Markov Chains and Random Walks

1 Fundamental Theorem of Markov Chains

We'll consider probability distributions $\boldsymbol{q}$ where q_i is the probability of being in state i (row vectors). We start with an initial distribution $\boldsymbol{q}_0$. The distribution one step later, starting from $\boldsymbol{q}$, is

$$\begin{aligned} q_i' &= \sum_j q_j p_{ji} \\ &= \boldsymbol{q}\mathbf{P}. \end{aligned}$$

So $\boldsymbol{q}_1 = \boldsymbol{q}_0\mathbf{P}$, $\boldsymbol{q}_2 = \boldsymbol{q}_0\mathbf{P}^2$, and therefore $\boldsymbol{q}_t = \boldsymbol{q}_0\mathbf{P}^t$. We write $\mathbf{P}_{i,j}^{(t)}$ for the (i,j) entry of $\mathbf{P}^t$ – the probability of transitioning from i to j in t steps.

Theorem 1. *[Fundamental Theorem of Markov Chains].* Let $\mathbf{P}$ be a finite irreducible, aperiodic Markov chain. Then,

1. *All states are persistent.*
2. $\mathbf{P}$ *has a unique stationary distribution* $\boldsymbol{\pi}$*, that is with* $\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}$*. Furthermore,*

   $$\boldsymbol{q}_0\mathbf{P}^t \to_{t\to\infty} \boldsymbol{\pi}$$

   for all distributions $\boldsymbol{q}_0$*.*
3. *At stationarity, he expected time to return to* i *after leaving it is* $1/\pi_i$*.*

Lemma 2. *If* G *is undirected, connected, and aperiodic, then the stationary distribution for a uniform random walk is* $\pi_i = \deg(i)/2m$ *for all* i*.*

Proof. Plug in $\boldsymbol{\pi}$ and verify that is stationary. We have

$$\begin{aligned} (\boldsymbol{\pi}\mathbf{P})_i &= \sum_j \pi_j \mathbf{P}_{ji} \\ &= \sum_j \frac{\deg(j)}{2m} \cdot \frac{1}{\deg(j)} \\ &= \sum_{j:\text{neighbor of } i} \frac{1}{2m} \\ &= \frac{\deg(i)}{2m}. \end{aligned}$$

The proof also shows that, in each step, each edge is traversed in each given direction with probability $1/2m$. □

Definition 3. The **hitting time** h_{ij} for a pair (i,j) is the expected number of steps, starting at i, to reach j for the first time. The **commute time** c_{ij} for a pair (i,j) is $c_{ij} = h_{ij} + h_{ji}$, the expected time to go back and forth. The **cover time** is the expected time to visit all states of the chain.

Lemma 4. *If (i,j) is an edge of an undirected graph G on which we perform a random walk, then $c_{ij} \leq 2m$.*

Proof. Consider the edge (i,j) in a fixed direction. At stationarity, the t^{th} step traverses (i,j) in that direction with probability $1/2m$. Consider the Markov chain whose states are the directed edges (i,j) – we have $2m$ states now. The probability of transitioning from (i,j) to (j,k) is $1/\deg(j)$. The stationary distribution of this Markov chain is $\pi_{(i,j)} = 1/2m$. By the fundamental theorem, once we leave (i,j), we return to (i,j) in expectation in $1/\pi_{(i,j)} = 2m$ steps.

Leaving the edge (i,j) means that we just traversed from i to j. Returning to the edge (i,j) means to make it back to i. So, this is a successful commute. (Only an upper bound of $2m$ because $i \to j$ might not use the edge (i,j)). □

Theorem 5. *If G is undirected and connected, its cover time is at most $2m(n-1)$.*

Proof. Consider any spanning tree T of G and any DFS order on T. For each edge (i,j) of T, we need to commute along it once, *i.e.,* traverse it once in each direction. So cover time is at most

$$\sum_{(i,j)\in T} c_{ij} \leq 2m(n-1).$$

□

2 An Algorithm for 2-SAT

We are given a 2-SAT formula Φ (each clause is an OR of at most two literals), and we'd like to find a satisfying assignment if one exists. This is polytime solvable based on a reduction to Eulerian paths.

