6.5420 Randomness and Computation February 19, 2026

Lecture 5

Lecturer: Ronitt Rubinfeld *Scribe: Nicole Shen*

In this lecture, we examine the behavior of Markov chains over time; in particular, we care about the how long it takes to reach stationary distribution (mixing time), and how this connects to the eigenvalues of the transition matrix. Randomized complexity classes, along with an algorithm to derandomize through enumeration are also introduced. A goal for future lectures will be to reduce the runtime of this algorithm.

1 Linear Algebra and Random Walks

1.1 Linear Algebra Review

First, we review some basic linear algebra to provide a foundation for our analysis on mixing times.

Definition 1. *v is an* eigenvector *of matrix A with corresponding* eigenvalue *λ iff $vA = \lambda v$.*

For example, a stationary distribution Π is defined by $\Pi \cdot A = \Pi$. Here, Π is an eigenvector of A with corresponding eigenvalue 1.

Definition 2. *We define L_2*-norm *of*

$$v = (v_1, \ldots, v_n) = \sqrt{\sum_{i=1}^{n} v_i^2} = \sqrt{v \cdot v}.$$

Definition 3. *Define the* inner product

$$u \cdot v = \sum_{i=1}^{n} u_i v_i.$$

Definition 4. *A set of vectors $v^{(1)}, \ldots, v^{(m)}$ are* orthonormal *if*

$$v^{(i)} \cdot v^{(j)} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

for all $i, j \in [1, m]$.

We care about orthonormal vectors because they can form a basis.

Example 5. *P is a transition matrix of a d-regular undirected graph, meaning it is doubly stochastic. Then, the following equations hold:*

- $(\frac{1}{n}, \cdots, \frac{1}{n}) \cdot P = 1 \cdot (\frac{1}{n}, \cdots, \frac{1}{n})$.
- $(\frac{1}{\sqrt{n}}, \cdots, \frac{1}{\sqrt{n}}) \cdot P = 1 \cdot (\frac{1}{\sqrt{n}}, \cdots, \frac{1}{\sqrt{n}})$.

Upon first glance, the eigenvector $(\frac{1}{n}, \cdots, \frac{1}{n})$ appears more natural, since it is a probability distribution vector representing the uniform distribution. However, in practice, the eigenvector $(\frac{1}{\sqrt{n}}, \cdots, \frac{1}{\sqrt{n}})$ is also used a lot, because it is often useful to have L_2-norm $= 1$.

Theorem 6. *Given real and symmetric transition matrix P, there exists eigenvectors $v^{(1)}, \ldots, v^{(n)}$ forming an orthonormal basis with corresponding eigenvalues $1 = \lambda_1 \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$, and $v^{(1)} = \frac{1}{\sqrt{n}}(1, \cdots, 1)$.*

The proof is omitted for this lecture.

1.2 Useful Facts

We motivate the importance of being able to express the eigenvectors and eigenvalues of variations of P. vP means we're taking a step in a random walk. $v \cdot [\frac{1}{2}(P+I)]$ represents taking a step with probability $\frac{1}{2}$, and staying in place with probability $\frac{1}{2}$. Similarly, $v \cdot P^2$ represents taking two steps in a random walk. The eigenvectors and eigenvalues of matrices corresponding to these situations are valuable to understand, since they give us insight into how Markov chains correlated to the one described by P behaves.

Theorem 7. *Let P be a transition matrix with all positive entries, having eigenvectors $v^{(1)}, \ldots, v^{(n)}$ with corresponding eigenvalues $\lambda_1, \ldots, \lambda_n$. Then, the following hold:*

1. *αP has eigenvectors $v^{(1)}, \ldots, v^{(n)}$ with corresponding eigenvalues $\alpha\lambda_1, \ldots, \alpha\lambda_n$.*
2. *$P + I$ has eigenvectors $v^{(1)}, \ldots, v^{(n)}$ with corresponding eigenvalues $\lambda_1 + 1, \ldots, \lambda_n + 1$.*
3. *P^k has eigenvectors $v^{(1)}, \ldots, v^{(n)}$ with corresponding eigenvalues $\lambda_1^k, \ldots, \lambda_n^k$.*
4. *If P is stochastic, then $|\lambda_i| \leq 1, \forall i$.*

Proof. We give simple proofs for each part individually.

