Michał Bujak

Network Science

Kraków 2021

Contents

1 Abstract

The project aims to consider two graphs, both corresponding to the social networks. One represents the virtual contacts of some individuals, while the other presents virtual contacts in a possibly different group. Social networks by their nature are multilayered. Their consist of neighbours, friends, colleagues at work. We will explore the two graphs, analyse them and try to find the optimal strategy to spread the information.

2 Data selection

The data has been downloaded from [10]. Attempts to perform analysis on the bigger datasets were impeded by the computational constraints. As a result, the analysis was performed on relatively small datasets.
First one is *dolphins* which is described as *a social network of bottlenose dolphins. The dataset contains a list of all of links, where a link represents frequent associations between dolphins.*
The second one is *karate*, *the dataset contains social ties among the members of a university karate club collected by Wayne Zachary in 1977.*

3 Data overview

As suggested in [11], when starting the work on the dataset, it is a good idea to get a general sense of data. The first step is to simply open the files and see what is inside. In this case, we analyse the undirected, not weighted graphs. Fact that the graphs are undirected is strictly in line with our case assumptions. We try to replicate possible connections between people which results in either disease or information spread. In both cases, the transition may take either direction.
One may argue that the graph should be weighted. Spreading disease depends on the precautions that one takes such as keeping the adequate distance, hygiene but also on the level of the relationship. Similarly with spreading the information. However, we assume that such additional information is not provided and we base our analysis on the simplified case when the contact between any two people is assumed the same.
Karate and dolphins are graphs a similar magnitude. Strictly speaking, for *dolphins*, we have the following:

```
Number of nodes: 62
Number of edges: 159,
```

and for *karate*:

```
Number of nodes: 34
Number of edges: 78.
```

4 Statistics of the datasets

The two important statistics of the network we consider in the first place are average node degree and density. Following the definitions in [1], the *degree* of a node represents

the number of links that it has to other nodes. In an undirected graph, *total number of links* L can be expressed as the sum of the node degrees:

$$L = \frac{1}{2}\sum_{i=1}^{N} k_i,$$

where k_i represents degree if i-th node in the N-node graph. *Average degree* $< k >$ is expressed as the mean of degrees of nodes can be expressed as

$$< k >= \frac{2L}{N}.$$

For the two aforementioned datasets we obtain the following results:

	Dolphins	Karate
Average degree	5.1290	4.5882

It is important to point out that both both network exceed so-called *connected regime*, which [1] defines as $ln(N)$ ($ln(159) = 5.0689$ and $ln(78) = 4.3567$). Exceeding the connected regime threshold usually results in obtaining the connected graph, which is indeed the case for both networks. The *degree distribution* provides the probability that a randomly selected node in the network has degree k. We observe that although both

