Fall 2014 Randomized Algorithms	Nov 12, 2014
Lecture 8	
Prof. Friedrich Eisenbrand	*Scribes: Mateusz Lewandowski*

In this lecture we will consider two topics:

- The Power Method for computing largest eigenvalue
- Probability amplification by random walks on expanders

1 The Power Method

Given a symmetric positive semi-definite array $M \in R^{n \times n}, M \succeq 0$ we would like to find an approximation of the largest eigenvalue of M.

Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$ be eigenvalues of M (all of them are nonnegative since M is symmetric PSD) and let $\mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_n}$ be a system of orthonormal eigenvectors corresponding to these eigenvalues
(i.e. $M\mathbf{v_i} = \lambda_i \mathbf{v_i}$ for every $i \in \{1, 2, \ldots, n\}$)

```
1 Pick uniformly at random x ~ {-1,1}^n
2 Return y = M^k x (for some k ∈ Z_+)
```

Algorithm 1: approximating the largest eigenvalue (actually the eigenvector)

Claim 1 *For every $\epsilon > 0, k \in \mathbb{Z}_+$ Algorithm 2 returns an $\mathbf{y}$ such that:*

$$\frac{\mathbf{y}^T M \mathbf{y}}{\mathbf{y}^T \mathbf{y}} \geq (1-\epsilon)\lambda_1 \frac{1}{1+4n(1-\epsilon)^{2k}}$$

with constant probability (at least $\frac{3}{16}$).

Observe that if we set $k = \Omega(\frac{\log n}{\epsilon})$ then this lower bound becomes
$\frac{\mathbf{y}^T M \mathbf{y}}{\mathbf{y}^T \mathbf{y}} \geq (1 - O(\epsilon))\lambda_1$
To prove our claim, we will prove two lemmas.

Lemma 2 *Let $\mathbf{v} \in R^n$ be any vector such that $\|\mathbf{v}\|_2 = 1$. If $\mathbf{x} \sim \{-1,1\}^n$ is picked uniformly at random then*

$$|\langle \mathbf{v}, \mathbf{x} \rangle| \geq \frac{1}{2}$$

with constant probability (at least $\frac{3}{16}$).

Proof Let $\mathbf{v} = (a_1, a_2, \ldots, a_n)$. Consider now an inner product $\langle \mathbf{v}, \mathbf{x} \rangle$ which is a random variable

$$S = \sum_{i=1}^{n} a_i x_i$$

We will now analyze the 1^{st}, 2^{nd} and 4^{th} moment of S.
Because $\mathbf{x} \sim \{-1, 1\}^n$ and $\|\mathbf{v}\|_2 = 1$ we have:

$$\mathbb{E}[S] = 0$$

$$\mathbb{E}[S^2] = \sum_{i=1}^{n} a_i = 1$$

$$\begin{aligned}
\mathbb{E}[S^4] &= \mathbb{E}[\sum_{i=1}^{n} x_i a_i] \\
&= \mathbb{E}[\sum_{i,j,k,l} x_i x_j x_k x_l a_i a_j a_k a_l] \\
&= \mathbb{E}[\sum_{i} x_i^4 a_i^4 + 6 \sum_{i<j} x_i^2 x_j^2 a_i^2 a_j^2] \\
&= \sum_{i} a_i^4 + 6 \sum_{i<j} a_i^2 a_j^2 \\
&= 3 \left(\sum_{i} a_i^2 \right)^2 - 2 \sum_{i} v_i^4 \\
&\leq 3
\end{aligned}$$

Fact 3 *(Paley-Zygmund inequality) If X is a non-negative random variable with finite variance, then, for every $0 \leq \delta \leq 1$*

$$\mathbb{P}\Big[X \geq \delta \mathbb{E}[X]\Big] \geq (1 - \delta)^2 \frac{\mathbb{E}[X]^2}{\mathbb{E}[X^2]}$$

Using now Fact 3 for $X = S^2$ and $\delta = \frac{1}{4}$ we have:

$$\begin{aligned}
\mathbb{P}\Big[S^2 \geq \frac{1}{4} \cdot 1\Big] &\geq (1 - \frac{1}{4})^2 \cdot \frac{1}{3} \\
&= \left(\frac{3}{4}\right)^2 \cdot \frac{1}{3} \\
&= \frac{3}{16}
\end{aligned}$$

Which proves the lemma because $S^2 \geq \frac{1}{4}$ implies that $|\langle \mathbf{v}, \mathbf{x} \rangle| \geq \frac{1}{2}$ ■

Now in the following lemma we will see that the result from Lemma 2 applied to eigenvector $\mathbf{v_1}$ gives us our desired proof of the claim.