We will use a simple local search algorithm: start with any assignment $\boldsymbol{x}$. While $\boldsymbol{x}$ is not satisfying Φ, we choose any unsatisfied clause c_j. Pick one of the (at most) two literals of c_{ij} uniformly at random and flip it. If no success after TBD steps, then declare unsatisfiable.

For the analysis we will assume Φ is satisfiable. Let $\boldsymbol{y}$ be any satisfying assignment. We would like to prove that our assignment becomes "more similar" to $\boldsymbol{y}$ over time. Measure similarity by the Hamming distance. Let $\boldsymbol{x}_t$ be the assignment of the algorithm after t steps. The distance

$$z_t := d_H(\boldsymbol{x}_t, \boldsymbol{y}) = \#\text{ of } i \text{ with } (\boldsymbol{x}_t)_i \neq \boldsymbol{y}_i.$$

Here, z_t is a random variable with $z_t \in \{0\ldots, n\}$. If $z_t = 0$ then the algorithm terminates. If $z_t \neq 0$, the algorithm might still terminate if it finds a satisfying assignment other than $\boldsymbol{y}$, but that only helps us. If $z_t \neq 0$ and $\boldsymbol{x}_t$ is not satisfying, then let c_t be an unsatisfied clause; the algorithm flips a variable in c_t uniformly at random.

Because $\boldsymbol{y}$ satisfies c_t, it differs from $\boldsymbol{x}_t$ in at least one variable. If we flip such a variable, $\boldsymbol{z}_t$ decreases by 1; otherwise, it increases by 1. So

$$\Pr[z_{t+1} = z_t - 1] \geq \frac{1}{2}$$
$$\Pr[z_{t+1} = z_t + 1] \leq \frac{1}{2}.$$

Therefore the time to hit 0 is upper bounded by the time for a uniform random walk on $\{0, 1, \ldots, n\}$ to hit 0, which is upper bounded by its cover time. By Theorem 5, the cover time is at most $2n^2$. By Markov's Inequality, $z_t = 0$ with probability at least $1/2$ within $4n^2$ steps. For higher probability, we can repeat. For 3-SAT, this is trickier because our probabilities are $1/3$ and $2/3$ instead of $1/2$, which removes the polynomial time guarantee.

3 Markov Chains for Sampling: Rapid Mixing

The main application is to sample complex objects such as graph colorings, points in high-dimensional polytopes, knapsack solutions, states of magnetic spin systems, etc. Our approach is to define a Markov chain on the objects. We start in a given state, run for "enough" steps to be close to a stationary distribution, and output the current object. We will design the chain such that the stationary distribution is what you want (usually uniform). This is also called Markov Chain Monte Carlo (MCMC).

Most early work was at IBM, in "card shuffling" – different chains for generating permutations. To generate a uniformly random permutation, you should pick a uniformly random element for the first position and repeat. But sometimes, all we can do is swap two random positions, swap a random position with an adjacent one, etc. How many swaps do we need to get "close" to a uniform distribution?

To talk about closeness to uniform distribution, we need a notion of distance between distributions. For now, we use the total variation distance (TVD):

$$d_{TV}(\mu_1, \mu_2) \coloneqq \max_{A \subseteq \Omega} |\mu_1(A) - \mu_2(A)|.$$

That is, the largest absolute difference in probability mass between μ_1 and μ_2 for any set A. It's easy to show that

$$\begin{aligned} d_{TV}(\mu_1, \mu_2) &= \frac{1}{2}\|\mu_1 - \mu_2\|_1 \\ &= \frac{1}{2}\sum_i |\mu_1(i) - \mu_2(i)|. \end{aligned}$$