1. $vP = \lambda v \iff v \cdot (\alpha P) = \lambda \cdot (\alpha v)$.
2. $v(P + I) = vP + vI = \lambda v + v = (\lambda + 1)v$.
3. $vP^k = (vP)P^{k-1} = \lambda v \cdot P^{k-1} = \cdots = \lambda^k v$.
4. $\forall i$, let $I = \{j \mid v_j^{(i)} > 0\}$. This is the set of indices for positive elements of each eigenvector. Then,

$$\begin{aligned}
\lambda \sum_{j \in I} v_j^{(i)} &= \sum_{j \in I} \sum_{k} v_k^{(i)} P_{kj} \\
&\leq \sum_{j,k \in I} v_k^{(i)} P_{kj} \\
&\leq \sum_{k \in I} v_k^{(i)} \sum_{j \in I} P_{ij} \\
&\leq \sum_{k \in I} v_k^{(i)}.
\end{aligned}$$

Thus, $\lambda \leq 1$. The second inequality follows because entries of v not in I are all ≤ 0. The final inequality follows from the stochasticity of P.

□

Note that if $v^{(1)}, \ldots, v^{(n)}$ is an orthonormal basis, then any vector w is expressible as a linear combination of $v^{(i)}$'s.

$$w = \sum_i \alpha_i v^{(i)},$$

and the L_2-norm of w is given by

$$\begin{aligned}||w||_2 &= \sqrt{(\sum_i \alpha_i v^{(i)})(\sum_j \alpha_j v^{(j)})} \\ &= \sqrt{\sum_{i,j} \alpha_i \alpha_j v^{(i)} v^{(j)}} \\ &= \sqrt{\sum_i \alpha_i^2}.\end{aligned}$$

The second equality follows from the definitions of orthonormal vectors. This result is useful because it allows us to calculate the L_2-norm of any vector that can be expressed using an orthonormal basis.

2 Mixing Times

Now that we have the necessary linear algebra background, we examine mixing times. The mixing time tells us how long it takes to reach a stationary distribution for a Markov chain or random walk.

Definition 8. *Given $\epsilon > 0$, we define the* mixing time*, $T(\epsilon)$ of Markov chain A with stationary distribution Π, to be the minimum t such that $\forall \Pi^{(0)}$ (any starting distribution), it holds that*

$$||\Pi - \Pi^{(0)} A^t||_1 \leq \epsilon.$$

Definition 9. *A Markov chain A is* rapidly mixing *if $T(\epsilon) = poly(\log n, \log \frac{1}{\epsilon})$, where n is the number of states in A.*

An example of a rapidly mixing Markov chain is the complete graph, which mixes after just one step.

There are a lot of graphs that are not rapidly mixing. For example, the line graph takes n^2 time to possibly reach every state. In fact, it is intuitive that a lot of graphs are not rapidly mixing, because $poly(\log n, \log \frac{1}{\epsilon})$ is generally much less than the cover time.

Note that mixing time of $\frac{P+I}{2}$ is at most 2 times more. This constant factor does not affect whether the Markov chain is classified as rapidly mixing. Hence, we can freely add self-loops.

Theorem 10. *P is a transition matrix of undirected, non k-partite, d-regular connected graph. Let Π_0 be the starting distribution, and let Π be the stationary distribution $(\frac{1}{n}, \cdots, \frac{1}{n})$. Then,*

$$||\Pi_0 P^t - \Pi||_2 \leq |\lambda_2|^t.$$

Proof. Since P is real and symmetric, there exists eigenvectors $v^{(1)}, \dots v^{(n)}$ which form an orthonormal basis with eigenvalues $1 = \lambda_1 \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$ and $v^{(1)} = \frac{1}{\sqrt{n}}(1, \cdots, 1)$ by Theorem 6.

So any distribution vector, in particular Π_0, can be expressed as a linear combination of $v^{(i)}$'s: $\Pi_0 = \sum_{i=1}^n \alpha_i v^{(i)}$.

Hence,

$$\begin{aligned}\Pi_0 P^t &= \sum_{i=1}^n \alpha_i v^{(i)} P^t \\ &= \lambda_i^t v^{(i)} \\ &= \alpha_1 \lambda_1^t v^{(1)} + \sum_{i=2}^n \alpha_i \lambda_i^t v^{(i)}.\end{aligned}$$

Now, the distance between the Markov chain after t steps and the stationary distribution can bounded as follows:

$$\begin{aligned}
||\Pi_0 \cdot P^t - \Pi||_2 &= ||\Pi_0 \cdot P^t - \alpha_1 v^{(1)}||_2 \\
&= ||\sum_{i=2}^{n} \alpha_i \lambda_i^t v^{(i)}||_2 \\
&= \sqrt{\sum_{i=2}^{n} \alpha_i^2 \lambda_i^{2t}} \\
&\leq |\lambda_2|^t \sqrt{\sum_{i=2}^{n} \alpha_i^2} \\
&\leq |\lambda_2|^t ||\Pi_0||_2 \\
&\leq |\lambda_2|^t.
\end{aligned}$$

The last inequality follows from the fact that the L_2-norm is always less than the L_1-norm when all entries are ≤ 1.

