

Homework 10 - Network Science

Gabriel Luciano Gomes (265673)

December 11, 2020

1 Network Robustness

For this assignment, we're going to analyse network robustness against random failures. This metric can be observed by measuring the relative size of the giant component after a random remove of f fraction of nodes. The upcoming code computes the random failures of a given network.

Listing 1: Analysing network robustness

```
import networkx as nx
from numpy import arange
import random

def randomFailures(G, increments):
    """
    Computes and plots the network robustness against random failures.
    To do so, calculate the relative size of the giant component after
    an  $f$  fraction of routes randomly removed.
    :param G: Graph to be analysed
    :param increments:  $f$  fraction of nodes to be removed
    :type G: Networkx graph
    :type increments: int
    """
    graph_copy = nx.Graph(G)
    n_nodes = graph_copy.number_of_nodes()
    remove_percent = int(increments * graph_copy.number_of_nodes())
    p_inf = []

    p_inf.append(
        len(max(nx.connected_components(graph_copy), key=len)) / n_nodes
    )

    for i, f in enumerate(arange(0.05, 1.0, 0.05)):
        current_nodes = list(graph_copy.nodes())
        nodes_to_remove = random.sample(current_nodes, remove_percent)
        graph_copy.remove_nodes_from(nodes_to_remove)
        p_inf.append(
            len(max(nx.connected_components(graph_copy), key=len))
            /
            (n_nodes - ((i+1)*remove_percent)) # Normalization [0, 1]
        )
```

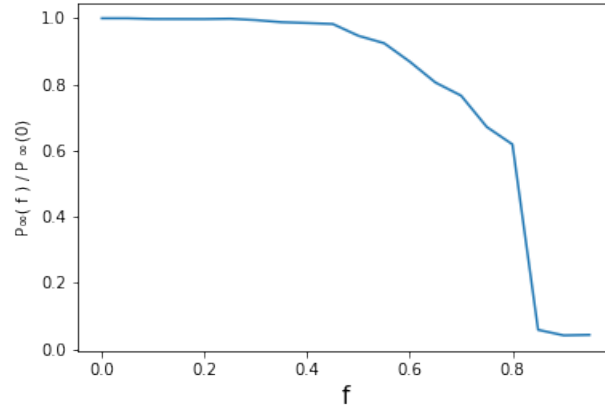


Figure 1: Network Robustness after random removals

We can observe that, after a random removal, this network is **robust**. This can be concluded analysing that the graph do not fall apart after the f removal. Just at the end of the plot we can observe that it breaks; however, almost all nodes were removed.