Giacomo Garbarino S4545532 Manuel Parmiggiani S4701853

Network Analysis 1st assignment

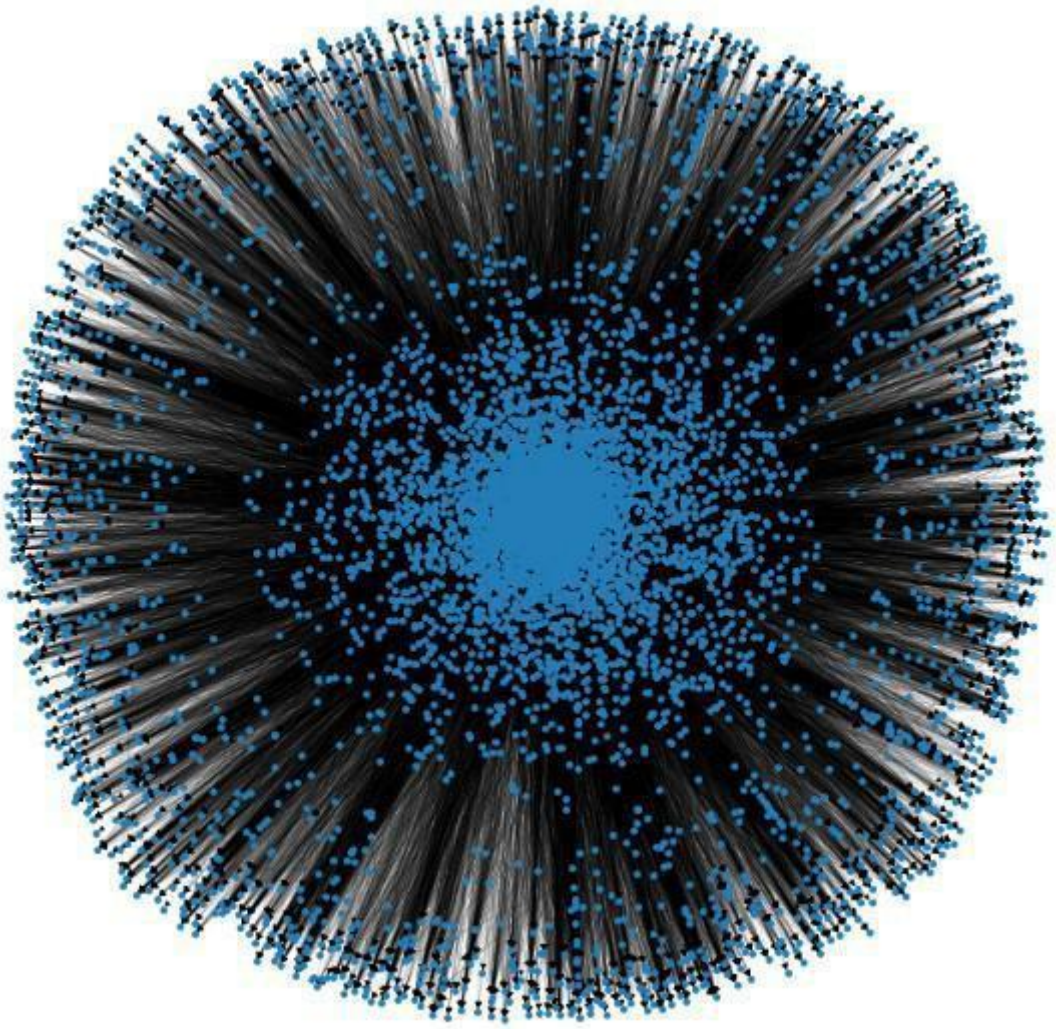
computing different measures on a realistic graph

Introduction

The aim of this assignment is taking a realistic graph expressing real-world data and compute different metrics over it, in order to infer its properties.
The graph is the following directed network: https://snap.stanford.edu/data/wiki-Vote.html, a wiki-vote dataset containing Wikipedia's voting results.
For the implementation, we decided to use the NetworkX Python library for the calculation part while we relied on NumPy and matplotlib for plotting the results in our last analysis about degree distribution.

