Colorado CSCI 5454: Algorithms **Fall 2020**

Random Walks on Graphs

Instructor: Bo Waggoner *Lecture 11*

This lecture reviews adjacency matrices from a linear algebra perspective and discusses some uses, including counting paths and random walks. We will consider uses of random walks for sampling: the Markov Chain Monte Carlo method and a specific variant, Metropolis-Hastings.

Objectives:

- Be able to use powers of the adjacency matrix for counting paths.
- Understand the model of a finite Markov chain as a random walk on a graph; know PageRank.
- Understand the definition of a stationary distribution and when/why a random walk converges to stationary.
- Understand the point of Markov Chain Monte Carlo methods and an overview of how they work.

1 Adjacency Matrices for Counting Paths

Given a possibly-directed, unweighted graph $G = (V, E)$ with $|V| = n, |E| = m$, its **adjacency matrix** $A_G \in \{0,1\}^{n \times n}$ is the matrix

$$A_G(i,j) = \begin{cases} 1 & (i,j) \in E \\ 0 & \text{otherwise.} \end{cases}$$

Here $A_G(i,j)$ is the entry in the ith row and jth column. Note the diagonal entries are 0, assuming that as usual we have no self-edges.

Example: Consider the undirected "path graph" on 3 vertices that looks like this: $\circ\!\!-\!\!\!-\ \circ\ -\!\!\!-\!\!\circ$. That is, the two edges are $(1,2)$ and $(2,3)$. We have

$$A_G = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

In general a matrix $A \in \mathbb{R}^{n \times n}$ is a representation of a linear function from $\mathbb{R}^n$ to $\mathbb{R}^n$.

In this case, we can picture the vertices of the graph as the basis vectors in $\mathbb{R}^n$, i.e. the first vertex sits at coordinate $e_1 := (1, 0, \dots, 0)$, the second sits at $e_2 := (0, 1, 0, \dots, 0)$, and so on.[1] Then multiplying e_i with A_G maps a vertex i to the *sum* of all its neighbors in this space.

In the example, if we start at e_1, then we move to its only neighbor, i.e. $e_1 A_G = e_2$. But if we start at e_2, then we move to the sum of its neighbors, i.e. $e_2 A_G = (1, 0, 1) = e_1 + e_3$.

What happens if we apply A_G a second time? Then we move to all "neighbors of neighbors". In the running example, $e_2 A_G^2 = (e_2 A_G) A_G = (e_1 + e_3) A_G = e_1 A_G + e_3 A_g = 2e_2 = (0, 2, 0)$. More generally, if we apply A_G multiple times, say t, this is equivalent to multiplying by the t-th power of A_G, and we get a vector $v \in \mathbb{R}^d$, consisting of natural numbers, where the jth entry counts the number of paths (repeats allowed) from i to j.

Now, note that $e_i A$ is the same as picking out the ith row of A. So this is equivalent to:

[1] By default, we will assume vectors are row vectors, and write $v^\intercal$ for their transpose, i.e. column vectors.

Theorem 1. *$A_G^t(i,j)$ is equal to the number of paths from i to j in the graph G, with repeated vertices allowed, of length exactly t.*

Proof. By induction. For $t = 1$, the result is true: $A_G(i,j) = 1$ if there is an edge from i to j (hence a path of length one), and zero if not. Now suppose it is true for t, and consider $A_G^t = A_G^{t-1} A_G$. The (i,j) entry is the dot-product of the ith row of A_G^{t-1} with the jth column of A_G, i.e. $A_G^t(i,j) = \sum_{k=1}^n A_G^{t-1}(i,k) A_G(k,j)$. This is the sum, over all vertices k, of the number of length $t-1$ paths from i to k, times 1 if there is an edge from k to j or zero otherwise. This is precisely the number of length t paths from i to j. □

Exercise 1. Consider the undirected "bowtie" graph on 5 vertices $V = \{1, 2, 3, 4, 5\}$ where $(1, 2, 3, 1)$ is a cycle and $(3, 4, 5, 3)$ is a cycle. (In other words, two triangles, sharing a vertex 3.) Compute A_G^2 and A_G^3, and check that this counts the number of paths.

