CSE525: Randomized Algorithms and Probabilistic Analysis May 16, 2013

Lecture 13

Lecturer: Anna Karlin *Scribe: Noah Siegel, Jonathan Shi*

1 Random walks and Markov chains

This lecture discusses Markov chains, which capture and formalize the idea of a memoryless random walk on a finite number of states, and which have wide applicability as a statistical model of many phenomena. Markov chains are postulated to have a set of possible states, and to transition randomly from one state to a next state, where the probability of transitioning to a particular next state depends only on, and is determined fully by, the state that it is currently in. This last property of being memoryless and having the next state's probabilities determined only by the current state is called the "Markov property".

1.1 Random-walk-based algorithm

Consider 2-SAT, the problem of determining whether or not a boolean formula in conjunctive normal form, when there are at most two literals per clause, has a satisfying assignment.

The following algorithm is based on a random walk. First start with an arbitrary assignment to the n variables, and then, up to $100n^2$ times, if the formula is not satisfied, choose a random unsatisfied clause and a random variable from that clause, and flip the value of that variable. If at any point during the $100n^2$ iterations of this, if the formula is satisfied, then return that satisfying assignment. Else return "unsatisfiable".

In order to analyze this algorithm, we'll consider how likely it is to stumble upon a particular satisfying assignment S during $100n^2$ iterations. For each iteration t, consider X_t, the number of variables that agree with S immediately before the tth iteration of the algorithm. The algorithm can then be projected to a random walk on X_t: each time a variable is flipped, it either increases or decreases X_t by one. The algorithm succeeds if X_t ever reaches n.

Since the flipped variable at each step is one of two variables in an unsatisfied clause, and since every clause is satisfied in S, there is at least an one-half chance that the flipped variable started out disagreeing with S and flipped into a state that agreed with S. We express this as:

$$\Pr[X_{t+1} = i + 1 \mid X_t = i] \geq \frac{1}{2}.$$

Consider the pessimal version of this where $\Pr[X_{t+1} = i + 1 \mid X_t = 1]$ is exactly one half at every step (unless $X_t = 0$). This gives the process the property of being memoryless... the transitions at each step are independent of the history of previous states, so this pessimal version is a Markov chain. Also assume that $X_0 = 0$. We will show that in this case, $\mathrm{E}[\min t \mid X_t = n] = n^2$.

This implies that, if the formula is satisfiable, the expected number of iterations it'll take for the algorithm to succeed is at most n^2, so by doing $100n^2$ iterations, we obtain a 2^{-100} chance at most of this algorithm reporting a false negative.

To show that $\mathrm{E}[\min t \mid X_t = n] = n^2$, let h_j be the expected number of steps to reach n on our random walk when we start at j. We get a straightforward recurrence by expanding h_j in terms of the number of steps to reach n one step after we leave j:

$$h_j \;=\; 1 + \tfrac{1}{2}h_{j-1} + \tfrac{1}{2}h_{j+1} \quad\Longrightarrow\quad h_j - h_{j+1} \;=\; h_{j-1} - h_j + 2.$$

Using the base case that $h_0 - h_1 = 1$, a simple induction yields $h_j - h_{j+1} = 2j + 1$. Then, using $h_n = 0$, another simple induction gives us:

$$h_0 \;=\; h_n + \sum_{j=0}^{n-1} h_j - h_{j+1} \;=\; \sum_{j=0}^{n-1} 2j + 1 \;=\; n + 2\left(\frac{n(n-1)}{2}\right) \;=\; n^2.$$

1.2 Finite Markov chains

A finite Markov chain is a random walk on a directed graph. The directed graph has its edges labeled with transition probabilities, in a way so that the law of total probability holds (ie for each vertex, the sums of its outgoing edge labels is exactly 1). Each vertex in this graph is called a *state* of the Markov chain.