Metric's calculation

First of all, we made up a visual representation of our data, this is what we obtained through the use of NetworkX data loading functions:



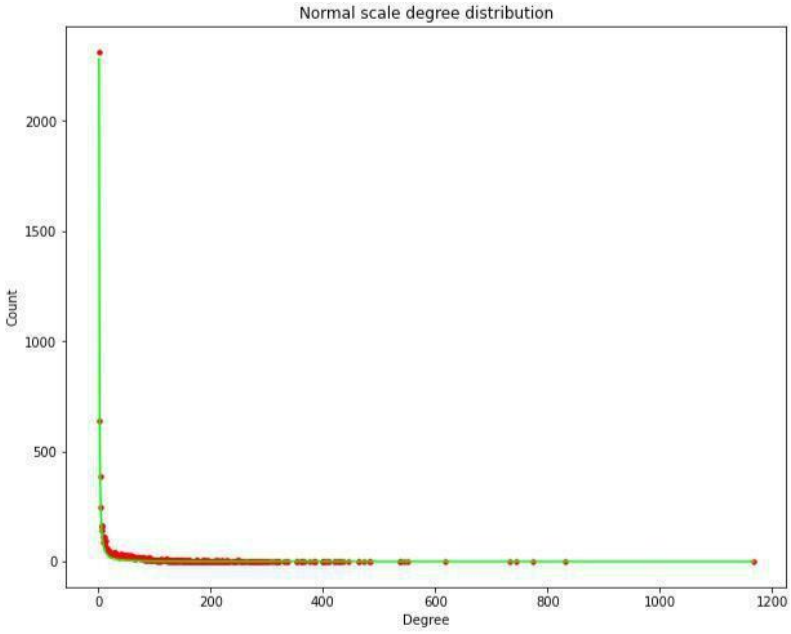
Even before calculating metrics we can identify the presence of a giant component by looking at this graph; thanks to NetworkX, we are easily able to demonstrate that the graph is not strongly connected (so the giant component doesn't correspond to the entire graph). We now proceed to calculate the main metrics on the graph and on the giant component:

Metric	Value on Graph	Value on Giant Component
Size	7115	1300
Number of edges	103689	39456
Diameter	***	9
Average degree	14.573295853829936	30.35076923076923
Density	0.0020485375110809584	0.023364718422455143
Global clustering	0.05285122817867548	0.08512174620084431
Average clustering	0.08156344522820935	0.18221830113253618
Average shortest path length	***	2.8792828803221413
Assortativity	-0.08324455771686863	-0.06306021987134619

*****not calculated since the graph is not strongly connected**

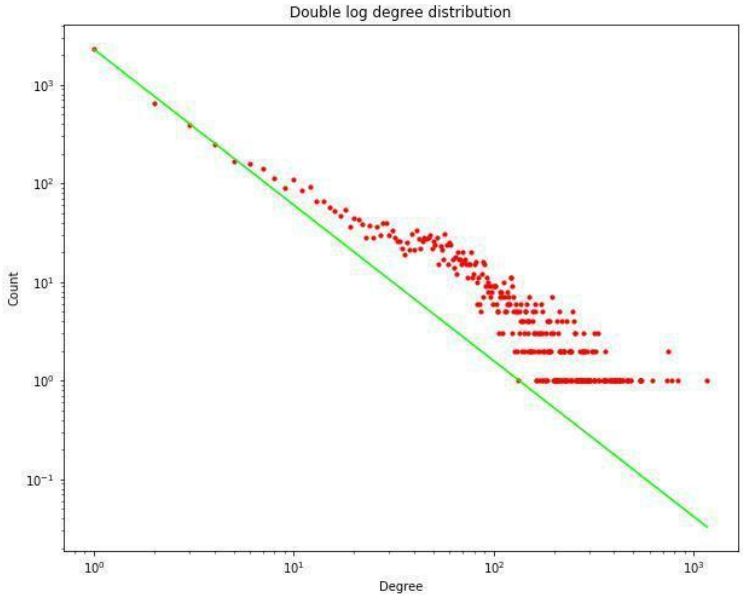
Now we will proceed to conduct our analysis by answering some relevant questions:

1) Does the graph have the same characteristics of a random or a power-law network?



