Athens University of Economics and Business

School of Business

Department of Management Science & Technology

Master of Science in Business Analytics

Program:	Full-time
Quarter:	3rd (Spring Quarter)
Course:	Social Network Analysis
Instructor:	Dr. Katia Papakonstantinopoulou
Assignment №:	Project 1
Students (Registration №):	Souflas Eleftherios - Efthymios (f2822217)

Table of Contents

Project 1: Network Analysis and Visualization with R and $igraph$

Introduction

In this Project, we used 'A Song of Ice and Fire' network of characters to create an undirected weighted graph through R and $igraph$, analyzed its properties, studied its structure, calculated some of its metrics and created some visualizations through our answers on the problems that are presented below.

Problem 1: 'A Song of Ice and Fire' network

We firstly loaded into R the file with the list of edges of the network of characters from 'A Song of Ice and Fire' by George R. R. Martin, found in GitHub. The file load was done not by downloading the file and reading it from the local filesystem, but by opening a connection with GitHub's raw data and reading its content. From this file we used the columns Source, Target, and Weight to create an undirected weighted graph using R's $igraph$ library.

Problem 2: Network Properties

Then, having created the forementioned graph and with the use of $igraph$'s commands, we explored its basic properties:

- The number of vertices in the network is 796.
- The number of edges is 2823.
- The diameter i.e., the length of the shortest path that connects the two most distant nodes is 53.
- The number of triangles in the graph is 5655 and was calculated by summing the number of triangles each vertex is part of and dividing the result by 3, because otherwise each triangle would be reported three times, one for each distinct vertex.
- The number of edges having weight more than 15 is 478.
- The top 10 characters of the network as far as their degree in descending order is concerned are: Tyrion Lannister, Jon Snow, Jaime Lannister, Cersei Lannister, Stannis Baratheon, Arya Stark, Catelyn Stark, Sansa Stark, Eddard Stark, and Robb Stark.
- The top 10 characters of the network as far as their weighted degree in descending order is concerned are: Tyrion Lannister, Jon Snow, Cersei Lannister, Joffrey Baratheon, Eddard Stark, Daenerys Targaryen, Jaime Lannister, Sansa Stark, Bran Stark, and Robert Baratheon.
- There exist 178 characters in the network that have local clustering coefficient equal to the maximum number that it can get i.e., 1. The top 10 of them in ascending alphabetic order are: Aegon Frey (son of Stevron), Albett, Alerie Hightower, Allar Deem, Alys Karstark, Alysane Mormont, Amabel, Arron, Baelor Blacktyde, and Baelor I Targaryen. For the whole set of characters, you can view the respective R code.
- The global clustering coefficient of the graph i.e., the degree to which its nodes tend to cluster together, is approximately 0.209.

Problem 3: Subgraph

A plot of the entire network is shown in Figure 1. For the creation of this plot, we chose not to show the labels of each vertex, make each edge and vertex relatively small and plot the entire graph using a sphere layout.