Comments:

- We can generally compute A_G^t in $\log t$ time, which can be useful for large t as compared to the size of the matrix.
- For large graphs, exactly computing matrix multiplication can be computationally intensive, but this can be a useful algorithm as well as a stepping stone to more advanced or approximate techniques.

Exercise 2. How do we compute A_G^t in $O(\log t)$ time? (You may treat the size of the matrix as a constant and assume that all arithmetic operations fit in the machine's word size.)

Hint: first, suppose t is a power of 2. Can you do it now? Second, write t in binary, i.e. as a sum of powers of 2.

2 The Normalized Adjacency Matrix and Markov Chains

The **normalized** adjacency matrix of a graph G on n vertices is $W_G \in \mathbb{R}^{n \times n}$ defined by

$$W_G(i,j) = \frac{1}{\text{degree}(i)} A_G(i,j).$$

In other words, we normalize each row of A_G so that it sums to 1.

In the example from above of the graph ○— ○ —○, we have

$$W_G = \begin{pmatrix} 0 & 1 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \end{pmatrix}.$$

W_G maps a vertex to the *average* of all its neighbors in this space, i.e. $e_i W_G = \frac{1}{\text{degree}(i)} \sum_{j:(i,j)\in E} e_j$.

In the running example, if we start at the point e_1 corresponding to vertex one, then we move to its only neighbor, i.e. $e_1 W_G = e_2$. But if we start at e_2, we move to the midpoint between e_1 and e_3, i.e. $e_2 W_G = \frac{e_1}{2} + \frac{e_3}{2} = (0.5, 0, 0.5)$.

2.1 As a Random Walk (Markov Chain)

Imagine you start at a vertex i of the graph and randomly select a neighbor (uniformly, with equal probability), and move to that neighbor. Then the row $W_G(i)$ gives the induced probability distribution over these choices.

Indeed, if we have any vector $v \in \mathbb{R}^n$ with $v_i \geq 0, \sum_{i=1}^n v_i = 1$, this gives a probability distribution over the vertices oGImagine sampling a vertex from that distribution, then randomly picking an edge and following it. What is the induced distribution on vertices? Well, the probability of landing on i is

$\sum_{j:(j,i)\in E} v_j \frac{1}{\text{degree}(j)} = (vW_G)_i$. That is, the ith component of the vector we get from multiplying v into W_G.

This is an example of a **Markov chain** on a finite state space with transition matrix W_G. In a Markov chain on a finite state space $\{1,\dots,n\}$, we have a *transition matrix* $M \in \mathbb{R}^{n\times n}$, where $M(i,j)$ is the probability of transitioning from i to j. Each row must be a probability distribution, i.e. nonnegative entries summing to 1. This says that, starting at i, we pick the next step of the random walk (or the next *state* of the Markov chain) from the probability distribution $M(i,1),\dots,M(i,n)$.

Next question: what happens if we iterate this random walk for a long, long time? I.e., what does the limit look like of vW_G^m as $m \to \infty$? Does it matter what the starting distribution v is?

2.2 Stationary Distributions and Eigenvalues

A **stationary distribution** $\pi \in \mathbb{R}^n$ of a Markov chain with transition matrix M is a probability distribution satisfying

$$\pi M = \pi.$$

In other words, if we draw a random vertex from π, then take a random step from the vertex, the distribution of our final endpoint is again π.

From linear algebra, we recall[2] such a vector is called a **left eigenvector** with corresponding **eigenvalue** 1. Given a transition matrix M, we name its n eigenvalues $\{\lambda_i\}$ and sort them from largest to smallest: $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. The list $\lambda_1,\dots,\lambda_n$ is called the **spectrum** of M; this is why this area of research is called "spectral" graph theory.

In particular, of course, the normalized adjacency matrix W_G is a type of Markov transition matrix. The following gives some basic known facts about its eigenvalues.

Claim 1. *The largest left eigenvalue of W_G is exactly 1, and the smallest is at least -1 (which is achieved if and only if G is bipartite). The multiplicity of 1, i.e. the number of eigenvalues that are 1, is equal to the number of connected components of the graph.*

Because the largest left eigenvalue is 1, we know that **a random walk on a graph has at least one stationary distribution**. That is, there is some π such that $\pi W_G = \pi$.

