

Statistical Methods for Social Network Analysis

Alexandros Moustakas

S - 6210050

Supervisor: Professor Dimitrios Karlis



Στατιστικές Μέθοδοι για την Ανάλυση Κοινωνικών Δικτύων

Αλέξανδρος Μουστάκας

S - 6210050

Επιβλέπων: Καθηγητής Δημήτριος Καρλής

Acknowledgments

First of all, I would like to sincerely thank my supervisor, Professor Dimitrios Karlis, for his guidance and encouragement throughout the course of this thesis. I am also very grateful for the knowledge I gained through his taught courses during my studies, where he emphasized a modern approach to statistics and highlighted the importance of methodological rigor.

I would also like to thank Professor Nikolaos Demiris and Professor Panagiotis Papastamoulis for their time and participation in the thesis commitee.

Lastly, I thank my parents and grandparents for the continuous support and motivation throughout my studies.

Short Biography

Alexandros Moustakas is a final – year student at the Department of Statistics, AUEB. During his studies, he was honored to receive the department's academic performance awards. During the summer of 2024 he joined the Microeconomic Data & Surveys Section of the Statistics Directorate, Bank of Greece as an intern. He finds statistical modelling and statistical machine learning very interesting, especially their applications to biostatistics and health.

Contents

1 Introduction **1**

2 Εισαγωγή στα Ελληνικά **3**

3 Visualization and Descriptive Statistics **5**

3.1 Fundamental Network Concepts 5

3.1.1 Directed Networks 5

3.1.2 Weighted Networks 5

3.1.3 Network Cliques 6

3.2 Special Cases of Networks 6

3.2.1 Some Special Cases 7

3.2.2 Bipartite Networks 7

3.3 Network Descriptive Analytics 8

3.3.1 Vertex Degree and Strength 8

3.3.2 Network Density 9

3.3.3 Vertex Centrality Measures 9

3.3.4 Clustering Coefficient 11

3.3.5 Average Path Length 11

3.4 Visualization Layouts 11

3.4.1 Kamada - Kawai Algorithm 11

3.4.2 Fruchterman - Reingold Algorithm 13

3.4.3 Visualization Enrichment 13

4 Some Statistical Models **15**

4.1 Elements of Bayesian Statistics 15

4.1.1 Metropolis - Hastings Algorithm 16

4.1.2 Gibbs Sampling 16

4.2 Erdős–Rényi Model 18

4.2.1 MLE Estimation 18

4.2.2 Properties of Erdős–Rényi 19

4.3 The p1 model 19

4.4 Network Block Model 21

4.4.1 MLE Estimation 21

4.5 Latent Distance Model 22

4.5.1 Metropolis - within - Gibbs Estimation 23

4.6 ERGM 25

4.6.1 Probability of an edge forming 26

4.6.2 MCMC - MLE Estimation 27

5 Community - Detection Methods **29**
5.1 Hierarchical Clustering . . . 29
5.2 Stochastic Block Model . . . 33
5.3 Latent Position Cluster Model . . . 33
5.3.1 Two - Step Approach . . . 34
5.3.2 Bayesian Estimation . . . 36

6 Application in real data **39**
6.1 Descriptive Analytics . . . 39
6.2 Statistical Models . . . 43
6.2.1 Erdős–Rényi Model . . . 43
6.2.2 Network Block Model . . . 45
6.2.3 Latent Distance Model . . . 47
6.2.4 Exponential family Random Graph Model . . . 48
6.3 Community - Detection . . . 50
6.3.1 Hierarchical Clustering . . . 51
6.3.2 Stochastic Block Model . . . 51
6.3.3 Latent Position Cluster Model . . . 53

7 Conclusions **57**

A Model Summary Tables **58**

B Packages Used **60**

List of Figures

1 A toy-network featuring cliques 6
2 Visualization of the special network cases. 7
3 A simple bipartite network. 8
4 A comparison of enriched versus generic visual. 14
5 The latent positions of the vertices of Figure 1. 23
6 The dendrogram of hierarchical clustering in the network of Figure 1. 30
7 A comparison of Jaccard versus geodesic distances in clustering of Figure 1 network. . . 32
8 The network of academic collaborations between AUEB and UNIPI faculty members. . . 39
9 Layouts of the individual networks. 40
10 Empirical degree distributions of the vertices of the two networks. 41
11 Empirical strength distributions of the vertices of the two networks. 42
12 Network visualization encompassing centrality measures as node size. 43
13 Comparing expected degree under the Erdős–Rényi with sample network data. 44
14 Empirical degree distribution for Erdős–Rényi simulated networks. 45
15 The latent positions of the academic collaboration network 47
16 Simulation goodness - of - fit for the Erdős–Rényi and block model ERGMs. 49
17 A simulated directed citations network. 50
18 Hierarchical Clustering in the presence of disconnected nodes. 51
19 Comparing ICL for different Q specifications. 52
20 The resulting $Q = 3$ communities of the SBM. 52
21 The plot with community membership probabilities of the LPCM. 54
22 The plot with community membership probabilities for the SBM. 54
23 The resulting communities of the LPCM after MAP rule. 55
24 Side - by - side comparison of the resulting communities of SBM & LPCM. 56