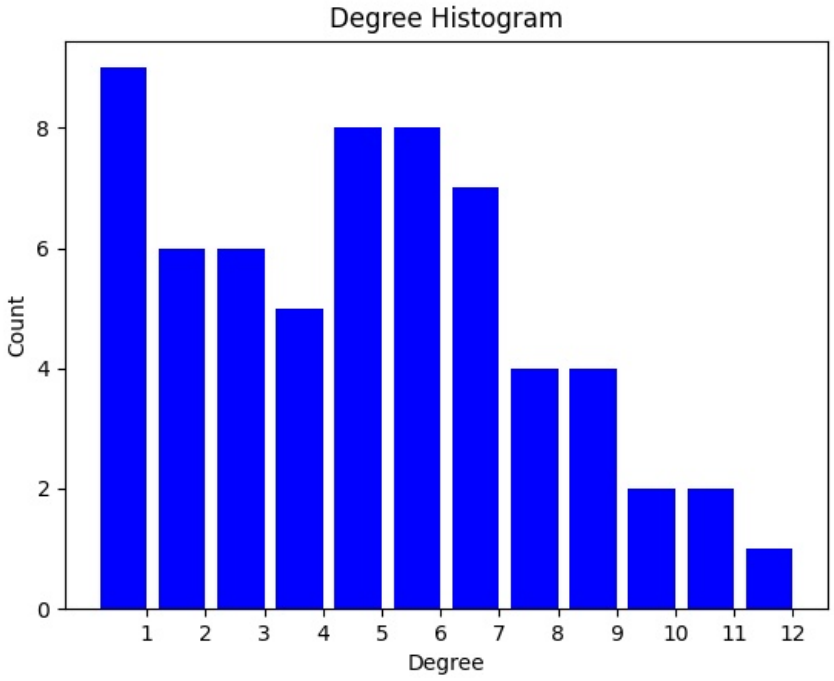

Figure 1: Distribution of nodes' degrees in dolphins dataset

networks have a similar average degree, they present different distributions. Dolphins have a right-skewed distribution with a more continuous-like plot of the pdf. On the other hand, karate has a much lower median of degrees with a few separated greater values. To assess the data more in detail, we present a top 5 nodes in terms of degrees (the pair corresponds to a node and its degree respectively).

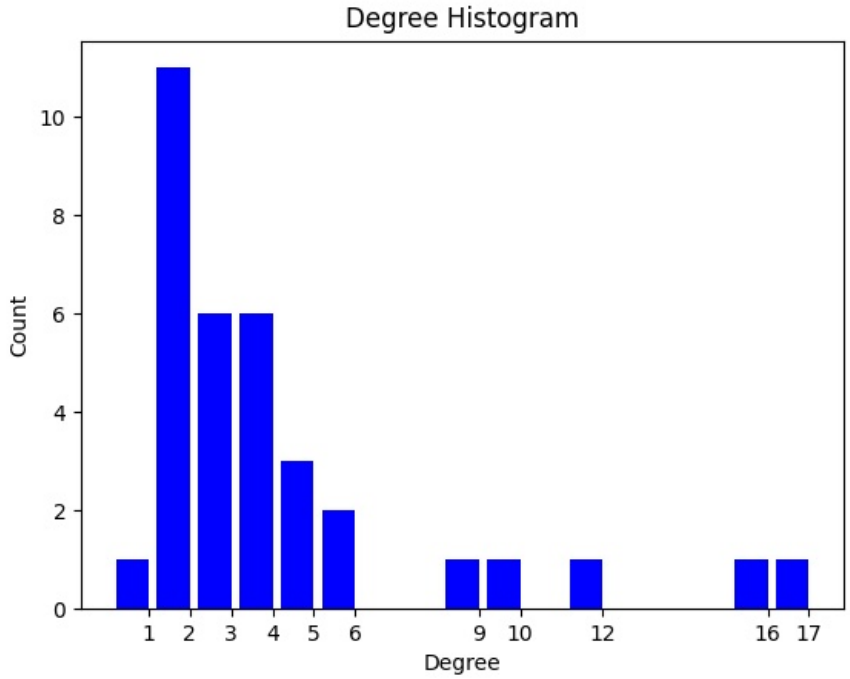

Figure 2: Distribution of nodes' degrees in karate dataset

```
Dolphins:
(15, 12)
(38, 11)
(46, 11)
(52, 10)
(34, 10)

Karate:
(33, 17)
(0, 16)
(32, 12)
(2, 10)
(1, 9)
```

Another similar network statistic is *density*. Citing [3], network density describes the portion of the potential connections in a network that are actual connections. A *potential connection* is a connection that could potentially exist between two "nodes" – regardless of whether or not it actually does. This person could know that person; this computer could connect to that one. Whether or not they do connect is irrelevant when you're talking about a potential connection. By contrast, an "actual connection" is one that actually exists. This person does know that person; this computer is connected to that one.

	Dolphins	Karate
Density	0.084	0.139

We can see that although dolphins have a higher average node degree, the network has a smaller density (due to higher number of nodes).

The *global clustering coefficient*[1] is based on triplets of nodes. A triplet is three nodes that are connected by either two (open triplet) or three (closed triplet) undirected ties. A triangle graph therefore includes three closed triplets, one centered on each of the nodes (n.b. this means the three triplets in a triangle come from overlapping selections of nodes). The global clustering coefficient is the number of closed triplets (or 3 x triangles) over the total number of triplets (both open and closed). For unweighted graphs, the clustering c_u of a node u is the fraction of possible triangles through that node that exist[2],

$$c_u = \frac{2T(u)}{deg(u)(deg(u)-1)},$$

where $T(u)$ is the number of traingles through node u and $deg(u)$ is the degree of u.