Exercise 3. For the running example graph, ○—○—○, can you find a stationary distribution? Confirm that $\pi W_G = \pi$.

3 Convergence to Stationary

Next, we will make some strong assumptions that imply there is exactly one stationary distribution and the distribution of a random walk, over a long time horizon, converges to it. This is just an example to show the flavor of these theorems and proofs in spectral graph theory; most such proofs are a bit more intricate and advanced, usually going to a related matrix called the graph Laplacian (which we won't need here).

Claim 2. *Suppose that every entry of the transition matrix M is strictly positive; then its eigenvalues satisfy $\lambda_1 = 1$ and $|\lambda_i| < 1$ for all $i = 2,\dots,n$.*

Now consider starting from some distribution $p^{(0)}$ over the vertices and repeatedly applying a transition matrix M, so that after t steps, we get $p^{(t)} := p^{(0)}M^t$.

Theorem 2. *Suppose every entry of the transition matrix M is strictly positive and suppose that M is a diagonalizable matrix. Let $\sigma = 1 - \max_{i=2,\dots,n} |\lambda_i|$. Then there exists a constant C such that for all t, we have*

$$\|\pi - p^{(t)}\|_1 \leq C \cdot n \cdot e^{-t\sigma}.$$

[2] In general we have a left eigenvector v with eigenvalue λ if $vW_G = \lambda v$.

(Recall that $\|x\|_1 = \sum_i |x(i)|$.)

Proof. Let the eigenvectors of M be $\pi, x_2, \ldots, x_n$ and suppose without loss of generality that each vector has $\|x_j\|_1 = 1$. Because M is diagonalizable, its eigenvectors $\pi, x_2, \ldots, x_n$ span all of $\mathbb{R}^n$, i.e. they are linearly indpendent (note we are not assuming they are orthogonal!). So any starting point $p^{(0)}$, we can write it as a linear combination of eigenvectors $\pi, x_2, \ldots, x_n$:

$$p^{(0)} = c_1\pi + \sum_{j=2}^{n} c_j x_j.$$

Note the coefficients c_i may be positive or negative. Then

$$\begin{aligned} p^{(t)} &= p^{(0)} M^t \\ &= \left(c_1\pi + \sum_{j=2}^{n} c_j x_j \right) M^t \\ &= c_1\pi + \sum_{j=2}^{n} c_j x_j \lambda_j^t. \end{aligned}$$

Now, we argue that $c_1 = 1$. Because we have $|\lambda_j| < 1$ for $j \geq 2$, the entire sum is converging to $\vec{0}$ as $t \to \infty$. So we have $p^{(t)} \to c_1\pi$, and because both are probability distributions, we must have $c_1 = 1$. So

$$p^{(t)} = \pi + \sum_{j=2}^{n} \alpha_j x_j \lambda_j^t.$$

Therefore, if we let $C = \max_{j=2,\ldots,n} |c_j|$, then

$$\begin{aligned} \|\pi - p^{(t)}\|_1 &= \left\| \sum_{j=2}^{n} c_j x_j \lambda_j^t \right\|_1 \\ &\leq \sum_{j=2}^{n} |c_j| \cdot |\lambda_j|^t \cdot \|x_j\|_1 && \text{triangle inequality} \\ &\leq \sum_{j=2}^{n} C(1-\sigma)^t && \text{because } \|x_j\|_1 = 1 \\ &\leq C \cdot n \cdot (1-\sigma)^t \\ &\leq C \cdot n \cdot e^{-\sigma t}, && \text{because } 1-\sigma \leq e^{-\sigma} \end{aligned}$$

□

To step back and appreciate this theorem: Even if the number of vertices of the graph is gigantic, say $n = 2^d$ for some d, we can get convergence very close to the stationary distribution in only $O(d/\sigma)$ steps of the random walk. As long as σ isn't too tiny, this is a small number of steps compared to n.

Exercise 4. Suppose M satisfies the assumptions of Theorem 2, i.e. has all positive entries and is diagonalizable. Use Theorem 2 to argue that M has a single unique stationary distribution.