A Markov chain can equivalently be described as a *transition matrix* P, where P_{ij} is the probability of transitioning to state j if the preceding state is i. Thus if X_t denotes the state of the Markov chain at time t, we write $\Pr[X_{t+1} = j \mid X_t = i] = P_{ij}$. The property that P satisfies the law of total probability means that it is a *stochastic* matrix.

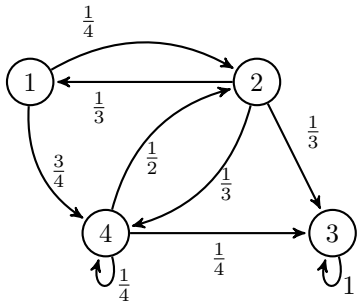

Figure 1: Markov chain, depicted in graph form.

	1	2	3	4
1	0	$\frac{1}{4}$	0	$\frac{3}{4}$
2	$\frac{1}{3}$	0	$\frac{1}{3}$	$\frac{1}{3}$
3	0	0	1	0
4	0	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$

Figure 2: Same Markov chain as a transition matrix.

Let $\mathbf{p}^{(t)}$ denote the probability distribution for the state of the Markov chain at time t, so that $\mathbf{p}_i^{(t)} = \Pr[X_t = i]$. Then $\mathbf{p}^{(0)}$ gives the initial probability distribution for the Markov chain is in, so that $\mathbf{p}^{(0)} = (1, 0, \ldots, 0)$ means to start in state 1, and $\mathbf{p}^{(0)} = (\frac{1}{n}, \ldots, \frac{1}{n})$ means to start in a uniformly random state.

The matrix formulation of a Markov chain gives us the following equivalences:

$$\mathbf{p}^{(t+1)} = \mathbf{p}^{(t)} P, \text{ and } \mathbf{p}^{(t+m)} = \mathbf{p}^{(t)} P^m.$$

Definition 1. *An* irreducible *Markov chain is one whose corresponding graph is strongly connected. The* period *of a state i in a Markov chain is the greatest common divisor of all $n \geq 1$ such that $(P^n)_{ii} > 0$. A Markov chain is* aperiodic *if the period of every state is 1.*

For example, a bipartite Markov chain is never aperiodic, and given any Markov chain P, we can construct an aperiodic one by taking $P' = \frac{1}{2}(I + P)$.

All the Markov chains we'll consider will be finite, irreducible, and aperiodic. This implies that there is some $N > 0$ such that for all $n \geq N$, the entries of the matrix P^n will be strictly positive (i.e. nonzero).

1.3 Stationary distributions

Definition 2. *A probability distribution π is a* stationary distribution *of a Markov chain with transition matrix P if $\pi = \pi P$.*

Consequently, for all states j, $\pi_j = \sum_i \pi_i P ij$.

A stationary distribution of P can be thought of as a fixed point, or an eigenvector of eigenvalue 1. The matrix P may have eigenvalues not equal to 1, but since the law of total probability must be satisfied, the corresponding eigenvectors will not be probability distributions (they must have negative components).

The following technical definitions and lemma will be needed to prove the Fundamental Theorem of Markov Chains later on:

Definition 3. *The* hitting time *T_{ij} is equal to $\min_{t \geq 1}\{X_t = j \mid X_0 = i\}$. That is, it is a random variable tracking the number of steps it takes to get to j from i.*

Let $h_{ij} = \mathrm{E}[T_{ij}]$. Then h_{ii} is the expected time of first return to state i after starting from state i.

Lemma 1. *For all states x and y in an irreducible Markov chain, $h_{xy} < \infty$.*

Proof There are $r \in \mathbb{N}$ and $\varepsilon > 0$ such that for all states z and w and for some $j \leq r$, it holds that $(P^j)_{zw} > \varepsilon$. This is true since we can guarantee a nonzero uniform lower bound for all nonzero entries in $\{(P^j)_{zw}\}$ just because there are a finite number of such entries, and we can guarantee that there exists a nonzero $(P^j)_{zw}$ for each pair of states z, w by taking a high enough r, because of irreducibility.

