Lecture 7

1 Summary

In this lecture, we theoretically analyze the bias introduced by traceroute sampling methods. For the analysis, we assume that the sampling is done using a breadth first search from a single monitor node. A surprising consequence of the analysis is that the degree distribution estimated by the sampling method on a randomly chosen d-regular graph follows a power law with high probability. This points to the fact that there is a significant bias in the estimate for the degree distribution if we use such traceroute sampling methods.

2 Bias in Traceroute Sampling

2.1 Problem Definition

We begin by introducing some notation.

- The input graph for traceroute sampling is denoted by G.
- Let $\bar{d} = \{d_1, d_2, \ldots, d_n\}$ be a degree sequence over n nodes. We assume that the graph G is given by $G_{n,\bar{d}}$. Thus G is randomly chosen from the set of graphs with n nodes and degree sequence $\bar{d}$.
- There is a single monitor node m. All other nodes of G are target nodes.
- $traceroute(m, t)$ finds the shortest path from the monitor node m to a target node t.
- Let T denote the shortest path tree obtained as a result of finding the $traceroute(m, t)$ for each node t in G.

Problem Statement: Compute the degree distribution of T and compare it with G.

2.2 Analysis

The degree of any node in G is a positive integer less than n. This allows us to represent the degree sequence $\bar{d}$ as a sequence $\{a_1, a_2, \ldots, a_n\}$, where a_k denotes the probability that a randomly chosen node from G has degree k.

$$a_k = \frac{\#\{v : deg(v) = k\}}{n}$$

We denote the sequence $\{a_1, a_2, \ldots, a_n\}$ as $\bar{a}$. We require that the degree sequence of G be *reasonable*. The definition of a reasonable degree sequence follows:

Definition 1. *A degree sequence $\bar{a}$ is reasonable iff*

- $a_k = 0$ *for* $k < 3$

- $\exists \alpha > 2, c > 0$ *such that* $a_k < ck^{-\alpha}$ *for all* $k \geq 3$

Theorem 1 (Main)**.** *Let $\bar{d}$ be a degree sequence such that corresponding $\bar{a}$ is reasonable and let $G = G_{n,\bar{d}}$ be the graph over which trace route sampling is done. Let T be the shortest path tree obtained. If $A_k^{obs} = \#\{v : deg_T(v) = k\}$ then there exists $\delta > 0$ such that with high probability $|A_k^{obs} - na_k^{obs}| = o(n^{1-\delta})$ for all k where*

$$
\begin{aligned}
a_{m+1}^{obs} &= \sum_i a_i \left[\int_0^1 it^{i-1} \binom{i-1}{m} p_{vis}(t)^m (1 - p_{vis(t)})^{i-m-1} \right] \\
p_{vis}(t) &= \frac{1}{\sum_j ja_j t^j} \sum_k ka_k t^k (\frac{\sum_j ja_j t^j}{dt^2})^k
\end{aligned}
$$

Intuition: Theorem 1 relates the observed degree sequence $\bar{a}^{obs}$ with the correct degree sequence $\bar{a}$. It shows that the observed and the correct degree sequence may be quite different. For example consider the sequence $\bar{a}$ corresponding to a 3-regular graph. Theorem 1 shows the observed degree $\bar{a}^{obs}$ sequence for a 3-regular graph is $\{1/3, 1/3, 1/3, 0, 0, \ldots, 0\}$ which can be thought of as following a power law.

Proof of Theorem 1: The key to the analysis is choosing the right generation process for the random graph. Given the degree sequence $\bar{d} = \{d_1, d_2, \ldots, d_n\}$ the graph is generated as follows:

- For each node $i \in [n]$ make d_i copies.
- For each copy c, compute x_c a uniformly chosen r.v in $[0, 1]$.
- Initialize a queue and enqueue all the copies of the monitor node.
- Use the following iterative process to maintain the queue:
  - Dequeue the copy from the front of the queue
  - Match it to the copy c with the highest x_c.
  - If c is a copy of a unvisited vertex u, enqueue all other copies of u.

It is easy to see that the above process gives a uniformly random matching on the copies. Let G be the graph obtained. The relationship between G and T is simple and given below .

Claim 2. *An edge $e = (u, v)$ of G is created when a copy c of u is popped from the queue and matched with a copy of v. It appears in T iff v is unvisited (not in the queue) when c is popped from the queue.*

Another useful way to think about the above process is to imagine it using *time*. Let $t \in [0, 1]$ be a monotonically increasing variable which in some sense represents the time at any instant. At time t check if a copy c has $x_c = t$. If true then match c with the copy from the front of the queue. In addition, if c is unvisited then enqueue all siblings of c. This representation of random process allows us to define the following random variables.