Lemma 4 *When* $|\langle \mathbf{v_1}, \mathbf{x} \rangle| \geq \frac{1}{2}$ *then for every* $\epsilon > 0$ *we have:*

$$\frac{\mathbf{y}^T M \mathbf{y}}{\mathbf{y}^T \mathbf{y}} \geq (1-\epsilon)\lambda_1 \frac{1}{1 + 4n(1-\epsilon)^{2k}}$$

Proof Since $\mathbf{v_1}, \mathbf{v_2}, \ldots \mathbf{v_n}$ is an orthonormal basis, we can express $\mathbf{x}$ as a linear combination of these eigenvectors:

$$\mathbf{x} = \alpha_1 \mathbf{v_1} + \alpha_2 \mathbf{v_2} + \cdots + \alpha_n \mathbf{v_n}$$

Then

$$\mathbf{y} = M^k \mathbf{x} = \sum_i \alpha_i M^k \mathbf{v_i} = \sum_i \alpha_i \lambda_i^k \mathbf{v_i}$$

which now implies that

$$\mathbf{y}^T M \mathbf{y} = \mathbf{y}^T \left(\sum_i \alpha_i \lambda_i^k M \mathbf{v_i} \right) = \sum_i \alpha_i^2 \lambda_1^{2k+1}$$

where in the last equality we used (?) Also we have that

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

To get a lower bound on $\frac{\mathbf{y}^T M \mathbf{y}}{\mathbf{y}^T \mathbf{y}}$ we will give a lower bound on the nominator and an upper bound on the denominator.
To do this, define first l to be the number of eigenvalues larger than $\lambda_1(1-\epsilon)$. This is the same as picking l so that $\lambda_i \geq \lambda_1(1-\epsilon)$ for each $i \in \{1, 2, \ldots, l\}$ and $\lambda_i < \lambda_1(1-\epsilon)$ for each $i \in \{l+1, l+2, \ldots, n\}$
We now lower bound the nominator:

$$\begin{aligned}\mathbf{y}^T M \mathbf{y} &= \sum_i \alpha_i^2 \lambda_1^{2k+1} \\ &\geq \lambda_1(1-\epsilon) \sum_{i=1}^{l} \alpha_i^2 \lambda_i^{2k}\end{aligned}$$

and upper bound the denominator:

$$\begin{aligned}
\mathbf{y}^T\mathbf{y} = \sum_i \alpha_i^2 \lambda_i^{2t} &= \sum_{i\le l} \alpha_i^2\lambda_i^{2k} + \sum_{i>l}\alpha_i^2\lambda_i^{2k} \\
&\le \left(\sum_{i>l}\alpha_i^2\right)\left(\lambda_1^{2k}\,(1-\epsilon)^{2k}\right) \\
&\le \|\mathbf{x}\|_2^2 \cdot \lambda_1^{2k} \cdot (1-\epsilon)^{2k} \\
&\le 4\|\mathbf{x}\|_2^2 \cdot (1-\epsilon)^{2k} \cdot \sum_{i=1}^{l} \alpha_i^2\lambda_i^{2k} \\
&\le (1+4n(1-\epsilon^{2k}))\sum_{i=1}^{l}\alpha_i^2\lambda_i^{2k}
\end{aligned}$$

where to obtain second line we used $M \succeq 0$ and to obtain 4^{th} line we used Lemma 2.

