1 Method

- computer aided network analysis ¡- distinction between 'verkostotutkimus' mentioned in Juuso Marttila's thesis. Quite a few textbooks have been written on network analysis.

Furthermore, in the field of technology network analysis does have some everyday application, such as the analysis of the internet.

1.1 Defining the network

Network analysis is based on the mathematical graph theory. A graph is a representation of the network. Graph includes nodes and edges.

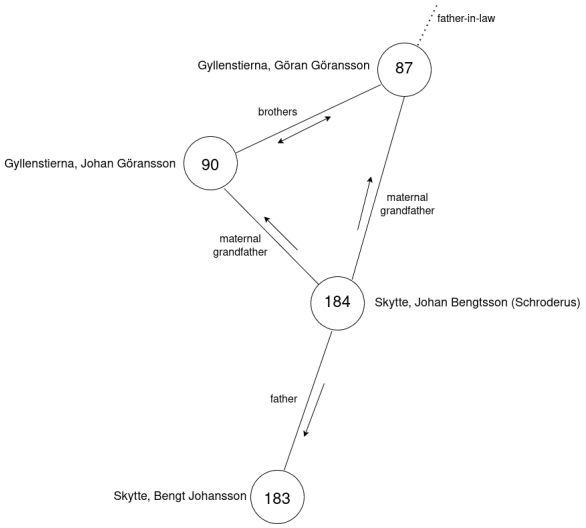

Figure 1: A sample from the graph

In this context the graph's nodes depict individual councillors with the input of name and id number. Correspondingly the edges represent the kinships between two nodes. For instance, in Figure 1 we can see that Johan Bengtsson (Schroderus) Skytte (id 184) is Bengt Johansson Skytte's (id 183) father and a maternal grandfather for Johan Göransson Gyllenstierna (id 90) and Göran Göransson Gyllenstierna (id 87). Johan Göransson and Göran Göransson are brothers, however, their father is not mentioned in the dataset. Göran Göransson also has further links in the network.[1]

[1]Marko Hakanen and Ulla Koskinen. 2017. *Swedish Councillors of the Realm, 1523-1680*. doi:`https://doi.org/10.17011/jyx/dataset/55523`.

1.2 Implementation of the network analysis

The data processing and analysis is conducted with a combination of Python programming language and Gephi software. Python is used for extracting the data from the councillors-dataset and formulating it in the right format: readable for Gephi. The actual network analysis, visualization and calculating statistics, is performed with Gephi.

Python is a programming language commonly used in scientific work. On my opinion simple syntax, easy to implement smaller tasks such as data processing. Readable, widely used therefore makes the work replicable. To be precise the script is written with Python 3.

As graphs are structures commonly used in programming, it would have been possible to conduct the actual network analysis using tools provided by Python, yet, Gephi software provides a visual user interface and more intuitive tools for the manipulation of the graph.

Gephi is ...

Gephi is not always the most intiutive to use. Problems especially with node labels. [2]

Both of these tools are also open source and free to download.

All scripts written for this work available on GitHub(TODO link)

Basically the steps of network analysis are : ... These will be discussed in detail in the next subsection.

1.2.1 Test run

TODO add also to the source section about the end TODO fix councillor / councillor typo

To draft the structure of the graph and understand the nuances of the given data, a test run was carried out. The test run was done with a simple Python script, and no attention was paid to the temporal aspects of the network or the potential directions within the graph. The script and Gephi project used, and the visualization of the graph of the test run is available in GitHub in the TestRun folder[3]

The data processing was started by manually cleaning the data in LibreOffice Calc (equivalent to Microsoft Excel). The columns and rows containing information of the source material of the dataset and councillor's years active were removed. That made the structure of the data coherent and easier to manipulate with the Python script. The manually cleaned data is exported

[2]For Linux environments opening Gephi from command line with command "LIBGL_ALWAYS_SOFTWARE=1 ./gephi" can sometimes help.

[3]`https://github.com/Heidi-Suurkaulio/mastersthesis/tree/main/TestRun`

as .csv (comma separated values) file. The .csv file's header (the first line of the file) should be modified so that the column name "No." is changed to "Id" and "Family members in the council of the realm" is changed to "Family", the first one can cause an error if referenced in the Python code, the latter is inconveniently long.

Table 1: Example of the raw .csv file

1	Name;	Id;	D.O.B.;	died;	Appointed;	Date;	Age;	Noble rank;	Family;	Spouse(s) / Father of Spouse / Date of Marriage
2	Ingemar Petri;	162;	;	1530;	1495;	;	;	Estate unknown, Bishop;	;	;
3	Tre Rosor, Ture Jönsson;	231;	;	1532;	1497;	;	;	Uradel (Ancient Nobility);	Father CR, Father-in-law CR, Sons 228, 230, Son (illegitimate) 175;	Anna Johansdotter/ Johan Christiernsson Vasa (CR)

The script itself reads the data from the .csv file. The connections between the councillors are separated from the "Family" column, based on the knowledge that each connection is marked with the id number of another councillor. The connections are then formatted and printed to .csv file. The connections .csv file containing values for "Source" id of the source concillor, "Target" id of target councillor, "Type" standard "Undirected", "Id" id number for the connection, "Weight" standard 1.0. Another .csv file is formatted and printed with the information of councillors' names and id numbers.

