CS 6850	Some Basic Calculations on Random Graphs
Jon Kleinberg	

A First Random Graph Model

In the most heavily-studied model of random graphs, we start with n nodes and join each pair by an undirected edge, independently with probability p. We will call this model $\mathcal{G}_{n,p}$.

If G is a graph generated using $\mathcal{G}_{n,p}$, we can compute the expected degree of one of its nodes v using linearity of expectation. Let X_v be a random variable denoting the degree of v, and for each other node w, let $X_{v,w}$ be a random variable equal to 1 if there is an edge joining v and w, and equal to 0 otherwise. We have

$$\begin{aligned} X_v &= \sum_w X_{v,w} \\ E[X_v] &= \sum_w E[X_{v,w}] \\ &= \sum_w p = (n-1)p. \end{aligned}$$

So if we think of p as $\frac{c}{n-1}$ for some quantity c, then the expected degree of v is c.

Now, if c is a constant, then despite the constant expected degree in G, there will still be many isolated nodes (that is, nodes with no incident edges). To see this, let $\mathcal{E}_v$ denote the event that v is isolated; this requires that each of its $n-1$ potential incident edges not be present, so we have

$$\begin{aligned} \Pr[\mathcal{E}_v] &= (1-p)^{n-1} \\ &= \left(1 - \frac{c}{n-1}\right)^{n-1} \\ &= \left(\left((1 - \frac{c}{n-1}\right)^{\frac{n-1}{c}}\right)^c. \end{aligned}$$

Now, the part inside the outermost parentheses on the last line is between $1/4$ and $1/e$ as $\frac{c}{n-1}$ ranges from $\frac{1}{2}$ down toward 0. Thus, $\Pr[\mathcal{E}_v]$ is between 4^{-c} and e^{-c}, which is a constant when c is constant.

Using this, we can ask how large c needs to be in order for there to be a high probability of no isolated nodes. Let $\mathcal{E}$ be the event that there is any isolated node in G; then by the Union Bound we can write

$$\begin{aligned} \mathcal{E} &= \bigcup_v \mathcal{E}_v \\ \Pr[\mathcal{E}] &\leq \sum_v \Pr[\mathcal{E}_v] \\ &\leq ne^{-c}. \end{aligned}$$

Now we choose c large enough so that e^{-c} is small enough to cancel n. In particular, $c = \ln n$ is not quite enough, but $c = 2 \ln n$ will easily do it:

$$\Pr[\mathcal{E}] \leq ne^{-2 \ln n} = n \cdot n^{-2} = n^{-1}.$$

Thus, $\mathcal{G}_{n,p}$ is not an appropriate model for considering random graphs in which all degrees are positive, yet constant — as we've just seen, the average degree in $\mathcal{G}_{n,p}$ needs to become logarithmic before the last isolated node is likely to vanish. We now turn to a model in which the degrees are explicitly fixed to whatever values we choose.

Random Graphs with Fixed Degrees

Suppose we want to completely specify the sequence of degrees in our random graph; that is, we want the n nodes to have degrees $d_1, d_2, \ldots, d_n$, and then produce a random graph subject to this constraint. We'll allow graphs that have self-loops (an edge goes from a node to itself) and parallel edges (two edges connect the same pair of nodes), which will make it much easier to construct random graphs with the desired properties. We'll also assume $\sum_i d_i$ is an even number, without which no graph with the desired degrees can exist.

One way to create a random graph with the specified degrees is to first define a set of nodes labeled $1, 2, \ldots, n$ in which node i has d_i "half-edges" sticking out of it. We then simply choose a random pairing on all $\sum_i d_i$ half-edges, glue the paired half-edges together, and declare the resulting graph to be our random graph G. (Note how self-loops and parallel edges may indeed arise from this construction.)

Now, let's consider the special case of this construction in which all d_i are equal to some constant $d > 0$. Graphs produced by this special case of the construction are called *random d-regular graphs*, meaning that all nodes have degree d. When $d = 1$, the only possible graph is a perfect matching (i.e. a collection of disjoint edges), and when $d = 2$, the only possible graphs are collections of disjoint cycles. But things get much more interesting once $d = 3$, and one of the fundamental properties of a random 3-regular is that it has good *expansion*.