	Dolphins	Karate
Global clustering coefficient	0.309	0.256
Average clustering coefficient	0.259	0.571

The average clustering coefficient for karate is higher which is in line with higher density.

5 Spreading the information

In the first attempt, we will try to spread information in the karate network from a single source. The radius of the network, so the length of path connecting the central point and the most distant, is equal to 3. Let us assume that in one time unit (for simplicity call it days) the information moves to the adjacent nodes. In our case, it means that for the information to reach all the people, it needs at least 3 days. However, for our model, we should also consider how quickly the information spreads. For example, we can find a so-called central point, so to reach all nodes in 3 days, but after 1 day only 10% of the nodes would be reached. On the other hand, we could start at a point such that it needs 4 days to spread information to all people, but after one day it already reaches half of the network.
To measure the efficiency of information spread, we propose the following formula:

$$E(s) = \frac{1}{\sqrt{\#\{s\} * K^2}} \sum_{n=1}^{K} r_i, \tag{1}$$

where s is a set of starting points, K is a number of days for the information to reach all nodes, r_i is a cumulative part of the network's nodes that is reached after i days. The coefficient K will be called the *spread value*.

5.1 Random starting point

To begin with, we start by choosing twice randomly picked starting point. The information reached half of the nodes after 2 days in both cases and all nodes in 5 days. Detailed characteristics for all runs are presented in the Appendix.
An attempt to use a more rigoristic approach is to start at the node with the highest

[1] Cited from [12], confirmed with [8].
[2] Referring to [8].

degree. The information reached the most distant nodes within 4 days. Also, it gives better coverage in the intermediate steps. As expected, the spread value is bigger.

5.2 Central points

Another approach bases central points. Having a list of them, we can start to see the dynamics of information spread. Surely, the information will reach nodes in no more than the radius of the network is. Comparing to the previous approach, we got both worse and better results. The highest spread value was obtained when starting from the node [0]. The score is not surprising as it was the node with the second highest degree (16 comparing to 17 for the [33] node).

5.3 Community

A *community* is defined as a subset of nodes within the graph such that connections between the nodes are denser than connections with the rest of the network([9]). For purpose of finding communities within the network, we can use the Girvan-Newman Algorithm (described in detail in Chapter 9 in [1]).
Using the algorithm, we split the network into two communities. Here we try the two approaches. First, calculate degrees for nodes in the whole network and pick the nodes with the highest degree from each community. Second, calculate degrees for nodes in subgraphs forming communities and pick the nodes with the highest degrees. It turns out, that the first strategy is much better, resulting in spreading information in 2 days and reaching over 90% of nodes after the first day. The obtained spread value is greater than for a single starting point.

5.4 Dominating set

Dominating Set, S, is defined as a subset of V such that each node in $V \backslash S$ is adjacent to at least one node in S ([2]). Naturally, starting from the nodes forming dominating set of the network, the information will reach all nodes in one day. In this example, the dominating set consists of 13 elements. Interestingly, following steps from the previous subsection and using Girvan-Newman so to get 13 communities also resulted in finding the dominating set, however different one. Another thing to point out is that the 13 nodes, given as the dominating set by the Networkx algorithm, are separated, which means that none of them is connected to the other. The run based on the dominating set yielded a lower spreading value than both one and two points run, because of the penalty set on the number of the starting points.

5.5 Optimal strategy

Based on the previous runs we know that sometimes it is more beneficial to choose starting points based on centrality rather than degree. Also, we know that an increasing number of starting points results in a bigger penalty. What we propose as the optimal strategy is to run a loop on the decreasing in terms of size communities. From each community choose the central point with the highest degree. Once an increase in the number of groups results in a deterioration of at least 0.01 spread value, we break the loop. The

break is optional, created for more complex networks, where the limited efficiency of the Girvan-Newman algorithm, searching for the central points and the dominating set could impede the process. The starting points obtained by the *optimal strategy* are the same captured by the highest degrees for the two communities.