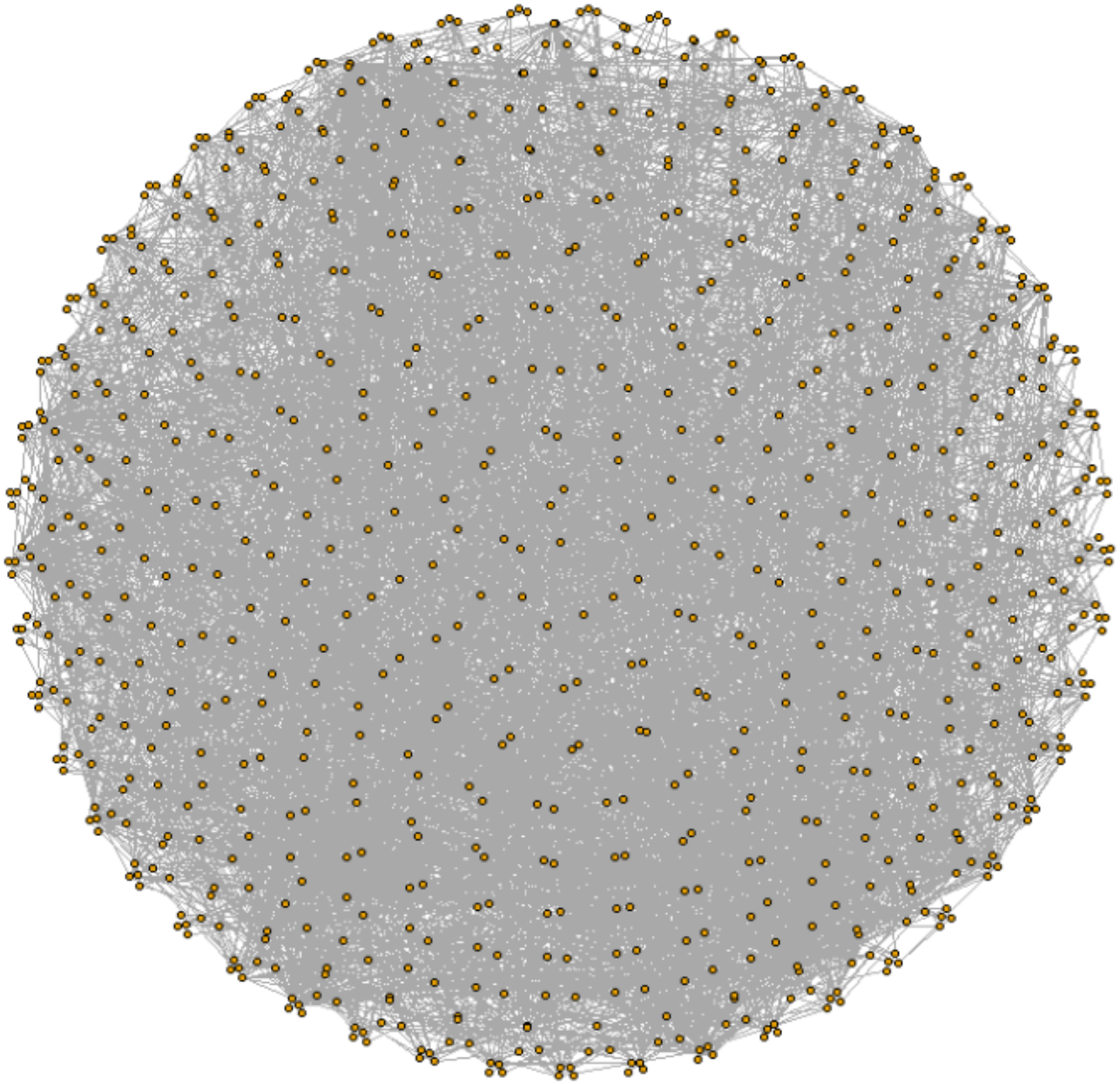

Figure 1 – Entire network sphere layout plot

Then, we created a subgraph by discarding all vertices that had less than 10 connections in the network, but plotted the entire set of vertices, as shown in Figure 1, with only the edges between the vertices belonging in the chosen subgraph (having 10 or more connections), in order to spot the location of them in the previous (entire) network as it was depicted in Figure 1. As we can derive from the resulting plot (Figure 2), in the sphere layout plot, the more connections a vertex has, the closer to the core of the sphere is placed.

