Complex Networks - TD2

Remy Cazabet, Lorenza Pacini

2019

1 networkx

Last time, we have written our own code *from scratch*. But that is not the only way. networkx `https://networkx.github.io` is a very popular graph manipulation library implementing most of the standard tools for graph manipulation.

Start by initiating yourself to the library with the tutorial: `https://networkx.github.io/documentation/stable/tutorial.html`

2 Erdös-Rényi (ER)

Here we calculate some properties of the ER random graph, a staple model in graph theory. $G(n, p)$ graph is defined by n its number of nodes, and a constant probability p of an edge existing between any pair of nodes. As nodes are distinguishable, each graph is drawn from the sets $G_{n,p}$ with probability

$$p^m(1-p)^{C_2^n - m}$$

where C_2^n is the binomial coefficient. We can think of p as a parameter that interpolates between empty graphs ($p = 0$), and complete graphs($p = 1$).

networkx has a function to generate ER graphs. Find it in the documentation and generate your first network.

2.1 Degree distribution

Show, using any tool at your convenience, including plots, that your graph generation methods obey the theoretical degree distribution

$$P(k) = C_k^{n-1} p^k (1-p)^{n-k-1}$$

where $P(k)$ is the probability of a node having degree k.

2.2 The Erdös-Rényi model and connected components

In their 1960 paper, Paul Erdös and Alfred Rényi provided a deep analysis of the class of graphs $G_{n,p}$, studying the characteristics of the graphs as a function of the parameter p. Their results demonstrated that

- if $np < 1$, a graph in $G_{n,p}$ will almost surely have no connected components of size larger than $\mathcal{O}(\log n)$,
- if $np = 1$, a graph in $G_{n,p}$ will almost surely have a largest component whose size is $\mathcal{O}(n^{2/3})$
- if $np > 1$, then a graph in $G_{n,p}$ will almost surely have a unique giant component containing a positive fraction of the vertices. No other component will contain more than $\mathcal{O}(\log n)$ vertices.

In this section, the goal is to carry out your own analysis to convincingly illustrate that your random graph methods obey the above properties. Remember that Erdös and Rényi proposed a probabilistic model, so you are not expected to provide a proof. Rather, design some experiments showing that under the right conditions, the above characteristics are exhibited with high probability.

3 The Watts-Strogatz model

ER graphs do not have two important properties observed in many real-world networks. Firstly,

- they do not generate local clustering and triadic closures. This is because there is a constant, random, and independent probability of two nodes being connected. Secondly,
- they do not account for the formation of hubs. Formally, the degree distribution of ER graphs converges to a Poisson distribution, rather than the power law observed in many real-world, scale-free networks.

The Watts-Strogatz model (1998) was designed to be the simplest possible solution to address the first of the two limitations. It accounts for clustering while retaining the short average path lengths of the ER model. It does so by interpolating between an ER graph and a regular ring lattice. Consequently, the model is able to at least partially replicate the 'small-world' phenomena present in a variety of networks, such as the power grid, the neural network of C. elegans, or networks of movie stars. The algorithm works as follows. Given the number of nodes n, the mean degree k, which is assumed to be an even integer, and a special parameter β satisfying $0 \leq \beta \leq 1$, all such that $n \gg k \gg \log n \gg 1$, the model constructs an undirected graph with $nk/2$ edges in the following way.

- One constructs a regular ring lattice, a graph with n nodes each connected to k neighbours, $k/2$ on each side.
- For every node $n_i = v_0, ..., v_{n-1}$ take every edge (v_i, v_j) with $i < j$, and rewire it with probability β.

Illustrate concisely (plots, tables...) that this model accounts for clustering while retaining the short average path lengths of the ER model. Remember that the networkx library provide many useful functions related to this problem.

4 Barabasi-Albert Model

The Barabasi-Albert, or BA model is an algorithm for generating random scale-free networks using a pref- erential attachment mechanism. The algorithm is as follows.

1. The network begins with a small random, connected network of nodes. Then,

2. new nodes are added to the network one at a time. Each new node is connected to m existing nodes with probability proportional to the degree of each existing node. Formally, the probability that a new node is connected to an existing node u is

$$p_u = \frac{k_u}{\sum k_v}$$

where k_u is the degree of node u, and the sum is made over all pre-existing nodes v, such that the denominator results in twice the current number of edges in the network. Heavily linked nodes, or hubs, tend to quickly accumulate even more links, while nodes with only a few links are unlikely to be chosen as the destination for a new link. As such, new nodes have a higher likelihood of attaching themselves to heavily linked nodes.

Plot the degree distribution $P(k)$ for various size n of networks. Show that the BA model produces scale-free networks.

Is it a small-world network? Why, or why not?

5 Bonus question

Pick your favorite random network generator in networkx, and analyse its properties according to its parameter, following the lines of your great predecessors.

6 Prepare for next TD

The final TD is a project, leading to a grade. Start by choosing, preparing and cleaning your network.