Network Science

Bruno Ramos Dias 86392
João Daniel Silva 86445

October 2019

1 Introduction

We were proposed to study and apply computational analysis to complex networks. With the freedom we were given to choose which subject to work on, and with our background in computer science, we decided to program our own Python library. We implemented our own data structure and some of the principal metrics discussed in the course. We also implemented two models of networks: *Erdős–Rényi* and *Barabási–Albert* models. Finally, we decided to do this report in LaTeX to practice for upcoming works.

2 Our Python Library

2.1 Data structure

We have created a Graph class in which we represent the network via adjacency list using a python dictionary. The keys of the dictionary are the nodes of the network and their respective values represent all the nodes that are connected to. Our Graph supports both direct and undirected network. Represented degree of each node is also stored in another python dictionary in which the keys are the nodes itself and the corresponding values are the corresponding degrees. At last, we also store the path lenghts from all nodes to other nodes of the network in another python dictionary in which the key values are the minimum path length between nodes and their corresponding values are the number of minimal paths between nodes that have that length.

2.2 Analysis Auxiliary Methods

In order to help us both check whether the networks that we have created are according to their respective models, or to check the results of some metrics of our library, we have created two methods, one that loads a network from a file that contains a list of edges, and other that writes our network into a file with the same shape as described. So, recurring to these methods we can create a network using our class models that we will explain later in this report, save it in a file and check in Gephi whether our library produces all the expected values or not. Plus, if we want to load any data set into our library we can do it as long as it is presented in the format that was previously described.

2.3 Calculating metrics

We implemented the principal metrics to study networks: average degree, degree distribution, clustering coefficient, average path length and distance. Some of these calculations might not be optimal. To measure the path length from a node to another we use BFS, and since we use adjacency list to represent the graph the complexity is $O(N + E)$. So to calculate the average path length we have to do this for every node making it $O(N^2)$. Also, in the calculation of the clustering coefficient, when we count the edges among neighbors, the complexity is $O(N^2)$ for each node.

Some optimizations suggested by the professor that we followed are to calculate only each pair of nodes once when calculating the shortest path between two nodes and to count the edges between neighboors (for the clustering coeefficient).

2.4 Network Models

To represent networks we have implemented both Erdos and Barabási python classes that can build the respective models and also have methods that can confirm some simple metrics demonstrations related to the network.

2.4.1 Erdős–Rényi Model

A random network consists of N nodes where each pair of nodes is connected with probability p. Our class Erdos builds a graph like this. We have methods to check if some properties we discussed in class hold true with our model, such as average number of edges, average degree and if the maximum distance is $\frac{lnN}{ln<k>}$.

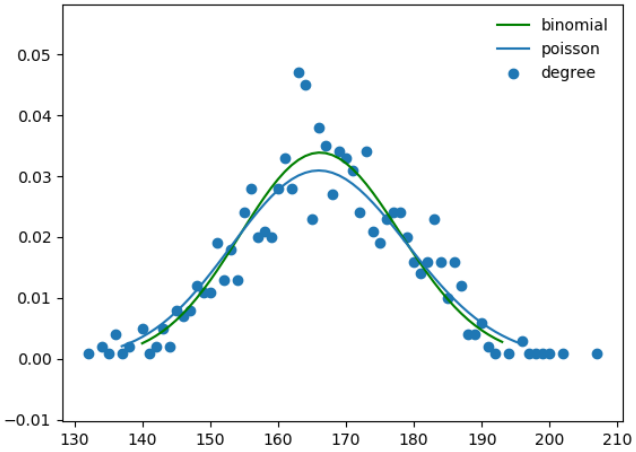
We also plot the degree distribution to try to see if it matches a binomial/poisson distribution.

2.4.2 Barabási–Albert Model

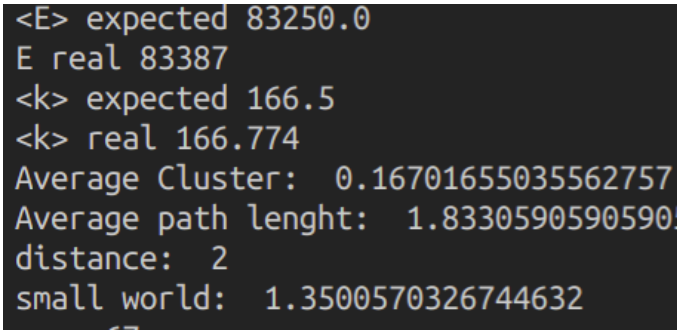
We use the following arguments to build the model: *InitNodes* that is the number of nodes that our starting network has, m that is the number of edges that every new node is going to have (initial degree of the node) and the number of nodes that we want to add to the network. We start by calling the *Erdos* Model constructor where p is 1 and number of nodes equal to InitNodes, so we have an initial ring. Then we have a initial network in which we add each node sequentially, adding m edges to this node in which we choose the nodes that we want to link with from a probability list in which entry is computed following the mathematical formula $\frac{k_i}{\sum k_j}$. This way, nodes with higher degree have an higher chance of linking compared to those with lower degrees.

3 Data Analysis

3.1 Erdős–Rényi Model



(a) Degree distribution



(b) Metrics

Figure 1: Erdos model N = 1000 and p = 1/6

With limited computational resources, we tried some simulations. In this model most of the values were according to the expectations. We observed the low clustering characteristic of this model, being one reason it may be inappropriate to represent real networks.