So consider a random walk on the Markov chain starting at x. Within the first r steps, there is at least a ε chance of the walk having reached y. Suppose it doesn't: then the walk is at some state $x_2 \neq y$ after the rth step. There is then at least a ε chance of reaching y in r more steps starting from x_2. Continuing on in this fashion, the expected value of T_{xy} is at least:

$$h_{xy} \;\geq\; r + (1-\varepsilon)r + (1-\varepsilon)^2 r + \ldots \;=\; r\sum_{k=0}^{\infty}(1-\varepsilon)^k \;=\; \frac{r}{\varepsilon}.$$

■

Theorem 1 (Fundamental Theorem of Markov Chains)**.** *For any finite, irreducible, aperiodic Markov chain:*

1. *There exists some stationary distribution π, and $\pi_x > 0$ for all states x.*
2. *π is unique.*
3. *$\pi_i = 1/h_{ii}$.*
4. *For all states i, j, $\lim_{t\to\infty}(P^t)_{ij} = \pi_j$.*

Proof of (1). For a random state z, define $\tilde{\pi}_y$ to be the expected number of times the random walk visits state y before returning to z. Then:

$$\tilde{\pi}_y \;=\; \sum_{t=1}^{\infty}\Pr_z[X_t = y \wedge (\forall u < t) X_u \neq z] \;\leq\; \sum_{t=1}^{\infty}\Pr_z[(\forall u < t) X_u \neq z] \;=\; h_{zz} \;<\; \infty.$$

We claim that $\tilde{\pi}$ is a fixed point of P. We verify this by calculating the yth component of $\tilde{\pi}P$, first by expanding the definition of $\tilde{\pi}_x$, then by substituting $\sum_x \Pr_z[X_t = x]P_{xy} = \Pr_z[X_{t+1} = y]$:

$$\begin{aligned}
(\tilde{\pi}P)_y &= \sum_x \tilde{\pi}_x P_{xy} \\
&= \sum_x \Big(\sum_t \Pr_z[X_t = x \wedge T_{zz} > t]\Big) P_{xy} \\
&= \sum_t \Big(\sum_x \Pr_z[X_t = x \wedge T_{zz} > t] P_{xy}\Big) \\
&= \sum_{t=0}^{\infty} \Pr_z[X_{t+1} = y \wedge T_{zz} \geq t+1] \\
&= \sum_{t=1}^{\infty} \Pr_z[X_t = y \wedge T_{zz} \geq t] \\
&= \tilde{\pi}_y - \Pr_z[X_0 = y \wedge T_{zz} > 0] + \sum_{t=1}^{\infty} \Pr_z[X_t = y \wedge T_{zz} = t] \\
&= \tilde{\pi}_y - \Pr_z[X_0 = y] + \Pr_z[X_{T_{zz}} = y] \\
&= \tilde{\pi}_y,
\end{aligned}$$

where the last equality is because we set $X_0 = z$, and so $X_{T_{zz}} = z$ since T_{zz} is the time it takes to get back to z.

So we concluded that $\tilde{\pi}P = \tilde{\pi}$, so all we have to do to get our stationary distribution is to normalize $\tilde{\pi}$:

$$\pi_x = \frac{\tilde{\pi}_x}{\sum_y \tilde{\pi}_y} = \frac{\tilde{\pi}_x}{h_{zz}}.$$

Proof of (2). Suppose π and $\tilde{\pi}$ are two stationary distributions of P and we'll show that $\pi = \tilde{\pi}$. Let $x = \text{argmin}_y \, \pi_y/\tilde{\pi}_y$, and let $c = \pi_x/\tilde{\pi}_x$. Then:

$$c = \frac{\pi_x}{\tilde{\pi}_x} = \frac{\sum_y \pi_y P_{yx}}{\sum_y \tilde{\pi}_y P_{yx}} \geq \frac{\sum_y c\tilde{\pi}_y P_{yx}}{\sum_y \tilde{\pi}_y P_{yx}} = c.$$

We substituted using $\pi_y \geq c\tilde{\pi}_y$ and found that that inequality is actually an equality. Therefore, since π and $\tilde{\pi}$ are both normalized and $\pi = c\tilde{\pi}$, we must have $c = 1$ and $\pi = \tilde{\pi}$.