List of Tables

1 Different possible k-cliques in the network of Figure 1 and their frequency. 6
2 Cluster covariance matrix constraining. 36
3 Comparison of network densities and transitivities of two universities. 40
4 Diameter and Average Path Length Descriptive Statistics. 41
5 Vertex degree relative frequency table. 42
6 Comparing AIC and BIC criteria between Erdős–Rényi and block model. 46
7 Erdős – Rényi through ERGM model summary. 58
8 Block Model through ERGM model summary. 58
9 Latent Distance Model without covariates model summary. 58
10 Latent Distance Model with categorical covariate model summary. 59
11 ERGM for the simulated directed citations network model summary. 59
12 Latent Position Cluster Model summary. 59

1 Introduction

Social networks, also referred to as graphs, are a collection of vertices connected with edges, which can represent numerous real – world phenomena. They have a wide scope of applications, including monitoring the spread of infectious diseases (epidemiology), friendship and relationship networks (social media), financial transaction networks (business and AML), observing the flow of information in server networks (IT), recording the sightings of species in habitats (ecology) and more.

Statistical analysis is accustomed to addressing dependencies in the data. For instance, time dependence in time – series data, spatial dependence in geospatial data, or both simultaneously. Network data, on the other hand, may present a unique set of dependencies. Conversely, reciprocity (tendency of directed edges to be mutual), homophily (tendency to form edges with vertices with similar characteristics), preferential attachment (vertices with higher degree form more edges), transitivity (tendency of vertex triads to form triangles), clique and k – core formation etc. Hence, appropriate specialized techniques will be discussed which capture these phenomena.

Network data present additional challenges, compared to typical tabular data. Indicatively, is there a correct way to place vertices as points in a two - dimensional plane and connect them with lines representing edges in order to visualize a network? The answer is naturally no, making network visualization an exciting objective.

The goal of social network analysis in most aforementioned disciplines is more or less the same: discover influential vertices (descriptive statistics), assess the size and structure of the network (descriptive statistics), visualize the network in a way as to allow some useful information to stand out (visualization), explain the mechanism behind edge creation (statistical models) and uncover hidden communities within the network (community – detection models).

The aim of this thesis is to discuss the established network descriptive statistics and present modern statistical models for network data and their interesting estimation methods. After the theoretical establishment of the statistical methods, their application to real, academic collaboration network data follows. The practical application should be considered equally important, as interesting practical and case - specific issues appear.

2 Εισαγωγή στα Ελληνικά

Τα κοινωνικά δίκτυα, τα οποία αναφέρονται συχνά και ως γράφοι, αποτελούν ένα σύνολο κορυφών ή κόμβων που συνδέονται με ακμές και μπορούν να αναπαραστήσουν ποικίλα φαινόμενα του πραγματικού κόσμου. Έχουν μεγάλο πλήθος εφαρμογών, όπως για παράδειγμα η παρακολούθηση της εξάπλωσης λοιμωδών νόσων (στην επιδημιολογία), δίκτυα φίλων, κοινωνικών σχέσεων και αλληλεπιδράσεων (στα κοινωνικά δίκτυα), οικονομικές συναλλαγές (στις επιχειρήσεις), παρατήρηση της ροής της πληροφορίας σε δίκτυα διακομιστών (στη πληροφορική), καταγραφή των παρατηρήσεων οργανισμών σε οικοσυστήματα (στην οικολογία) και πολλές άλλες.

Η μέθοδοι στατιστικής ανάλυσης δεν είναι ξένες στην αντιμετώπιση εξαρτήσεων στα δεδομένα. Για παράδειγμα, αντιμετωπίζουν χρονική εξάρτηση σε δεδομένα χρονοσειρών, χωρική εξάρτηση σε χωρικά δεδομένα ή και τις δύο αυτές εξαρτήσεις ταυτόχρονα. Τα δεδομένα δικτύων, ωστόσο, παρουσιάζουν ένα σύνολο ιδιαίτερων εξαρτήσεων. Μερικές από αυτές είναι η αμοιβαιότητα (τάση των κατευθυνόμενων ακμών να είναι αμοιβαίες), η ομοφυλία (τάση των κόμβων να δημιουργούν ακμές με κόμβους παρόμοιων χαρακτηριστικών), η προσκόλληση υπό προτίμηση (το φαινόμενο κόμβοι με υψηλό βαθμό ακμών να αποκτούν περισσότερες ακμές), μεταβατικότητα (η τάση τριάδων κόμβων να γίνονται τρίγωνο), κλίκες και κ – πυρήνες κ.α. Για το λόγο αυτό, κατάλληλες, εξειδικευμένες τεχνικές θα συζητηθούν, που συλλαμβάνουν αυτά τα φαινόμενα.

