Todo list

Degree distribution in GCC

Benoît Richard Guiyuan Shi

June 21, 2018

Abstract

Write abstract

1 Introduction

Studying the fundamental properties of networks require to be able to abstract from the particular examples found in nature. This is usually done [?] by using a random model for the network generation and averaging the properties of interest over the set of possible networks. One common model is the configuration model [?] that allows to uniformly sample the space of all networks with a given degree distribution [?]. However, many real examples of networks are connected, as for example the World Wide Web or railroad networks, but no model known to us allows to sample the space of all connected networks of a given degree distribution.

A way to still study connected networks is to only consider the Giant Connected Components (GCC) of networks generated using the configuration model [?]. We study here how this method implies bias on the degree distribution of the GCC and propose an algorithm based on this knowledge to generate connected networks of arbitrary degree distribution.

Missing ref

Missing ref

Missing ref

Find other examples

Missing ref

2 Degree distribution in GCC

Per Bayes theorem we have for two random events A and B

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = P(B|A)\frac{P(A)}{P(B)}. \tag{1}$$

{Bayes theorem}

We can apply it to compute the probability r_k that a vertex in the GCC has degree k

$$\begin{aligned} r_k &= P(deg(v) = k | v \in GCC) && (2)\\ &= P(v \in GCC | deg(v) = k)\frac{P(deg(v) = k)}{P(v \in GCC)} && (3)\\ &= (1 - P(v \notin GCC | deg(v) = k))\frac{p_k}{S} && (4)\\ &= \frac{p_k}{S}(1 - u^k)., && (5) \end{aligned}$$

{Degree distribution in GCC}

where S is the probability that a random node is part of the GCC, p_k is the probability that a node has degree k and u is the probability that a node reached by following an edge of the network is not part of the GCC.

More details for the last step ? Maybe a ref ?

Plot showing difference between degree distribution of the whole network and the GCC for various distributions.

In the context of the configuration model we choose the probabilities p_k, which determine u and S as

$$u = \frac{\sum_{k=1}^{\infty} k p_k u^{k-1}}{\sum_{k=1}^{\infty} k p_k} \tag{6}$$

$$S = 1 - \frac{\sum_{k=1}^{\infty} p_k u^k}{\sum_{k=1}^{\infty} p_k}, \tag{7}$$

thus eliminating all unknown in eq. (5).

3 Generating connected networks

The knowledge of the degree distribution in the GCC can be used generate a connected component of a given degree distribution r_k: first we determine a degree distribution p_k fulfilling eq. (5). Then we generate a network with degree distribution p_k using the configuration model. Finally we take its GCC as our connected network. By virtue of eq. (5) it has degree distribution r_k. The main problem here is to determine the p_k. This is hard since u is a function of all p_k, making eq. (5) complicated to solve. We now propose an algorithm to determine it numerically.

We first define $\tilde{p}_k = p_k/S$, giving the equations

$$\tilde{p}_k = \frac{r_k}{1 - u^k} \tag{8}$$

$$u = \frac{\sum_{k=1}^{\infty} k \tilde{p}_k u^{k-1}}{\sum_{k=1}^{\infty} k \tilde{p}_k} \tag{9}$$

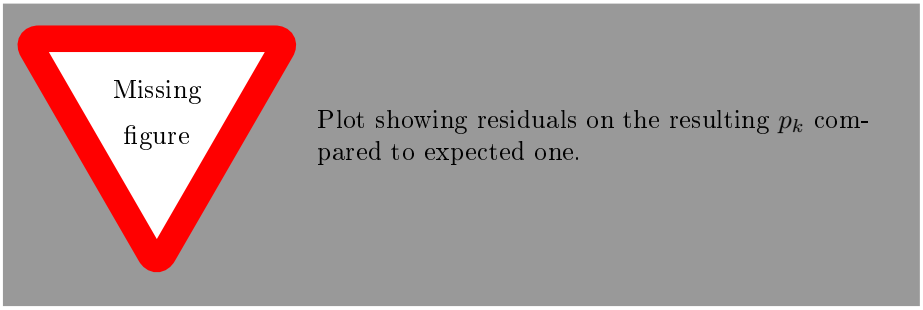
$$p_k = \tilde{p}_k / \sum_{k=1}^{\infty} \tilde{p}_k. \tag{10}$$

Note that the u and $\tilde{p}_k$ solving this equations can be considered as being fixpoints. Iteratively inserting a guess into it should therefore make it converge towards a solution. Also the case $u = 1$ implies that no GCC is present and therefore does not need to be considered here.

Not true in general, prove that claim ?

We can not deal numerically with infinite sum, thus we choose a cutoff index K. We also choose a tolerance δ and a starting value u_0 for u. The algorithm then goes as follow

1. Set $\tilde{p}_k := r_k$, $\forall k$ and $u_{ref} := u_0$, $u := u_0$.
2. Compute $u_{new} := \sum_{k=1}^{K} k \tilde{p}_k u^{k-1} / \sum_{k=1}^{K} k \tilde{p}_k$.
3. If $|u_{new} - u| > \delta$ set $u := u_{new}$ goes to 2, otherwise set $u := u_{new}$ and continue.



Plot showing residuals on the resulting p_k compared to expected one.

4. Set $\tilde{p}_k := r_k/(1-u^k)$, $\forall k$.
5. If $|u_{ref} - u| > \delta$ set $u_{ref} = u$ and go back to 2, otherwise continue.
6. Compute $p_k := \tilde{p}_k / \sum_{k=1}^{\infty} \tilde{p}_k$ for all k and return them.

This gives us the first K probabilities p_k, which is sufficient to sample random numbers between 1 and K with probability p_k. If K is chosen such that $r_k << 1$ for $k > K$, the degree distribution in the GCC closely approximate the distribution r_k.

Would be interesting to have a use case where the difference in distribution has a meaningful impact

This may not be possible for scale-free network due to fat tail, check ?

4 Discussion