Now putting these two bounds together we have:

$$\begin{aligned}
\frac{\mathbf{y}^T M\mathbf{y}}{\mathbf{y}^T\mathbf{y}} &\ge \frac{\lambda_1(1-\epsilon)\sum_{i=1}^{l}\alpha_i^2\lambda_i^{2k}}{(1+4n(1-\epsilon^{2k}))\sum_{i=1}^{l}\alpha_i^2\lambda_i^{2k}} \\
&= (1-\epsilon)\lambda_1 \frac{1}{1+4n(1-\epsilon)^{2k}}
\end{aligned}$$

■

Remark Derandomization of the algorithm. Observe that in Lemma 2 we only use independence on 4 coordinates. So any distribution giving proper moments will work. We could use a 4-wise independent distribution over $\{-1,1\}^n$ s.t $\mathbb{E}[x_i] = 0$.

Observation 5 *If we knew exactly the eigenvector* $\mathbf{v_1}$ *we could use similar algorithm to compute approximation of* $\mathbf{v_2}$ *(and* λ_2*)*

1 *Pick uniformly at random* $\mathbf{x} \sim \{-1,1\}^n$
2 $\mathbf{x}' = \mathbf{x} - |\langle \mathbf{v_1}, \mathbf{x}\rangle| \cdot \mathbf{v_1}$
3 **Return** $\mathbf{y} = M^k\mathbf{x}'$ *(for some* $k \in \mathbb{Z}_+$*)*

Algorithm 2: approximating second largest eigenvalue

Analysis will be similar.

2 Probability amplification by random walks on expanders

Let's consider a following problem.
We're given an BPP algorithm $\mathcal{A}$ deciding language $\mathcal{L}$ i.e. for any input $x \in \{0,1\}^*$

- if $x \in \mathcal{L}$ then $Pr[\mathcal{A}(x,r) \text{ rejects}] \leq \frac{1}{100}$
- if $x \notin \mathcal{L}$ then $Pr[\mathcal{A}(x,r) \text{ accepts}] \leq \frac{1}{100}$

where $\mathcal{A}(x,r)$ is an output of $\mathcal{A}$ on input x and vector x of random bits of length n (assume that $\mathcal{A}$ uses n random bits).

Our **goal** is to reduce the probability of errors.

Consider now usual naive approach to tackle this problem:

1. run independently algorithm $\mathcal{A}$ k times
2. output majority (most frequent answer)

Using Chernoff bounds we can easily show that error probability is now reduced to $2^{-\Omega(k)}$.
This is fine, but we are using kn random bits.

We can do better. We will obtain the same probability guarantee using only $n + O(k)$ bits. We will use random walks on some class of expanders to do this.

Definition 6 *An (n,d,c)-expander is a d-regular bipartite (multi)graph $G(X \cup Y, E)$ with $|X| = |Y| = |\frac{n}{2}|$ such that for any $S \subseteq X$:*

$$|\Gamma(S)| \geq \left(1 + c(1 - \frac{2|S|}{n})\right)|S|$$

where $\Gamma(S)$ is a set of vertices neighboring to S

It is worth mentioning that a random graph (taken with some care) will be an expander with high probability. However checking if any graph is an expander is a hard problem. So we are looking for some explicit construction.

2.1 Gabber-Galil expanders - construction

Let m be a positive integer. Consider a bipartite graph $G(X \cup Y, E)$ with $|X| = |Y| = m^2$. We can label each vertex in X by a pair $(a,b) \in \mathbb{Z}_m^2$. We do the same for vertices in Y. Now we define the set of edges E by saying that each vertex (a,b) from X is connected to following vertices from Y:

- (a,b)

- $(a, 2a + b)$
- $(a, 2a + b + 1)$
- $(a, 2a + b + 2)$
- $(a + 2b, b)$
- $(a + 2b + 1, b)$
- $(a + 2b + 2, b)$

where the operation $+$ is modulo m.
Now the following fact can be shown (we omit the proof since it's not relevant to the lecture)

Fact 7 *G is $(2m^2, 7, \frac{2-\sqrt{3}}{2})$-expander*

Observer only that $n = 2m^2$ and degree of each vertex of G is 7.
Note also that if A is the adjacency matrix of G then we have following eigenvalues of A:

$$7 = d = \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{2m^2} = -d = -7$$

and $|\lambda_2| \leq 7 - \epsilon$ for some $\epsilon > 0$.
Note also that we don't have to store the whole graph. It is enough that we can quickly compute the set of neighbors.