- $A(t)$ = number of unmatched copies at time t. Note that $\mathbf{E}[A(t)] = dn\Pr$[c is unmatched at time t] $= dnt^2$. Moreover the actual value of $A(t)$ is w.h.p within $o(\sqrt{n})$ from $\mathbf{E}[A(t)]$.
- $B(t)$ = number of unvisited copies at time t. Note that probability that a copy of vertex of degree k is unvisited at time t is simply t^k. Thus $\mathbf{E}[B(t)] = \sum_k ka_k nt^k$. Moreover the actual value of $B(t)$ is w.h.p within $o(n^{1-\beta})$ (for some constant β) from $\mathbf{E}[B(t)]$.
- $v_j(t)$ = number of vertex of degree j unvisited at time t. Note that $\mathbf{E}[v_j(t)] = a_j nt^j$. Moreover the actual value $v_j(t)$ is w.h.p within $o(\sqrt{n})$ from $\mathbf{E}[v_j(t)]$.

Thus for $A(t)$, $B(t)$ and $v_j(t)$ their expected values give a good approximation to their true values, w.h.p. We will use this fact to simplify expressions involving these random variables.

Next we compute the probability that a degree k vertex v has a degree l in the tree T given that v is visited at time t. Let this probability be denoted as $P_{t,k,l}$. To compute it, we use the following property of the random process: When v is visited for the first time, all the copies of v are enqueued. All edges of v are decided by matching a copy of v with a copy of some node w. If the matched node w is already visited then the edge (v, w) occurs in G but not in T. If the matched node w is unvisited then the edge (v, w) occurs in both G as well as T.

Using this property we compute the probability that an edge (v, w) of G is also present in T given that v is visited at time t. Denote this probability as $p_{vis}(t)$. This is equivalent to probability that w is unvisited at the time when it is matched with a copy of v from the queue. This means that w should have been unvisited at time t (when v was visited). The probability of this happening is simply $\frac{B(t)}{A(t)}$. Moreover, when at time t the copies of v were enqueued, there might be copies of other nodes already in the queue. w should remain unvisited as the copies ahead of the copies of v are matched. This happens when all the copies of w are eventually matched with copies of nodes that were visited after time t. Thus

$$p_{vis}(t) = \frac{B(t)}{A(t)} \sum_j \frac{jv_j(t)}{B(t)} (\frac{B(t)}{A(t)})^{j-1} \tag{1}$$

$$\sim \frac{1}{\sum_j ja_j t^j} \sum_k ka_k t^k (\frac{\sum_j ja_j t^j}{dt^2})^k \tag{2}$$

Eq 2 occurs w.h.p and is obtained by replacing $A(t)$, $B(t)$ and $v_j(t)$ with there expected values. $P_{k,t,l}$ is the probability that l-1 of the k-1 nodes w were unvisited at the time copies of v were being matched. This is simply the binomial distribution with parameters $k-1$ and $p_{vis}(t)$. Thus $P_{k,t,l} = \binom{k-1}{l-1} p_{vis}(t)^{l-1}(1 - p_{vis}(t))^{k-l}$. Integrating over t gives the desired result proving Theorem 1. $\square$

2.3 Regular Graphs

If the graph is Δ-regular then the expressions for $\bar{a}^{obs}$ can be simplified.

$$\begin{aligned} a^{obs}_{m+1} &= \sum_i a_i \left[\int_0^1 it^{i-1} \binom{i-1}{m} p_{vis}(t)^m (1 - p_{vis(t)})^{i-m-1} \right] \\ &= \sum_i \int_0^1 \binom{i-1}{m} x^{(\Delta-2)(l-1)} (1 - x^{\Delta-2})^{i-l} \end{aligned}$$

For a 3-regular the expression gets simplified to $\sum_i \int_0^1 \binom{i-1}{m} x^{(l-1)} (1-x)^{i-l}$. This gives the degree sequence $\bar{a}^{obs} = \{1/3, 1/3, 1/3, 0, 0, \ldots, 0\}$

3 Further reading

D. Achlioptas, A. Clauset, D. Kempe, and C. Moore, On the bias of Traceroute sampling, STOC'05.

`http://www.cs.ucsc.edu/~optas/papers/traceroute.pdf`