Hand-in 1 FRTN30 Gustaf Sundell

Gustaf Sundell, gu0147su-s

April 2021

1 Acknowledgements

For this project, I have discussed the hand-in to a limited extent with Mark Emilson, Douglas Ihre and Fredrik Sidh.

2 Task 1

2.1 Task 1.a) - degree centrality

The out-degree and in-degree centralities of a graph $\mathcal{G}$ are proportional to the out-degree and in-degrees:

$$\begin{aligned} z_{out} &\propto w = W\mathbb{1} \\ z_{in} &\propto w^{-} = W'\mathbb{1} \end{aligned} \tag{1}$$

Since we are interested in the three nodes (sectors) with the highest centrality, proportionality will to just fine, as ordering is what is most interesting. As the centrality only needs to be proportional to the in- and out-degree, the magnitude of the answer does not matter. I.e., the magnitudes of the degrees are around 10^6 and 10^8 for Sweden and Indonesia respectively, but as proportionality is all that is needed here, the presented numbers have smaller magnitude. The results are presented in Table 1.

2.2 Task 1.b) - Eigenvector centrality of largest connected components

In order to solve this task, the largest connected components first needed to be accessed. The MATLAB function *conncomp*(), given in the instructions, was very helpful. The function takes in a *Graph*-object, and thus a MATLAB *digraph*-object was created for Sweden and Indonesia respectively. The constructor takes the adjacency matrix as input and *digraph* is preferred over *graph*, as the graphs are both directed, which must be captured. The *graph*-objects are now also plotable in MATLAB, the results in Figure 1. Already from the plots it becomes reasonably clear that Sweden has 6 completely isolated nodes, whereas Indonesia appears to already be connected.

By letting *conncomp*() partition each graph into connected components, it can be confirmed that Indonesia is indeed already connected, whereas Sweden has one large connected component and 6 isolated nodes.

The eigenvector centrality is a more refined method than degree centrality, and involves taking into account the centrality of the neighbours of each node. The solution ends up being finding the eigenvector corresponding to the dominant eigenvalue of W (or W', as they share eigenvalues). Also see chapter 2 in the lecture notes.

$$\lambda z = W'z \tag{2}$$

The eigenvector that should be chosen here is the dominant eigenvector of W, i.e. we set $\lambda = \lambda_W$. The eigenvector is then normalized by the sum of its elements, such that (see the same chapter):

$$z'\mathbb{1} = 1 \tag{3}$$

The results are summarized in Table 2.

2.3 Task 1.c) - Katz centrality for Sweden and Indonesia

The Katz centrality is a further refinement of eigenvector centrality, in that its centrality vector not only fulfills Equation 2, but rather that it fulfills a convex combination of $\frac{1}{\lambda_W}W'z$ and some chosen initial intrinsic centrality μ:

$$z^{(\beta)} = \left(\frac{1-\beta}{\lambda_W}\right) W'z^{\beta} + \beta\mu \tag{4}$$

$$\Longrightarrow$$

$$z^{(\beta)} = \left(I - \lambda_W^{-1}(1-\beta)W'\right)^{-1} \beta\mu \tag{5}$$

In Equation 4, $\beta \in (0, 1]$, ensuring that it is indeed a convex combination of the aforementioned vectors. In this task, the choice of both β and μ are done beforehand, with $\beta = 0.15$ and $\mu_1 = \mathbb{1}$ and μ_2 having all zeros except for in index 31, which contains the sector Wholesale and retail, as specified in the instructions. The results of this centrality are presented in Table 3.

3 Task 2

3.1 Task 2.a) - Iterative pagerank

To prepare the data for this task, the sparse matrix had to be made quadratic. This was done by adding zero-columns to the matrix, yielding a $n \times n$ matrix.