6 Applying methodology on the second dataset

Many networks have similarities. The two datasets that are analysed are of the closes average node degree, density and global clustering coefficient. Those similarities may lead us to the belief that similar patterns should govern both. Indeed, repeating the same approaches to the problem, we got strongly resembling results. Again, the results obtained by the *optimal strategy* are the best. Interestingly, once more splitting the network into two communities proved to be the most efficient solution. The results, including the highest-degree single node, dominating set and the *optimal strategy*, are presented in the Appendix.

7 Questions

Is this problem temporal/dynamic or static?
Citing [1], *in real networks the number of links rarely stays fixed*. In particular, when we consider social networks it is a very dynamic network. People get to know others with time, some relationships expire. Also, the number of nodes in the network changes - people travel, bear, die. However, we may assume the established network will be a good approximation for some period.

Is it deterministic or probabilistic?
The problem as defined in the Abstract is fully deterministic. The outcome can be precisely computed. However, in the real-world epidemic models, one usually assumes a fixed probability of infection. Citing [5], *when we write down deterministic equations, we are implicitly assuming that the actual proportion in each state is closely approximated by the expected number in each state*. [4] presents and refers to deterministic models.

What is the upper bound and median of spreading - how to approach it?
One way to approach the problem is through the SIR epidemic model. The classical SIR[3]. model can be expressed by the following nonlinear differential equations

$$\begin{aligned} \frac{dS}{dt} &= -\beta SI, \\ \frac{dI}{dt} &= \beta SI - \gamma I, \\ \frac{dR}{dt} &= \gamma I, \\ S + I + R &= N. \end{aligned}$$

where S, I, and R represent susceptible, infected, and recovered compartments, respectively, in the whole population. S represents people who have not been infected yet but

[3]Cited from [6].

can be infected in future. I represents infected people who can spread the epidemic to susceptible people through physical contact. R denotes people who have recovered or died from the epidemic and who no longer participate in the epidemic spreading process. The sum of the S, I, and R values represents the whole population size N. Epidemics have two parameters in the SIR model, transmission rate (β) and recovery rate (γ). As discussed in [7], another approach to the problem of spreading speed is through numerical simulations, which yield useful results on small scales but, for increasingly large complex networks, may prove slow and impractical. Alternative approximations have been made by considering only the most probable path between a given target node and the source. It is known that this shortest-path approach can significantly overestimate the infection arrival times. [7] presents exact mathematical formulas which allow approximating the speed of spreading disease. The tree-like assumption is said to be the fastest spreading regime: the mid-outbreak phase of exponential growth. However, it can be shown that, in a network of size N, the time needed for an infection to take hold grows like $\mathcal{O}(\frac{logN}{\lambda-1})$, where λ represents the unique maximum eigenvalue of the nonbacktracking matrix.

8 Appendix

In the Appendix, there are presented results of the aforementioned runs. Until stated else, all the results below correspond to the karate dataset.

8.1 Start at one random point

Columns represent the time step, number of nodes reached, fraction of nodes reached and if the fraction reached the threshold set to 0.5.

```
Starting points:  [26]
----------------
0 1 0.029411764705882353 False
1 3 0.08823529411764706 False
2 18 0.5294117647058824 True
3 24 0.7058823529411765 True
4 33 0.9705882352941176 True
5 34 1.0 True
Spread coefficient: 0.09150326797385622
------- Random node -------
----------------
Starting points:  [14]
----------------
0 1 0.029411764705882353 False
1 3 0.08823529411764706 False
2 19 0.5588235294117647 True
3 25 0.7352941176470589 True
4 33 0.9705882352941176 True
5 34 1.0 True
Spread coefficient: 0.09313725490196079
```

8.2 Start at the point with the highest degree

Top 5 nodes in terms of a degree.

```
Nodes degrees:  [(33, 17), (0, 16), (32, 12), (2, 10), (1, 9)]
```