If we now consider a random walk on G with a transition matrix $P = \frac{A}{7}$, we observe that this results in periodic Markov chain (because G is bipartite), hence there is no stationary distribution. We can handle this problem by performing a lazy random walk (in which we stay at vertex with probability $\frac{1}{2}$). So now our transition matrix will be simply:

$$Q = \frac{I + \frac{A}{7}}{2}$$

Let now $\lambda'_1 \geq \lambda'_2 \geq \cdots \geq \lambda_{2m^2}$ be the eigenvalues of P. It is easy to see that $\lambda'_i = \frac{1}{2} \cdot \left(1 + \frac{\lambda_i}{7}\right)$. Hence we have now that $1 \geq \lambda'_1 \geq \lambda'_{2m^2} \geq 0$ and also $\lambda'_2 = 1 - \frac{\epsilon}{14}$

2.2 Algorithm for efficient probability amplification

We are now ready to give the algorithm to reduce the probability of error of $\mathcal{A}$ which uses only $n + O(k)$ bits.
We will assume w.l.o.g that n is odd (recall that n is the number of random bits used by $\mathcal{A}$, i.e. $r \in \{0, 1\}^n$)

Set $m = 2^{\frac{n-1}{2}}$
`//` so number of vertices $2m^2 = 2^n = N$ is equal to a total number of $\{0,1\}^n$ strings
Fix some distinct identifiers to vertices of G from $\{0,1\}^n$
Pick a starting vertex v uniformly at random
Perform a lazy random walk from v according to Q: let $X_0, X_1, \dots$ be the states of the resulting Markov chain
Set $r_i = X_{i\cdot\beta}$ `//` β is an integer constant such that $\lambda_2^{'\beta} \leq 10$
Output majority of $\mathcal{A}(x, r_1), \mathcal{A}(x, r_2), \dots, \mathcal{A}(x, r_{7k})$

Algorithm 3: probability amplification

2.3 Analysis

Observe that this algorithm uses only $n+O(k)$ random bits. We need n random bits to choose a random starting vertex v and at most 4 bits for each of the $7k\beta$ steps of the random walk. Moreover the algorithm runs in polynomial time (we don't store the whole graph G of exponential size, because we know how to quickly obtain neighbors).

Lemma 8 *Algorithm 3 has at most $\frac{1}{2^{\Omega(k)}}$ probability of error.*

Proof
Fix some input x.
Let $\mathcal{W} = \{r \in \{0,1\}^n : \mathcal{A}(x,r) \text{ is correct}\}$ be a set of witnesses.
We know that $|\mathcal{W}| \geq 0.99N$ ($\mathcal{A}$ is BPP algorithm).
Define $n \times n$ diagonal matrix W such that $W_{i,j} = \begin{cases} 1 & \text{if } i = j \text{ and } i \in \mathcal{W} \\ 0 & \text{otherwise} \end{cases}$
Let $\overline{W} = I - W$.
Let also $p^0 = (\frac{1}{N}, \dots, \frac{1}{N}) \in \mathbb{R}^n$ be an initial distribution of our random walk and define $p^i = p^0 Q^i$.
Now the probability that X_i is a witness is equal to $\|p^i W\|_1$
Define now the sequence of matrices $\mathcal{S} = (S_1, \dots, S_{7k}) \in \{W, \overline{W}\}^{7k}$ where $S_i = \begin{cases} W & \text{if } r_i \in \mathcal{W} \\ \overline{W} & \text{otherwise} \end{cases}$
We can see that: $Pr[\mathcal{S} \text{ occurs}] = \|p^0(Q^\beta S_1)(Q^\beta S_2) \cdot \dots \cdot (Q^\beta S_{7k})\|_1$

Claim 9 *For each $p \in \mathbb{R}^N$*

$$\|pQ^\beta W\| \leq \|p\|$$

$$\|pQ^\beta \overline{W}\| \leq \frac{\|p\|}{5}$$

Before we prove Claim 9 we will see how it helps us prove our lemma.
Let $\mathbb{S}$ be a fixed erroneousness signature (majority of the elements is equal to