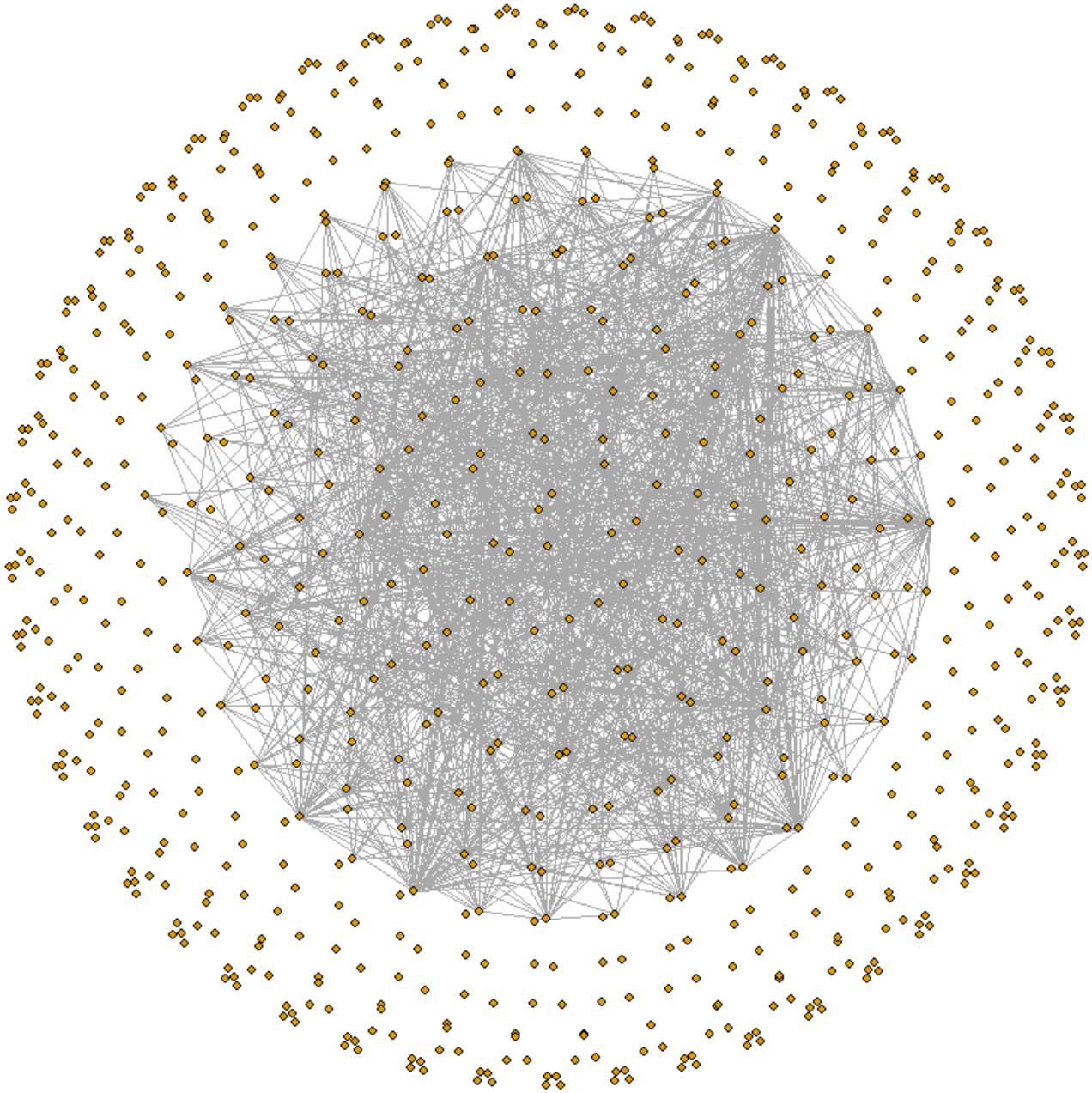

Figure 2 - Entire network sphere layout plot with edges of those vertices belonging in the subgraph

Then, we zoomed in the subgraph and plotted it alone, as shown in Figure 3 and calculated the edge density of the entire graph and the subgraph, which were approximately 0.009 and 0.117 respectively. Thus, the subgraph was more than 13 times denser than the entire graph. This is due to the definition of edge density, which is defined as the ratio of the actual number of edges to the largest possible number of edges in the graph i.e., every vertex being connected to everyone else. The number of vertices in the subgraph were found to be 137 and the number of edges 1090, thus in a subgraph having less than sub-fivefold number of vertices exist more than a third of the edges found in the whole graph.