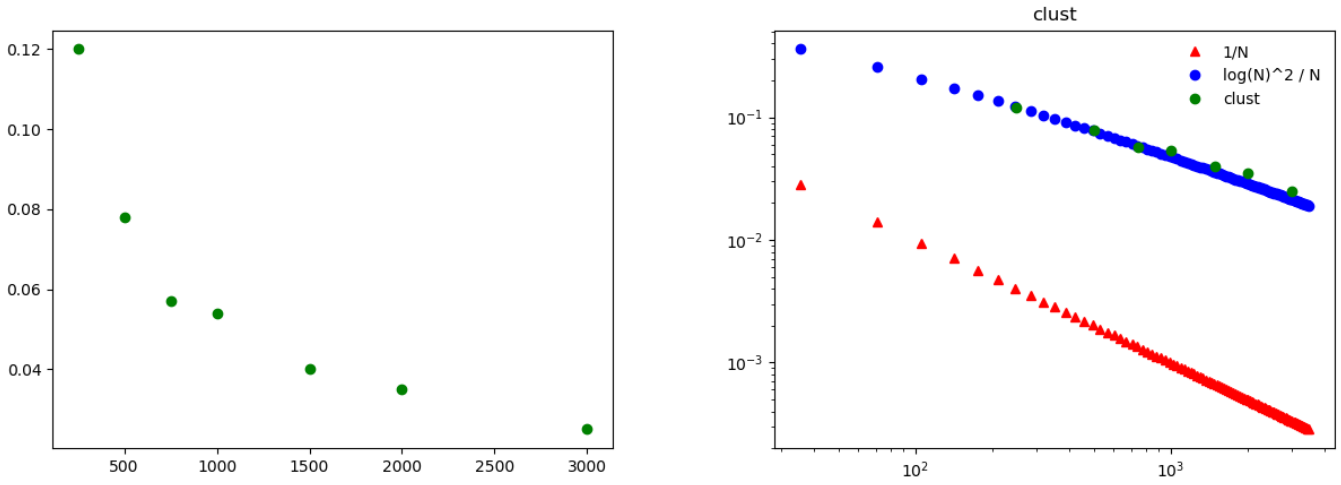
The real distance of the network has some deviation form the calculations, yet we can observe the small world effect. The degree distribution graph tends to follow more the binomial distribution than the poisson, due to our inability to simulate with N >1000.

3.2 Barabási–Albert Model

In the following chart we describe BA model simulations with the number of nodes taking values between 260 and 3010 nodes, that starts its building from an Erdos model with 10 nodes fully connected, and with the configuration that every new node makes 5 connections once its added.

Every pair of figures represents the metric that we want to analyze, being the first picture the metric result for each number of nodes computed and the second picture the comparison between the metric (colored in green), its expected value (colored in blue), and also the expected value if we were dealing with an Erdos model(colored in red).

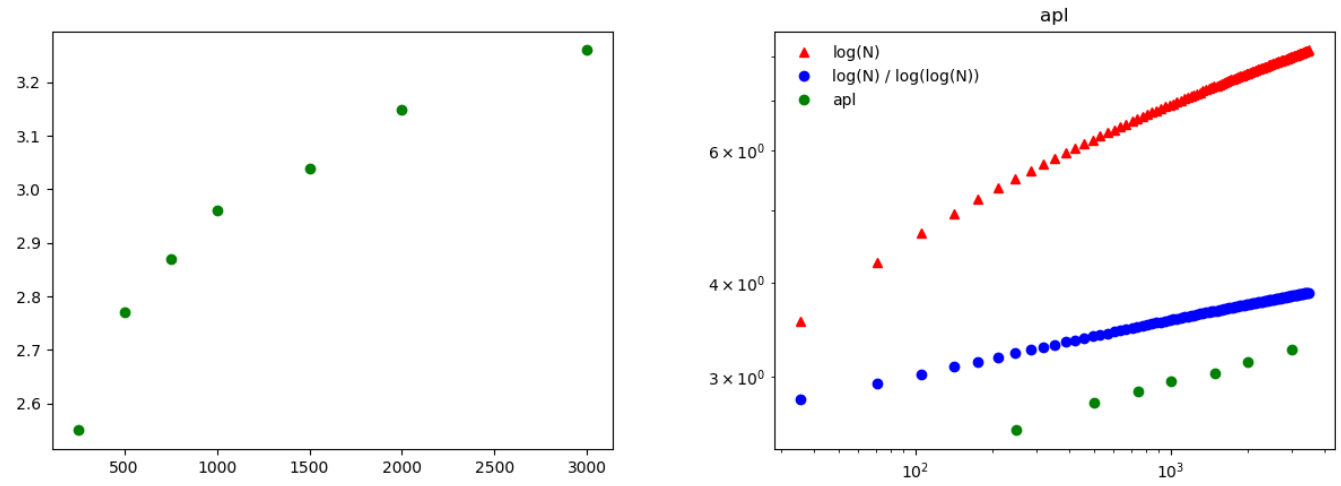
This way, we can check that the results are according/tending to their respective expected values.



(a) Clustering coefficient

(b) Comparison

Figure 2: Clustering Coefficient simulation



(a) Average Path Length

(b) Comparison

Figure 3: Average Path Length simulation

4 References

https://en.wikipedia.org/wiki/Erd%C5%91s%E2%80%93R%C3%A9nyi_model

https://fenix.tecnico.ulisboa.pt/disciplinas/CRC7/2019-2020/1-semestre/slides-reading-list-and-supplemental-material-7c7