Proof of (3). Remember from our construction of the stationary state π that:

$$\pi_x = \frac{\tilde{\pi}_x}{\sum_y \tilde{\pi}_y} = \frac{\tilde{\pi}_x}{h_{zz}}.$$

By uniqueness, we could have set $z = x$ and have come up with the same value for π_x. So then:

$$\pi_x = \frac{\tilde{\pi}_x}{h_{zz}} = \frac{\text{E}_x[\text{\# of visits to x before returning to x}]}{h_{xx}} = \frac{1}{h_{xx}}.$$

Proof of (4). Because P is irreducible, there exists an r such that all the entries of P^r are greater than zero. Thus, for some sufficiently small $\delta > 0$, $(P^r)_{xy} \geq \delta\pi_y$ for all states x, y.

Let $\underline{\Pi}$ the the matrix formed by using π as each of its rows, so that:

$$\underline{\Pi} = \begin{pmatrix} \longleftarrow \pi \longrightarrow \\ \longleftarrow \pi \longrightarrow \\ \vdots \\ \longleftarrow \pi \longrightarrow \end{pmatrix}.$$

Then we can define a stochastic matrix Q by setting:

$$P^r = \delta\underline{\Pi} + (1-\delta)Q,$$

where Q must be stochastic because it is a weighted average of stochastic matrices with nonnegative entries. Now, two properties of any stochastic matrix M are that $M\underline{\Pi} = \underline{\Pi}$, and if $\pi M = \pi$ then $\underline{\Pi}M = \underline{\Pi}$.

To show that $\lim_{t\to\infty}(P^t)_{xy} = \pi_y$ for all x, y, we will show that:

$$P^{rk} = (1-(1-\delta)^k)\underline{\Pi} + (1-\delta)^k Q^k.$$

This proceeds by induction on k. Since we can write $P^{rk} - \underline{\Pi} = (1-\delta)^k(Q^k - \underline{\Pi})$, and $\underline{\Pi}P = \underline{\Pi}$, we can multiply both sides by P^r and get:

$$\begin{aligned} P^{rk+r} - \underline{\Pi} &= (1-\delta)^k(Q^k P^r - \underline{\Pi}) \\ &= (1-\delta)^k(Q^k(\delta\underline{\Pi} + (1-\delta)Q) - \underline{\Pi}) \\ &= (1-\delta)^k(\delta Q^k\underline{\Pi} + (1-\delta)Q^{k+1} - \underline{\Pi}) \\ &= (1-\delta)^k((1-\delta)Q^{k+1} - (1-\delta)\underline{\Pi}), \\ &= (1-\delta)^{k+1}(Q^{k+1} - \underline{\Pi}), \end{aligned}$$

so that:

$$P^{r(k+1)} = (1-(1-\delta)^{k+1})\underline{\Pi} + (1-\delta)^{k+1}Q^{k+1}.$$

2 Maximum Matching in Regular Bipartite Graphs

Hall's marriage theorem guarantees that regular bipartite graphs (bipartite graphs where all vertices have the same degree) always have a perfect matching. In this section, we will give an algorithm for finding a perfect matching for a regular bipartite graph G with n nodes and m edges in $O(n \log n)$ expected time.

A partial matching of G is a set of edges with no shared nodes. Given a partial matching, an augmenting path for that matching is a path which starts and ends at two separate unmatched nodes and alternates between edges in the match and edges outside of it.