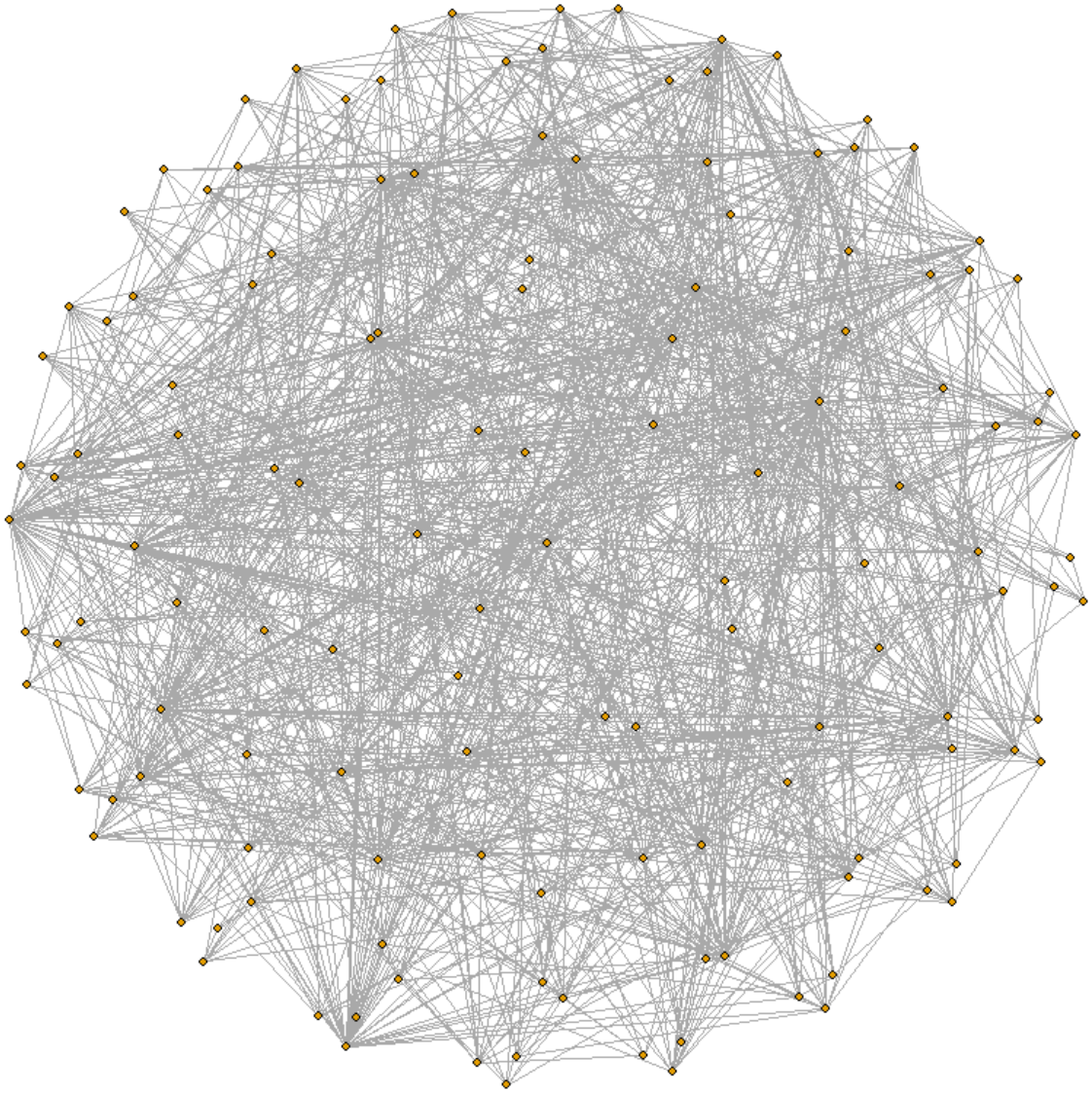

Figure 3 - Subgraph sphere layout plot

Problem 4: Centrality

The top 15 nodes as far as their closeness centrality in descending order is concerned are those depicting the following characters: Jaime Lannister, Robert Baratheon, Stannis Baratheon, Theon Greyjoy, Jory Cassel, Tywin Lannister, Tyrion Lannister, Cersei Lannister, Brienne of Tarth, Jon Snow, Joffrey Baratheon, Rodrik Cassel, Eddard Stark, Doran Martell, and Robb Stark.

The top 15 nodes as far as their betweenness centrality in descending order is concerned are those depicting the following characters: Jon Snow, Theon Greyjoy, Jaime Lannister, Daenerys Targaryen, Stannis Baratheon, Robert Baratheon, Tyrion Lannister, Cersei Lannister, Tywin Lannister, Robb Stark, Arya Stark, Barristan Selmy, Eddard Stark, Sansa Stark, and Brienne of Tarth.

Jon Snow is ranked 10^{th} regarding his closeness centrality and 1^{st} regarding his betweenness centrality. This means that although Jon Snow needs more steps to access every other character in the network than nine (9) other characters do, the greatest number of shortest paths between every pair of characters in the network pass through Jon Snow. Thus, we can derive that Jon Snow has the largest amount of influence over the flow of information in the network.