□

Essentially, we have used basic linear algebra to make statements on how quickly stationary distribution can be reached for Markov chains and random walks, when $|\lambda_2| < 1$.

As a sanity check and exercise, we now show how to find α_1 and $\alpha_1 \lambda_1^t v^{(1)}$.

By definition, $v^{(1)} = \frac{1}{\sqrt{n}}(1, \cdots, 1)$. Then,

$$\Pi_0 v^{(1)} = \alpha_1 v^{(1)} \cdot v^{(1)} + \sum_{i=2}^{n} \alpha_i v^{(i)} \cdot v^{(1)} = \alpha_1,$$

since $v^{(1)} \cdot v^{(1)} = 1$ and $v^{(i)} \cdot v^{(1)} = 0, \forall i \neq 1$.

Also, we have

$$\Pi_0 v^{(1)} = \Pi_0 \cdot \frac{1}{\sqrt{n}}(1, \cdots, 1) = \frac{1}{\sqrt{n}} \Pi_0 \cdot (1, \cdots, 1) = \frac{1}{\sqrt{n}}.$$

Since Π_0 is a probability distribution over the states, $\Pi_0 \cdot (1, \cdots, 1) = 1$. Thus, we have $\alpha_1 = \frac{1}{\sqrt{n}}$, and $\alpha_1 \lambda_1^t v^{(1)} = (\frac{1}{n}, \cdots, \frac{1}{n}) = \Pi$. Note that this argument does not use any knowledge of Π_0, other than that it is a distribution.

3 Randomized Complexity Classes

We introduce randomized complexity classes. These give us ways to describe problems that are solvable within a specific time complexity when given access to a randomness generator.

Definition 11. *A* language *L is a subset of* $\{0,1\}^*$.

Intuitively, the strings represent bit encodings, and a language can be thought of as the set of encodings that correspond to the solutions of a problem. For example, $\{x \mid x$ is a graph with a Hamiltonian path$\}$ and $\{x \mid x$ is a collection of sets with a proper 2-coloring$\}$ are languages.

Definition 12. *P is a class of languages L with a polynomial-time* ***deterministic*** *algorithm A such that*

$$x \in L \implies A(x) \textit{ accepts}$$
$$x \notin L \implies A(x) \textit{ rejects.}$$

Definition 13. *RP is a class of languages L with a polynomial-time probabilistic algorithm A such that*

$$x \in L \implies \Pr[A(x) \textit{ accepts} \geq \frac{1}{2}]$$
$$x \notin L \implies \Pr[A(x) \textit{ accepts}] = 0.$$

This is 1-sided error: if $A(x)$ accepts, then x must be in the language L. Note that $1/2$ can be replaced with any positive constant: these can be made equivalent by the use of amplification techniques.

Definition 14. *BPP is the class of languages L with a polynomial-time probabilistic algorithm A such that*

$$x \in L \implies \Pr[A(x) \textit{ accepts} \geq \frac{2}{3}]$$
$$x \notin L \implies \Pr[A(x) \textit{ accepts}] \leq \frac{1}{3}.$$

This is 2-sided error. Note that the actual constants chosen $(\frac{2}{3}, \frac{1}{3})$ are arbitrary and can be chosen to be any distinct constants, again by amplification techniques. With a multiplicative overhead of $O(\log \frac{1}{\beta})$, we can get an error $\leq \beta$.

Clearly, $P \subseteq RP \subseteq BPP$, but whether $P = BPP$ is still an open question.

4 Derandomization via Enumeration

Finally, we explore derandomization via enumeration. The key idea is to simulate randomness by executing a large number of branches.

We are given a probabilistic algorithm A and input x. Consider the following algorithm: Run A on **every** possible random string of length $r(n)$, and output the majority answer.

The behavior matches the behavior of languages in BPP:

- If $x \in L$, then at least 2/3 of random strings will cause A to accept. Hence, the majority answer will be to accept.
- If $x \notin L$, then at least 2/3 of random strings will cause A to reject. Hence, the majority answer will be to reject.

The runtime is given by $O(2^{2(n)} \cdot t(n)) \leq O(2^{t(n)} \cdot t(n))$, where $t(n)$ is a time bound on A. Note that $r(n) \leq t(n)$ because every bit must be processed. However, we can improve runtime further if we could get a better bound on $r(n)$. For example, if $r(n) = O(\log n)$ and $t(n) = \textit{poly}(n)$, then the overall runtime would be $\textit{poly}(n)$.

Corollary 15. *$BPP \subseteq EXP$.*

The goal is to find ways to save random bits so that $r(n)$ is as small as possible.