Theorem 2. *A matching M is maximal if and only if there is no augmenting path for M.*

Proof If there is an augmenting path to M, then we can use it to construct a bigger matching. Each node in M's path except for the ends was an endpoint of an edge in the matching, and the ends of the path were two different unmatched nodes, so each node is visited exactly once. Therefore, since the matched and unmatched edges in the augmenting path alternate, the unmatched edges form a matching. Since the path started and ended with unmatched edges, there is one more edge in this new matching than the original matching, making it larger than M and meaning M is not maximal. Therefore, M is maximal only if there is no augmenting path.

Next, we want to show that if there is no augmenting path for M, then M is maximal. Assume for contrapositive that M is not maximal. Then there exists some larger matching M' such that $|M'| > |M|$. Since M' contains at least one more edge than M, there are at least two more matched vertices in M' than in M. Start at some vertex matched in M' but not in M, and follow edges matched in one of them (which must alternate, since each vertex is matched to at most one other vertex in each matching) until either reaching a vertex matched in M' but not in M, or vice versa. If the vertex is matched in M' but not M, then this is an augmenting path for M. If the vertex is matched in M but not M', then this path had the same number of vertices matched in M as in M'. Since we know M' has more vertices matched than M, we will always eventually find an augmenting path for M. ■

The traditional approach to finding a perfect matching for G is the augmenting path algorithm: Repeatedly find an augmenting path alternating between matched and unmatched edges until no augmenting path can be found. This algorithm runs in $O(m)$ steps. We will use a randomized algorithm to improve on this time, running in expected time $O(n \log n)$.

We will still use the augmenting path strategy, but instead of performing breadth first search to find an augmenting path, we will use a random walk. On each iteration of the algorithm, we start with a partial matching M. Our graph is bipartite, so let L be one of the independent sets of nodes and R be the other. We will refer to L as being on the left, and R as being on the right. Construct a directed graph G' corresponding to our original graph G by having all edges in M go from R to L, and all other edges go from L to R. Add two vertices s and t such that there is an edge from s to each unmatched vertex in L for each edge out of it, and an edge from each unmatched vertex in R for each edge into it to t. Add edges from t to s equal to the number of edges out of s (which is equal to the number of edges into t). Contract each edge in M to a single vertex with the edges of both endpoints. By this construction, and because G is regular, each vertex in G' will have the same in-degree as out-degree, so G' is Eulerian.

G has an augmenting path if and only if there is a path from an unmatched vertex in L to an unmatched vertex in R in G', or equivalently by our construction, G' has a cycle from s to s.

Our algorithm will be to do a random walk starting from s. The expected time to find an augmenting path is the expected travel time from s to s, $E(T_{s,s}) = h_{s,s}$. As we showed earlier, $h_{s,s} = \frac{1}{\pi_s}$.

Claim 3. *In a Eulerian directed graph, the stationary distribution is*

$$\pi_v = \frac{d^{out}(v)}{m}$$

where d^{out} is the out degree of vertex v (which is equal to its in degree) and m is the total number of edges in the graph.

Proof

In the stationary distribution, π_v is the sum of in flow to v from all vertices:

$$\pi_v = \sum_{u|(u,v)\in E} \pi_u p_{uv} = \sum_{u|(u,v)\in E} \frac{d^{out}(u)}{m} \frac{1}{d^{out}(u)} = \frac{d^{out}(v)}{m} = \frac{d^{in}(v)}{m}$$

■

Therefore, $h_{s,s} = \frac{m}{d^{out}(s)}$.

In each iteration of the random walk, we add an edge to our matching. In iteration i, there are i edges in the matching, meaning that $d^{out}(s) \geq (n-i)d$, where d is the degree of G. Then $h_{s,s} \leq \frac{dn}{d(n-i)} = \frac{n}{n-i}$. Since we must find a cycle in each of the n iterations, the total running time of the algorithm is at most:

$$\sum_{i=0}^{n-1} \frac{n}{n-i} = O(n \log n)$$

The algorithm runs in $O(n \log n)$ time as desired.