Pagerank is a version of the so-called Bonacci centrality. It is similar to Katz, but differs in that it uses the normalized weight matrix $P = diag(w)^{-1}W$, in stead of the adjacency matrix. In order to solve this task, P first needed to be computed, which was done by adding self-loops to those nodes with zero out-degree. In this setup, $\beta = 0.15$ once again and $\mu = n^{-1}\mathbb{1}$ were set. The instructions were to set $\mu = \mathbb{1}$, however the same ordering is obtained, only difference being that the numbers in the centrality vector don't blow up to too large values, and the computations seem quicker in MATLAB. More formally:

$$
\begin{aligned}
z^{(\beta)} &= \left(\frac{1-\beta}{\lambda_P}\right) P' z^{\beta} + \beta\mu && (6)\\
&\implies \\
z^{(\beta)} &= \left(I - (1-\beta)P'\right)^{-1} \beta\mu = \\
&= \beta \sum_{k=0}^{\infty} (1-\beta)^k (P')^k \mu && (7)
\end{aligned}
$$

In Equation 6 we utilize the fact that P is a stochastic matrix, having dominant eigenvalue $\lambda_P = 1$. Furthermore, we utilize the fact that the centrality vector is the reslt of the matrix equivalent of a geometric series, yielding the possibility of expressing $z^{(\beta)}$ as the result of a sum. This is how the algorithm was implemented in MATLAB, borrowing heavily from the code written in the solutions to Exercise 4. The pagerank centralities are presented in Table 4. Note however that sadly the usernames are omitted in the table, this is due to some rounding error, when I tried pasting the values from the result into the provided link, only about one or two usernames were attainable.

3.2 Task 2.b) - Discrete time consensus algorithm with 2 stubborn nodes.

The discrete-time consensus algorithm for this task does not necessarily converge to a consensus, as the graph is not connected (verified by looking at *conncomp*() for the graph, and by seeing some sources and sinks in the plot of the graph, see Figure 2). In this task, two nodes are set as stubborn, one with "opinion" 1, and one with 0. The rest of the nodes/twitter users are initially set as neutral, with opinion 0.5. For this task, the stubborn nodes were selected quite arbitrarily with node 3 having opinion 0 and node 9 having opinion 1.

Thereafter, the matrix P was partitioned following identical notation as in Chapter 6.3 in the lecture notes. The simulation for the discrete-time consensus algorithm was run for 1000 iterations and followed the following scheme:

$$
\begin{cases}
x_{regular}(t) = Qx_{regular}(t-1) + Ex_{stubborn}(t-1) \\
x_{stubborn}(t) = x_{stubborn}(t-1)
\end{cases}
\quad (8)
$$

Where x(t) denotes the opinion-vector at time t. The implementation of this part in MATLAB was also inspired by that of the solution to a similar problem from Excercise 4. The evolution of the opinion vector for this network can be seen in Figure 3. This is a quite messy plot, with many nodes, so see Figure **??** for the same plot, but tracking fewer nodes. The selection of these nodes was the stubborn ones, as well as some regular nodes tending to above and below their common initial opinion, namely 0.5.

As can be seen in Figure 3, node 9 is likely more central than node 3, as more people appear to be influenced by node 9's stubborn opinion.

3.3 Task 2.c) - Stubborn nodes with respect to Pagerank

In this task, the setup from the previous task remains the same, however we choose nodes that have high pagerank-centrality. For instance, choosing the nodes ranking 1 and 2 respectively result in Figure 5 (here presented as a histogram over the asymptotic opinion vector). Here, we see a clear tendency for the other nodes to converge towards the stubborn nodes, and it is also clear that the highest ranking node draws more followers (the one having 0 as stubborn opinion). The same was done with ranking 1 and 5, where a larger difference can be observed, see Figure 5. The conclusion here is that the pagerank centrality contributes to the influence of the nodes as stubborn.

4 Appendix - Tables and figures

Centrality type and country	**Rank**	**Sector name**	**Centrality**
Out-degree, Sweden	1	43 Other Business Activities	2.5782
	2	39 Real estate activities	1.1511
	3	31 Wholesale & retail trade; repairs	1.0581
In-degree, Sweden	1	19 Radio	1.4243
	2	21 Motor vehicles	1.4143
	3	43 Other Business Activities	1.3514
Out-degree, Indonesia	1	31 Wholesale & retail trade; repairs	1.7788
	2	1 Agriculture	1.7493
	3	2 Mining and quarrying (energy)	1.1014
In-degree, Indonesia	1	4 Food products	2.1446
	2	30 Construction	1.5068
	3	31 Wholesale & retail trade; repairs	1.1347

Table 1: Table of degree-centralities for task 1.a.