Problem 5: Ranking and Visualization

In the final step of this project, we ranked the characters of the network with regards to their PageRank value. Then, we created a plot of the entire network and highlighted characters that got the highest rankings with a bigger node and font size. We firstly used a grid layout for the labels to be better placed. However, because still some important with regards to PageRank value characters were placed next to each other, some of the labels could not be clearly read. Thus, we used the $tkplot$ function of R's $igraph$ library to interactively tune the placement of the vertices in specific areas of the grid layout plot for the labels to be more easily readable and then queried the coordinates of the vertices and wrote them in a separate txt file. Then, with the use of this txt file that specified the layout of the vertices, we created a plot that made highly ranked vertices bigger in size and placed labels on vertices that their characters exceeded PageRank value of 0.1. The resulting plot is shown in Figure 4.

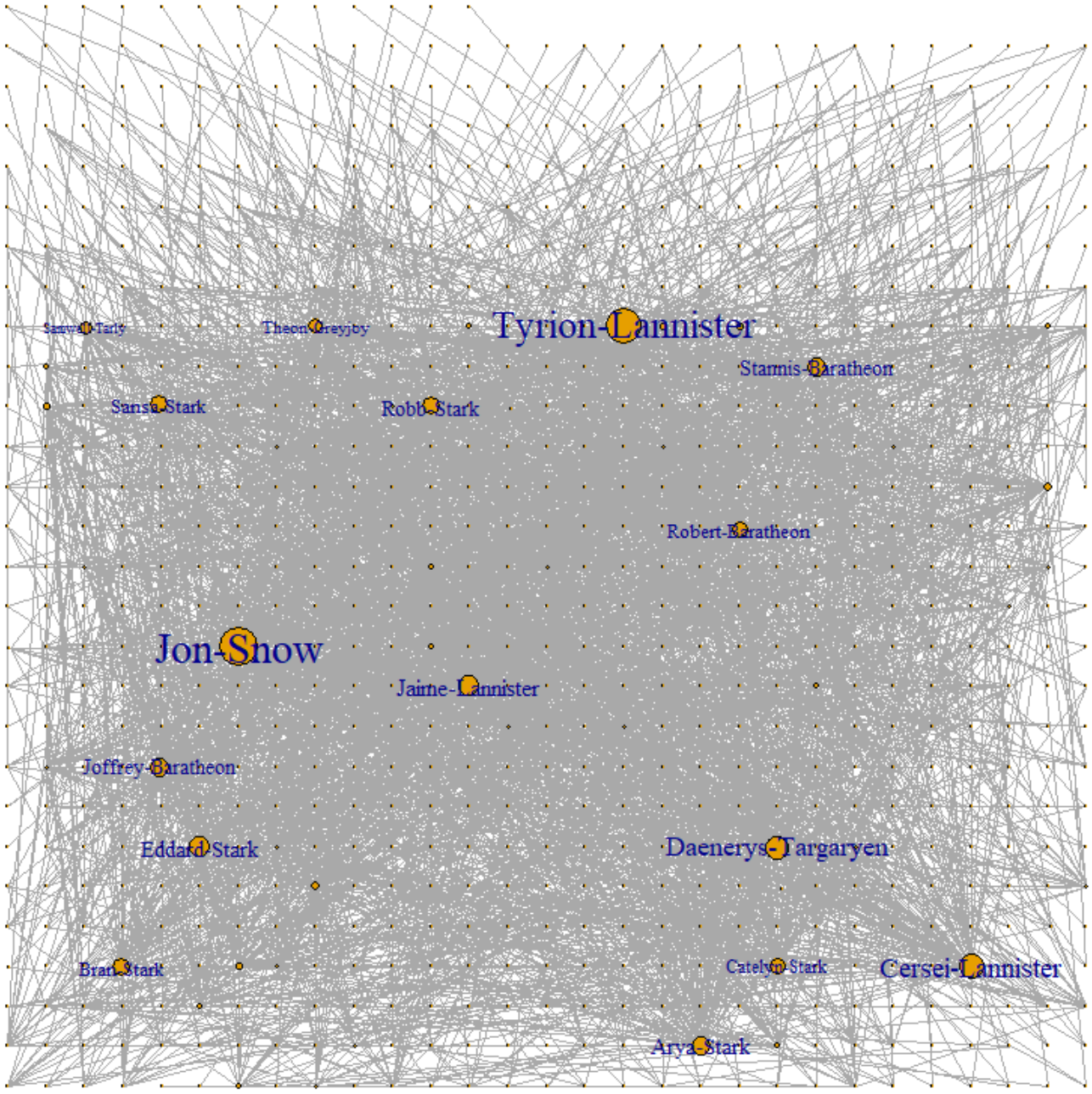

Figure 4 - PageRank grid layout plot