4 PageRank

The key idea of PageRank is to let G be the directed graph of hyperlinks on the web, where each vertex is a webpage with edges to every page it has links to.

Now we can imagine the Markov chain (random walk) of the normalized adjacency matrix W_G. But in PageRank, we make the following modification: with probability α, we jump to a new, completely uniformly random webpage. With probability $1-\alpha$, we follow a random link on the current page.

This gives rise to the following transition matrix:

$$M(i,j) = \frac{\alpha}{n} + (1-\alpha) W_G(i,j).$$

One nice consequence is that, for $\alpha > 0$, every entry is strictly positive and there is a unique stationary probability distribution π of this random walk. This is the **PageRank** of G. In particular, for each page i, its rank or score is $\pi(i)$, with larger being better.

π has the following nice property, inherited from the equation $\pi M = \pi$:

$$\begin{aligned} \pi(j) &= \sum_{i=1}^{n} \pi(i) M(i,j) \\ &= \frac{\alpha}{n} + (1-\alpha) \sum_{i:(i,j)\in G} \frac{\pi(i)}{\text{degree}(i)}. \end{aligned}$$

This says the asymptotic probability of being on page j is the sum of two processes:

- With probability α, no matter where we are, we jump randomly, in which case there is a $\frac{1}{n}$ chance of landing on page j.
- With probability $1-\alpha$, we are on page i with probability $\pi(i)$, and we jump to page j with probability $\frac{1}{\text{degree}(i)}$ if there is a link (i,j).

Now, with billions of webpages, we cannot even store G in memory as an adjacency matrix, let alone compute its powers exactly, but we still have ways to approximately sample from π; we'll look at this next.

5 Markov Chain Monte Carlo

We now will look briefly at a more general, powerful technique. The idea is that we need to sample or estimate an integral from a challenging distribution over a very large space. We can't write down the distribution, but we have some info about it.

For example, suppose we want to sample a web page with probability proportional to the number of words on the page. So if $f(j)$ is the number of words on page j, we want $\pi(j) = \frac{f(j)}{\sum_i f(i)}$. But the denominator is expensive to compute, e.g. because of how many web pages there are.

(One could also ask about computing an expectation or integral of some function with respect to this difficult distribution.)

Claim 3. *Let $p^{(0)}$ be any distribution and $p^{(t)} = p^{(0)} M^t$, where M is the transition matrix of a Markov chain on a finite space. If M is* connected *(meaning any state i is reachable from any state j), then it has a unique stationary distribution π, and as $t \to \infty$, the* average *$\frac{p^{(1)}+\cdots+p^{(t)}}{t}$ converges to π.*

This suggests the general recipe for sampling once from π:

- Pick a vertex $v^{(0)}$ from some distribution $p^{(0)}$.
- Take steps according to M, obtaining $v^{(1)}, \dots, v^{(t)}$.
- Pick v *uniformly at random* from $v^{(1)}, \dots, v^{(t)}$.

If the task is drawing many samples from π (which is more common), one can take the entire sequence of samples $v^{(1)}, \dots, v^{(t)}$. These will be correlated with each other (for example $v^{(t-1)}$ and $v^{(t)}$ are not at all independent), but as a whole they will constitute a representative sample, for large enough t.

Exercise 5. One thing we should not do is just take the final sample $v^{(t)}$. Why not?

Hint: consider a bipartite graph and suppose t is even.

The details of how to implement this recipe depend on the setting. In some cases M may be hard to compute or sample from, and more work is needed. Next, we will look at a setting where the target distribution π is "known" in a sense, but it's so large that sampling from it is hard. So, we will set up a Markov chain and execute the above recipe.

5.1 Metropolis Hastings

For a running example, think of a grid of points in high dimensional space, for example, the Boolean hypercube $\{0,1\}^k$. (In other words, each "vertex" is labeled by a string of length k of zeros and ones; there are 2^k vertices.)

For this algorithm, we suppose that someone gives us a likelihood or weight function $f : \{1,\dots,n\} \to \mathbb{R}_+$, and asks us to sample points with probability proportional to f. So in this case we know the probability distribution at each state u: it is

$$\pi(u) := \frac{f(u)}{\sum_v f(v)}. \tag{1}$$

The problem is that n is too large to compute the sum efficiently (and even if we knew it, it's still not obvious how to sample).