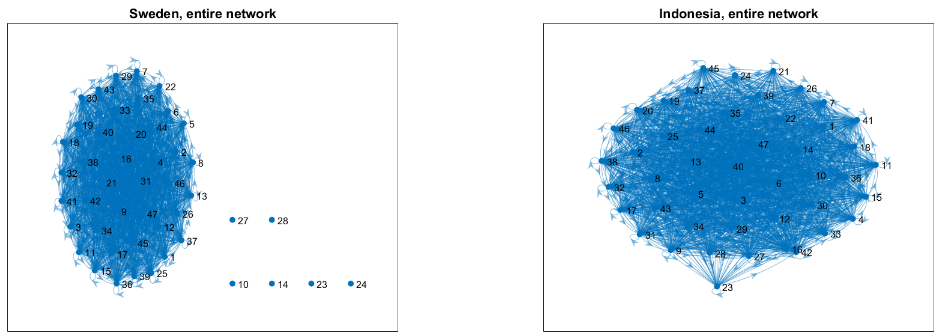

Figure 1: The networks of Sweden and Indonesia

Centrality type and country	**Rank**	**Sector name**	**Centrality**
Eigenvector, Swedens largest connected component	1	21 Motor vehicles	0.1179
	2	19 Radio	0.0975
	3	43 Other Business Activities	0.0774
Eigenvector, Indonesia's largest connected component (the initial graph)	1	4 Food products	0.4721
	2	32 Hotels & restaurants	0.1434
	3	1 Agriculture	0.1418

Table 2: Eigenvector centrality of largest connected components in Sweden and Indonesia

Katz-centrality	Rank	Sector name	Centrality
Sweden, mu = ones	1	21 Motor vehicles	2.2317
	2	19 Radio	2.0082
	3	43 Other Business Activities	1.6989
Sweden, mu = 1 for Wholesale only	1	31 Wholesale & retail trade; repairs	0.2071
	2	21 Motor vehicles	0.1087
	3	19 Radio	0.0903
Indonesia, mu = ones	1	4 Food products	3.0172
	2	32 Hotels & restaurants	1.0233
	3	1 Agriculture	1.0128
Indonesia, mu = 1 for Wholesale only	1	4 Food products	0.4396
	2	31 Wholesale & retail trade; repairs	0.1883
	3	32 Hotels & restaurants	0.1479

Table 3: Katz centrality for Sweden and Indonesia with the two different specified intrinsic centralities

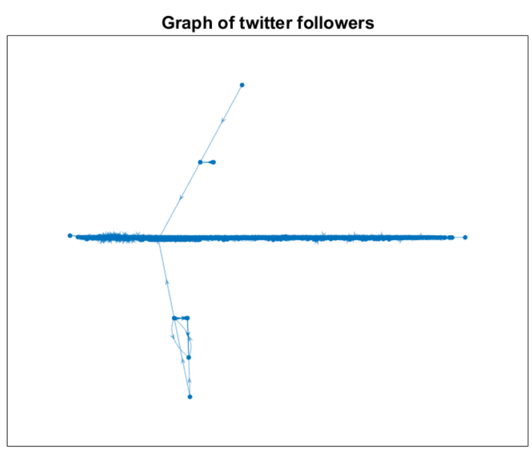

Figure 2: The Twitter Graph

Pagerank-ranking	User id	Centrality
1	876641	0.1042
2	99123165	0.0178
3	483996827	0.0119
4	1128920917	0.0103
5	4417595361	0.0091

Table 4: Pagerank of top 5 twitter users. Unfortunately, some rounding errors made it difficult to get their usernames.