Run on the node with the highest degree.

```
Starting points:  [33]
----------------
0 1 0.029411764705882353 False
1 18 0.5294117647058824 True
2 24 0.7058823529411765 True
3 33 0.9705882352941176 True
4 34 1.0 True
Spread coefficient: 0.12823529411764706
```

8.3 Start at the central points

```
Starting points:  [0]
---------------
0 1 0.029411764705882353 False
1 17 0.5 True
2 26 0.7647058823529411 True
3 34 1.0 True
Spread coefficient: 0.14154411764705882
---------------
Starting points:  [1]
---------------
0 1 0.029411764705882353 False
1 10 0.29411764705882354 False
2 23 0.6764705882352942 True
3 34 1.0 True
Spread coefficient: 0.12316176470588236
---------------
Starting points:  [2]
---------------
0 1 0.029411764705882353 False
1 11 0.3235294117647059 False
2 31 0.9117647058823529 True
3 34 1.0 True
Spread coefficient: 0.13970588235294118
---------------
Starting points:  [3]
---------------
0 1 0.029411764705882353 False
1 7 0.20588235294117646 False
2 23 0.6764705882352942 True
3 34 1.0 True
Spread coefficient: 0.11764705882352941
---------------
Starting points:  [8]
---------------
0 1 0.029411764705882353 False
1 6 0.17647058823529413 False
2 31 0.9117647058823529 True
3 34 1.0 True
Spread coefficient: 0.13051470588235292
---------------
Starting points:  [13]
---------------
0 1 0.029411764705882353 False
1 6 0.17647058823529413 False
2 31 0.9117647058823529 True
3 34 1.0 True
```

```
Spread coefficient: 0.13051470588235292
---------------
Starting points:  [19]
---------------
0 1 0.029411764705882353 False
1 4 0.11764705882352941 False
2 31 0.9117647058823529 True
3 34 1.0 True
Spread coefficient: 0.12683823529411764
---------------
Starting points:  [31]
---------------
0 1 0.029411764705882353 False
1 7 0.20588235294117646 False
2 33 0.9705882352941176 True
3 34 1.0 True
Spread coefficient: 0.1360294117647059
```

8.4 2 communities

The two communities within the network:

```
[0, 1, 3, 4, 5, 6, 7, 10, 11, 12, 13, 16, 17, 19, 21],
[2, 8, 9, 14, 15, 18, 20, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33]
```

Results, when we picked nodes with the highest degree in whole network.

```
Starting points:  [0, 33]
---------------
0 2 0.058823529411764705 False
1 31 0.9117647058823529 True
2 34 1.0 True
Spread coefficient: 0.15020242084027968
```

For the highest degree inside subgraph (community).

```
Starting points:  [13, 15]
---------------
0 2 0.058823529411764705 False
1 8 0.23529411764705882 False
2 31 0.9117647058823529 True
3 34 1.0 True
Spread coefficient: 0.09488749085775361
```

8.5 Dominating set

Results, when we picked nodes with the highest degree in whole network.

```
Dominating set:  {0, 9, 14, 15, 16, 18, 20, 22, 23, 24, 26, 28, 30}
---------------
0 13 0.38235294117647056 False
1 34 1.0 True
Spread coefficient: 0.09488749085775361
```

8.6 Optimal strategy

Results, when we picked nodes with the highest degree in whole network.

```
Starting points:  [0]
----------------
0 1 0.029411764705882353 False
1 17 0.5 True
2 26 0.7647058823529411 True
3 34 1.0 True
Spread coefficient: 0.14154411764705882
----------------
Starting points:  [0, 33]
----------------
0 2 0.058823529411764705 False
1 31 0.9117647058823529 True
2 34 1.0 True
Spread coefficient: 0.15020242084027968
----------------
Starting points:  [0, 33, 9]
----------------
0 3 0.08823529411764706 False
1 31 0.9117647058823529 True
2 34 1.0 True
Spread coefficient: 0.1226397630631558
```

