda_n \geq -1$ and that $\frac{1}{\sqrt{n}} \cdot \mathbf{1}$ is an eigenvector of λ_1.

Consider now the matrix $M + I$. Every eigenvector of M with eigenvalue λ is clearly also an eigenvector of $M + I$ with eigenvalue $1 + \lambda$, and vice versa, thus $M + I$ has eigenvalues $2 = 1 + \lambda_1 \geq 1 + \lambda_2 \geq \cdots \geq 1 + \lambda_n \geq 0$ and it is PSD.

This means that we can run algorithm Power2 on the matrix $I + M$ using the vector $\mathbf{v}_1 = \frac{1}{\sqrt{n}} \mathbf{1}$ and a parameter $t = O(\epsilon^{-1} \log n/\epsilon))$. The algorithm finds, with probability $\geq 3/16$, a vector $\mathbf{x}_t \perp \mathbf{1}$ such that

$$\frac{\mathbf{x}_t^T (M + I) \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{x}_t} \geq (1 + \lambda_2) \cdot (1 - 2\epsilon)$$

which is equivalent to

$$\frac{\mathbf{x}_t^T M \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{x}_t} \geq \lambda_2 - 2\epsilon - 2\epsilon\lambda_2 \geq \lambda_2 - 4\epsilon$$