Expansion. The expansion of a graph is the minimum "surface-to-volume" ratio of any set of nodes. More precisely, if we use $|S|$ to denote the size of a set of nodes S, use $\overline{S}$ to denote the complement of a set of nodes S, and use $e^{out}(S)$ to denote the set of edges with exactly one end in S, then the expansion α of a graph G is defined as

$$\alpha = \min_{S \subseteq V} \frac{|e^{out}(S)|}{\min(|S|, |\overline{S}|)}.$$

So we look at the number of edges crossing a cut from S to $\overline{S}$, and we compare it to the size of the smaller side. The worst such bottleneck in the graph is the expansion. Given that we're only interested in the smaller sides of cuts, another way to write expansion is by explicitly minimizing only over "small" S:

$$\alpha = \min_{|S| \leq n/2} \frac{|e^{out}(S)|}{|S|}.$$

Here the basic fact about random d-regular graphs:

> *For each $d \geq 3$, there is a constant α depending only on d, such that a random d-regular graph (of any size) has expansion at least α.*

That is, a random d-regular has constant expansion, regardless of how large it is. While we won't go into this further right now, it's striking that a random construction produces constant-degree graphs with constant-expansion so easily, given that it's very hard to explicitly describe (by a deterministic construction) a family of constant-degree graphs that retain constant expansion as the number of nodes goes to infinity.

Expansion implies short paths. Intuitively, expansion implies a strong "robustness" to the graph: to split it into multiple large pieces, one must destroy correspondingly many edges. This property has many other consequences; later in the course, for example, we'll that it implies that a random walk on the graph "mixes" rapidly (approaching its stationary distribution in a small number of steps). For right now, we prove the following "small-world" property of graphs with good expansion:

> *If an n-node graph of maximum degree d has expansion at least α, then every pair of nodes s and t is connected by a path of length at most $O(\frac{d}{\alpha} \log n)$.*

To prove this, we try constructing a path from s to t using breadth-first search (BFS). Let S_j be the set of nodes encountered anywhere in the first j levels of the BFS outward from s. To determine the next, $(j+1)^{\text{st}}$, level of the BFS, we need to follow all the edges out of S_j; the nodes that these edges lead to, together with S_j, will form the set S_{j+1}.

As long as S_j consists of fewer than $n/2$ nodes, the expansion of G implies that it has at least $\alpha|S_j|$ edges leading out of it. Some of these edges may lead to the same nodes, but since no node has degree more than d, we can conclude that at least $\frac{\alpha}{d}|S_j|$ new node are discovered by looking one more BFS level out from S_j. In other words,

$$|S_{j+1}| \geq (1 + \frac{\alpha}{d})|S_j|.$$

This says that the BFS layers out from s grow exponentially, due to the expansion of G, and so as long as S_j has fewer than half the nodes, we have

$$|S_j| \geq (1 + \frac{\alpha}{d})^j.$$

Now, since $\alpha < d$, we get the following by choosing $\ell = \frac{d}{\alpha} \log n$:

$$(1 + \frac{\alpha}{d})^\ell = (1 + \frac{\alpha}{d})^{\frac{d}{\alpha} \log n} > 2^{\log n} = n.$$

Here we go from the second expression to the third using the fact that $(1 + \frac{1}{k})^k$ increases from 2 to e as k ranges from 1 to infinity. So in particular, since $\alpha < d$, the quantity

$$(1 + \frac{\alpha}{d})^{\frac{d}{\alpha}}$$

is greater than 2.

What's the point of this calculation? The point is that the size of S_j can never exceed n, so sometime in the first $\ell = \frac{d}{\alpha} \log n$ steps of the BFS, the inequality

$$|S_{j+1}| \geq (1 + \frac{\alpha}{d})|S_j|$$

must stop holding — and this only happens once S_j contains strictly more than half the nodes.

Let's consider the first $j \leq \frac{d}{\alpha} \log n$ when $|S_j|$ strictly exceeds $n/2$. If t belongs to this set S_j, then we have the short path from s to t that we wanted. If t doesn't belong to this set S_j, then we do the following: we repeat this construction, but starting the BFS outward from t. Again, in at most $\frac{d}{\alpha} \log n$ BFS levels from t, we have set T_i that contains strictly more than half the nodes. Now, S_j and T_i each contain more than half the nodes of the graph, so there must be at least one node that's in both; call this node v. By finding a short s-v path through S_j, and gluing it together with a short t-v path through T_i, we have the desired short path from s to t.