In order to answer this question, we have to observe degree distribution among the nodes in the graph trying to infer some general property of it. From this plot, we can see how degree distribution in our data is an inversely proportional relation. It's clearly noticeable that, the higher the node degree is, the lower is the number of nodes having that degree, so we can state that our graph is very similar to a power law network, where the most of the nodes are lowly linked.

We also plotted the degree distribution in double logarithmic scale to check and confirm what we've inferred from the previous plot: our network is very similar to a power law network, but it doesn't fit a pure power law distribution at all and real networks rarely do it.



2) Which are the most important nodes, with respect to a given centrality measure?

We now proceed to calculate some centrality measures in order to find the "most important" nodes. The adopted measures are degree, closeness and betweenness. The betweenness measures the number of shortest paths which pass through a node. The removal of nodes with high betwenness may disrupt communications since they may have important informations. The closeness of a node measures the mean distance of a node to other nodes. Nodes with high closeness have more direct influence on other vertices or better information access.
For each measure, we calculate the top 10 nodes for that metric.

Degree

Node	2565	1549	766	11	1166	457	2688	1374	1151	5524
Value	1167	832	773	743	743	732	618	551	543	538

Betweenness

Node	2565	1549	15	72	737	1166	5079	2328	2237	28
Value	0.0177	0.0166	0.0116	0.0080	0.0061	0.0058	0.0054	0.0052	0.0047	0.0046

Closeness

Node	4037	15	2398	1549	2535	3089	762	5412	2565	5254
Value	0.2965	0.2915	0.2909	0.2819	0.2799	0.2780	0.2780	0.2778	0.2776	0.2765

Calculating centrality using different centrality notions is useful because we can see how some nodes (like 2565 and 1549) are in the top 10 in all the three cases or in almost two cases (for example node 15 is in the top 10 nodes for betweenness and closeness but not for degree), so we can consider nodes that are in the top 10 in more than one case as the most relevant ones and this is almost confirmed by the fact that nodes like 2565 and 1549 are the best ones for 2 metrics out of 3.

3)Are the paths short with respect to the size of the network?

The average shortest path is the average of the shortest paths between all pair of nodes. Since the network is not strongly connected, we cannot calculate the average shortest path lenght for all the nodes, but we can do it for the giant component.
Average shortest path lenght in the giant component is almost 2.88 that's very short compared to the 1300 nodes the giant component has, so we can detect there's a small-world effect here.
We can have a further confirm of it looking at the diameter (expressing the longest path among two nodes): as the previous metric, we can say this value is very short compared to the total nodes of the giant component.

4) Is the network dense?

The density of network measures the ratio between 0 and 1 of the number of edges to the number of all possible edges in the network.
The network is not dense, the density value is 0.002 which means the network is more likely a sparse network.
The situation changes a bit if we look at the giant component only; as we can see, the giant component has almost 20% of the nodes of the entire graph but almost 40% of its edges and this explains why the giant component has a bigger average degree and density. This affects also clustering metrics; we can observe that global and average clustering results have higher values in the giant component than in the full graph.

5) Is the network assortative?

The assortativity measures the tendency of the network's nodes to attach to other nodes with similar degree. The calculated assortativity of our network is negative, this means that large-degree nodes tend to attach to low-degree nodes, so the network is disassortative since similar degree nodes don't tend to be directly linked.