Table 2: Example of the connections .csv file

1	Source,	Target,	Type,	Id,	Weight
2	231,	228,	Undirected,	0,	1.0
3	231,	230,	Undirected,	1,	1.0

Table 3: Example of the councillors .csv file

1	Id;	Label
2	162;	Ingemar Petri
3	231;	Tre Rosor, Ture Jönsson

These .csv files are readable for Gephi. The outcome was an undirected graph of the councillors' affiliation network that had accumulated during the 160 years. The graph consisted of 261 nodes (257 real + 4 "ghosts") and 372 edges (including self loops and "ghost" nodes). The test run revealed three problems within the graph: the emergence of the empty "ghost" nodes, parallel edges and thirdly self loops.

The "ghost" nodes were excess nodes with no name and only an id number and one or two connections in the graph. They were due to the references

to the data points removed from the original dataset, and therefore can be ignored. The ghosts are discussed further in the subsection sources. However, the more essential problem were parallel edges and self loops.

The parallel edges occur because one relationship, such as father and son, is sometimes marked parallel in the dataset. For example, in the case of Göran Göransson Gyllenstierna (id 87) the relatives are "Maternal Grandfather 184, Brother 90, Father-in-law 3, ...", and the same relationship is found in his grandfather's Johan Bengtsson (Schroderus) Skytte's (id 184) links: "Son 183, Grandson through daughter 87". Yet, the connection to Göran Göransson's brother Johan Göransson Gyllenstierna (id 90) is not marked in the grandfathers links. This means that the node of Göran Göransson Gyllenstierna (id 87) has one excess link compared to his brother's node. The case is visualized in the Figure 2.

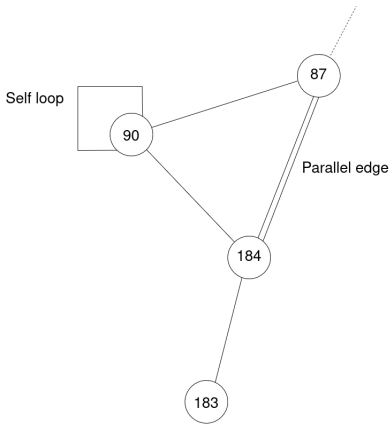

Figure 2: Visualisation of the parallel edge and self loop

These duplicate edges would cause bias to the calculation of the node degrees and any statistics based on them. A node degree is a sum of all the edges connected to one node, and if the relationships are inconsistently marked with one or two edges, the factually similar nodes would get different degrees. These inconsistent node degrees would accumulate when counting the average degrees an so forth. The problem of parallel edges is widely recognized in the field of network analysis, and therefore Gephi does have some builtin features for handling it.

While importing data to Gephi (on Import Spreadsheet) the strategy for merging the parallel edges can be chosen. One option is, for example, placing the sum or average of the parallel edges in the edge's degree, yet using only one connection to to represent the edge in the graph. In this context a more simple solution was chosen, with the option "Firs" Gephi will use only the

first connection between two nodes ignoring any latter ones. This will reduce the amount of connections from 698 found in the connections.csv to only 372.

Self loops occur when one node has – for some reason or another – a connection to itself. Similarly to the parallel edges, they cause bias to the node degrees. In this graph a self loop can be found at least on the node with id 5 and id 90. In the case of id 5: Gustaf Axelsson Baner, his relatives are "Father 4, Father-in-law 217, Brother 9, Sons 5, 7, 8, 10, Sons-in-law 152 and 197", and similarly with id 90: Johan Göransson Gyllenstierna his family reads "Maternal Grandfather 184, Brother 90". These self loops are most likely caused by a typo in the dataset, because it is reasonable to assume that none is a son or brother to themselves.

Gephi does have a switch whether or not self loops are allowed in the graph, and it can automatically remove them based on the preference. The self loops are present in the test run graph alongside with the ghost nodes, yet those will be removed from the subsequent analysis. To highlight the ghosts they are colored cray, and the four nodes referred as an example here are colored red in the test run graph.

The last step in the preparation of the network analysis is the selection of the layout algorithm. For the test run an algorithm called Yifan Hu was used with default configurations except parameter theta set to 2.0. Then layout option "noOverlap" was chosen to separate possibly overlapping nodes, and some further manual placement of the nodes was done to make the graph more readable. The outcome was visually somewhat dense network in the middle and mostly unconnected isolated nodes around it.

Problems: we don't have data about the women