$\overline{W}$). Let's say that it has $t \geq \frac{7k}{2}$ elements $\overline{W}$

$$\begin{aligned} Pr[\mathcal{S} \text{ occurs}] &= \|p^0(Q^\beta S_1)(Q^\beta S_2) \cdot \dots \cdot (Q^\beta S_{7k})\|_1 \\ &\leq \sqrt{N}\|p^0(Q^\beta S_1)(Q^\beta S_2) \cdot \dots \cdot (Q^\beta S_{7k})\|_2 \\ &\leq \sqrt{N}\left(\frac{1}{5}\right)^t \|p^0\|_2 \\ &\leq \sqrt{N}\left(\frac{1}{5}\right)^{\frac{7k}{2}} \|p^0\|_2 \\ &\leq \left(\frac{1}{5}\right)^{\frac{7k}{2}} \end{aligned}$$

where the second line is using Cauchy-Schwartz inequality and the third follows from repeatedly used Claim 9 and the last line follows from the fact that we chose p^0 uniformly from N vertices.
Now we can estimate probability of error.

$$Pr[\text{Algorithm 3 makes error}] \leq 2^{7k} \cdot \left(\frac{1}{5}\right)^{\frac{7k}{2}} = \frac{1}{2^{\Omega(k)}}$$

We now finish the proof by proving Claim 9.
Let $v_1, v_2, \dots v_n$ be an orthonormal set of eigenvectors of Q corresponding to eigenvalues λ_i'. We can express p as a linear combination of these eigenvectors. So let $p = \sum_i a_i v_i$. Having in memory that each λ_i' lies in $[0, 1]$ we have:

$$\begin{aligned} \|pQ^\beta W\|^2 &\leq \|pQ^\beta\|^2 \\ &= \|\sum_i a_i \lambda_i'^\beta v_i\|_2^2 \\ &\leq \sum_i a_i^2 \lambda_i'^{2\beta} \\ &\leq \|p\|^2 \end{aligned}$$

which after removing squares gives us the first inequality of the claim.
To prove second inequality of the claim, decompose $p = x + y$ where $x = a_1 v_1$ and $y = \sum_{i=2}^N a_i v_i$.
Observe that $\|x\| \leq \|p\|$ and $\|y\| \leq \|p\|$.
We will now see that $\|xQ^\beta \overline{W}\| \leq \frac{\|x\|}{10}$.
See that $\overline{W}$ zeros out all but $\frac{1}{100}$ fraction of the entries of x which have all components equal. So the L_2 norm of x will be reduced by $\sqrt{100}$ after multiplying by $\overline{W}$. So we have

$$\|xQ^\beta \overline{W}\| = \|x\overline{W}\| \leq \frac{\|x\|}{10}$$

where the first equality is due to fact $\lambda_1' = 1$
Now we will see that $\|yQ^\beta \overline{W}\| \leq \frac{\|y\|}{10}$.
Observe that $yQ^\beta = \sum_{i=2}^{N} a_i v_i Q^\beta = \sum_{i=2}^{N} a_i \lambda_i^{'\beta} v_i$ and $\|yQ^\beta \overline{W}\| \leq \|yQ^\beta\|$.
Recall also that we chose β so that $\lambda_2^{'\beta} \leq \frac{1}{10}$.
Putting these together we have

$$\|yQ^\beta \overline{W}\| \leq \sqrt{\sum_{i=2}^{N} a_i^2 \lambda_i^{'2\beta}} \leq \lambda_2^{'\beta} \sqrt{\sum_{i=2}^{N} a_i^2} \leq \frac{\|y\|}{10}$$

Finally we obtain

$$\begin{aligned} \|pQ^\beta \overline{W}\| &\leq \|xQ^\beta \overline{W}\| + \|yQ^\beta \overline{W}\| \\ &\leq \frac{\|x\| + \|y\|}{10} \\ &\leq \frac{\|p\|}{5} \end{aligned}$$

which proves the claim and finishes our analysis. ■