8.7 Results for dolphins

Results for dolphins dataset.

```
--- Highest degree node ---
Nodes degrees:  [(15, 12), (38, 11), (46, 11), (52, 10), (34, 10)]
----------------
Starting points:  [15]
----------------
0 1 0.016129032258064516 False
1 13 0.20967741935483872 False
2 34 0.5483870967741935 True
3 47 0.7580645161290323 True
4 54 0.8709677419354839 True
5 61 0.9838709677419355 True
6 62 1.0 True
Spread coefficient: 0.08920342330480578
----- Dominating set  -----
----------------
Starting points:  {1, 2, 3, 4, 5, 6, 7, 8, 12, 13, 17, 19, 23, 24, 26, 32, 33, 35, 36
----------------
0 25 0.4032258064516129 False
1 62 1.0 True
Spread coefficient: 0.06330909956640583
```

```
---- Optimal strategy  ----
---------------
Starting points:  [41]
---------------
0 1 0.016129032258064516 False
1 9 0.14516129032258066 False
2 36 0.5806451612903226 True
3 52 0.8387096774193549 True
4 61 0.9838709677419355 True
5 62 1.0 True
Spread coefficient: 0.09856630824372759
---------------
Starting points:  [15, 18]
---------------
0 2 0.03225806451612903 False
1 23 0.3709677419354839 False
2 54 0.8709677419354839 True
3 62 1.0 True
Spread coefficient: 0.09908048647674406
---------------
Starting points:  [15, 18, 62]
---------------
0 3 0.04838709677419355 False
1 26 0.41935483870967744 False
2 55 0.8870967741935484 True
3 62 1.0 True
Spread coefficient: 0.08322690372390776
```

References

[1] Albert Laszlo Barabasi. *Network Science*. URL: http://networksciencebook.com/.

[2] Jeremy Blum Min Ding Andrew Thaeler Xiuzhen Cheng. *Connected Dominating Set in Sensor Networks and MANETs*. URL: https://www2.seas.gwu.edu/~cheng/Publication/CDSSurvey-Handbook.pdf.

[3] Organizational Evolution. *What is Network Density – and How Do You Calculate It?* URL: https://www.the-vital-edge.com/what-is-network-density/#:~:text=%5C%E2%5C%80%5C%9CNetwork%5C%20density%5C%E2%5C%80%5C%9D%5C%20describes%5C%20the%5C%20portion,or%5C%20not%5C%20it%5C%20actually%5C%20does..

[4] Kieran J.Sharkey. *Deterministic epidemic models on contact networks: Correlations and unbiological terms*. URL: https://www.sciencedirect.com/science/article/pii/S0040580911000128.

[5] Joel C. Miller Istvan Z. Kiss. "Epidemic spread in networks: Existing methods and current challenges". In: *Math Model Nat Phenom. 2014 Jan; 9(2): 4–42.* (). DOI: 10.1051/mmnp/20149202.

[6] Kiseong Kim Sunyong Yoo Sangyeon Lee Doheon Lee and Kwang-Hyung Lee. "Network Analysis to Identify the Risk of Epidemic Spreading". In: *Appl. Sci. 2021, 11, 2997* (). DOI: https://doi.org/10.3390/app11072997.

[7] Sam Moore and Tim Rogers. "Predicting the Speed of Epidemics Spreading in Networks". In: *PHYSICAL REVIEW LETTERS 124, 068301 (2020)* (). DOI: : 10.1103/PhysRevLett.124.068301.

[8] Networkx. *Documentation*. URL: https://networkx.org/documentation/stable/index.html.

[9] Filippo Radicchi Claudio Castellano Federico Cecconi Vittorio Loreto Domenico Parisi. *Defining and identifying communities in networks*. URL: https://www.pnas.org/content/101/9/2658.

[10] Ryan A. Rossi and Nesreen K. Ahmed. "The Network Data Repository with Interactive Graph Analytics and Visualization". In: *AAAI*. 2015. URL: http://networkrepository.com.

[11] John R. Ladd Jessica Otis Christopher N. Warren Scott Weingart. *Exploring and Analyzing Network Data with Python*. URL: https://programminghistorian.org/en/lessons/exploring-and-analyzing-network-data-with-python.

[12] Wikipedia. *Clustering coefficient*. URL: https://en.wikipedia.org/wiki/Clustering_coefficient.