The idea is that we can construct an undirected graph on $\{1,\dots,n\}$ and run a Markov chain that converges to this stationary distribution π, without ever writing down π.

The first step is to come up with a graph. We want the following properties, for reasons we'll see.

- We want the *mixing time* to be small, meaning that the random walk converges fast.
- We want the maximum degree, call it r, to to be relatively small.
- We want vertices to have edges to other vertices with similar "weight" $f(v)$.

For the Boolean hypercube example, perhaps we create an edge (u,v) if we can get from u to v by flipping one bit of the string. In some applications, it's reasonable that if u and v are the same string except for one bit, then their weights $f(u), f(v)$ aren't too different. The maximum degree is also only k if we are in k dimensions, which is much smaller than the total number of vertices 2^k.

Algorithm 1 Metropolis-Hastings on finite graph. The goal is to sample vertices with probability proportional to f.

```
Input: oracle access to $f : \{1,\dots,n\} \to \mathbb{R}_+$; oracle access to adjacency list of $G$ on vertices $\{1,\dots,n\}$;
  maximum degree $r$ of $G$.
Let $u^{(0)}$ be chosen uniformly at random from $\{1,\dots,n\}$
for some number of trials $t = 1,\dots,T$ do
  Let $v_1,\dots,v_\ell$ be the $\ell$ neighbors of $u^{(t-1)}$
  Let $v = \begin{cases} v_i & \text{w.prob. } \frac{1}{r} \\ u^{(t-1)} & \text{w.prob. } \frac{r-\ell}{r} \end{cases}$
  if $f(v) \geq f(u^{(t-1)})$ then
    Set $u^{(t)} = v$.
  else
    Set $u^{(t)} = \begin{cases} v & \text{w.prob. } \frac{f(v)}{f(u^{(t-1)})} \\ u^{(t-1)} & \text{o.w.} \end{cases}$.
  end if
end for
```

We can see that Algorithm 1 gives a Markov chain with

$$M(u,v) = \frac{1}{r}\min\left\{1, \frac{f(v)}{f(u)}\right\}$$

if (u,v) is in the graph, and $M(u,v) = 0$ otherwise.

To show that it has the target stationary distribution, we need the following lemma.

Lemma 1. *For a Markov chain with transition matrix M, if strongly connected, if there exists a distribution π satisfying*

$$\pi(i)M(i,j) = \pi(j)M(j,i),$$

then π is the unique stationary distribution.

Note that this does not claim the stationary distribution always satisfies this relationship. For example, in some graphs we may have $M(i,j) > 0$ while $M(j,i) = 0$, so it is impossible to satisfy. However, Lemma 1 is useful because if we *are* able to construct π satisfying the condition, we know it is the stationary distribution.

Theorem 3. *If the Metropolis-Hastings graph is connected, then the stationary distribution of the Markov chain with transition matrix M is π (Equation 1).*

Proof. We just need to verify that the conditions of Lemma 1 are satisfied by the target distribution π. Consider any edge (u,v) and suppose without loss of generality that $f(u) \geq f(v)$. Then $M(u,v) = \frac{1}{r}\frac{f(v)}{f(u)}$, while $M(v,u) = \frac{1}{r}$.

So $\pi(u)M(u,v) = \frac{f(u)}{\sum_{u'} f(u')}\frac{1}{r}\frac{f(v)}{f(u)} = \frac{f(v)}{r\sum_{u'} f(u')}$.

Meanwhile, $\pi(v)M(v,u) = \frac{f(v)}{\sum_{u'} f(u')}\frac{1}{r}$.

These are equal, so the conditions of Lemma 1 are satisfied. □

To conclude: Metropolis-Hastings gives us a general way to construct a Markov chain such that a sample from it converges to the target distribution π that we wanted to sample from. Of course there are lots of design questions left: how to chose the graph G and when you are able to access such a function f. The answers to these vary depending on the problem being solved in practice.

References

Strang, *Linear Algebra and Its Applications*, Ch5.