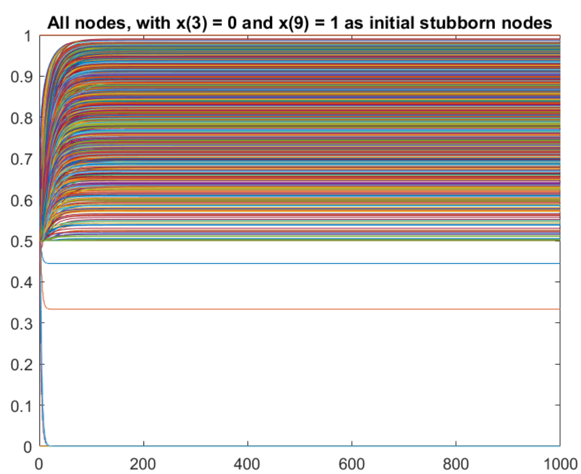

Figure 3: All nodes plotted from task 2.b)

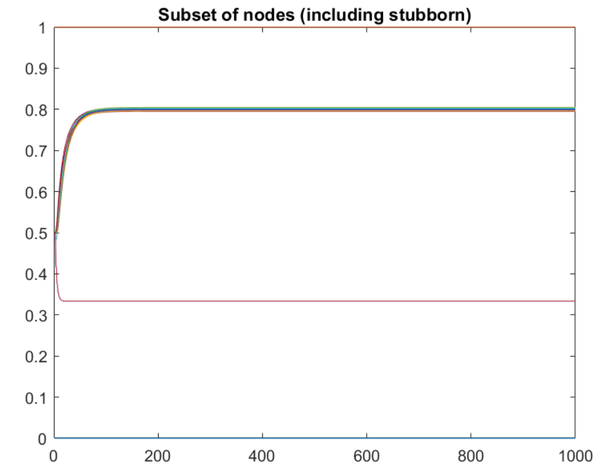

Figure 4: Subset of nodes from task 2.b)

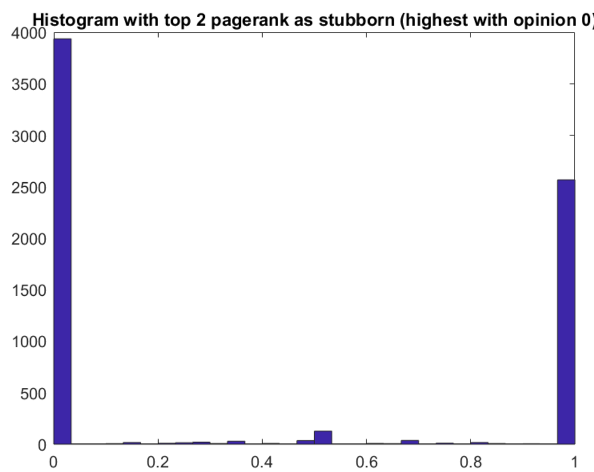

Figure 5: Asymptotic opinion vector distribution with top 2 pagerank nodes as stubborn

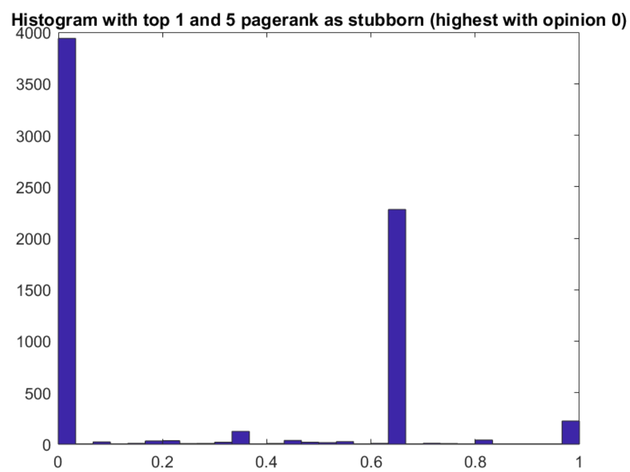

Figure 6: Asymptotic opinion vector distribution with 1st and 5th pagerank nodes as stubborn.