Τα δεδομένα δικτύων παρουσιάζουν επιπλέον προκλήσεις, σε σύγκριση με τα συνηθισμένα δεδομένα που απεικονίζονται σε πίνακες. Ενδεικτικά, υπάρχει σωστός ή κατάλληλος τρόπος να τοποθετηθούν οι κόμβοι ως σημεία σε ένα δισδιάστατο επίπεδο και να συνδεθούν με ακμές; Η απάντηση, φυσικά, είναι πως όχι, κάτι το οποίο καθιστά την οπτικοποίηση δικτύων ενδιαφέρουσα.

Τα κίνητρα πίσω από την ανάλυση κοινωνικών δικτύων στις περισσότερες επιστήμες είναι πάνω – κάτω τα εξής: αποκάλυψη κόμβων υψηλής επιρροής (περιγραφική στατιστική), αξιολόγηση του μεγέθους και της δομής του δικτύου (περιγραφική στατιστική), οπτικοποίηση του δικτύου με τρόπο τέτοιο ώστε να αναδεικνύεται κάποια χρήσιμη πληροφορία (οπτικοποίηση), εξήγηση του μηχανισμού πίσω από την δημιουργία ακμών (στατιστική μοντελοποίηση) και, τέλος, ανακάλυψη υπό - κοινοτήτων μέσα στο δίκτυο (συσταδοποίηση/ομαδοποίηση).

Ο στόχος αυτής της εργασίας είναι να συγκεντρώσει και να συζητήσει τις καθιερωμένες μεθόδους περιγραφικής ανάλυσης δικτύων και να παρουσιάσει μοντέρνα στατιστικά μοντέλα που αφορούν δεδομένα δικτύων, μαζί με τις ιδιαίτερες μεθόδους εκτίμησής τους. Έπειτα από το θεωρητικό μέρος, ακολουθεί η εφαρμογή των κατάλληλων μεθόδων σε πραγματικά δεδομένα. Η πρακτική εφαρμογή έχει ισοδύναμη αξία με το θεωρητικό μέρος, καθώς μέσα από αυτή προκύπτουν ορισμένα ενδιαφέροντα πρακτικά ζητήματα.

3 Visualization and Descriptive Statistics

3.1 Fundamental Network Concepts

A network refers to a system of inter-connected entities. It is intuitively represented by a graph $G = (V, E)$, which is a structured collection of vertices: V and edges: E. Vertices, often also called nodes, refer to the entities in the system and edges, or links, refer to the connections between those entities. For instance, vertices could represent students attending various schools in an area and edges could portray friendships between them.

3.1.1 Directed Networks

Naturally, networks are subdivided into various types. Firstly, they can be either directed or undirected. Directed networks capture relationships which, by nature, may not be mutual. On the other hand, undirected networks capture mutual relationships. To explain these two types, the case of social networking apps will be employed. In the case of apps such as LinkedIn or Facebook, a connection (edge) between two users (vertices) is mutual. Thus, they could be represented by undirected graphs. However, in apps like Instagram or Tik-Tok, where one user has to follow another to keep up with their activity, relationships are not necessarily mutual. For example, Leo may follow Phil, but Phil may not certainly follow him back. In the previous example, if Phil were to follow Leo back, this would constitute a mutual edge Kolaczyk & Csárdi (2014). Therefore, undirected graphs can be thought of as directed graphs were all edges are mutual. Some examples of directed networks are:

- Financial transactions: Company A wires Company B an amount.
- Interpersonal relationships: Person A likes Person B but Person B may not like Person A.
- Sports: Player A passes the ball to Player B.

3.1.2 Weighted Networks

The next case would be of weighted or unweighted networks. Weighted networks are cases of networks where edges are accompanied by some attribute, a weight Kolaczyk & Csárdi (2014). In this case, E, are non-negative numbers. Some examples of weighted networks are:

- Online communication: User A sent User B X e-mails.
- Financial transactions: Company A owes Company B an amount of Y monetary units.
- Social Media: User A has interacted with Z posts of User B.

In these examples, the weights are X, Y and Z accordingly. Also, one can notice that, the previous examples refer to directed networks. Most often, networks are communicated through an adjacency matrix A, which is a matrix of dimensions $|V| \times |V|$, where $|V|$ refers to the number of vertices. Its elements are:

$$A_{i,j} = \begin{cases} 1 & \text{if vertices } i \text{ and } j \text{ are connected,} \\ 0 & \text{if vertices } i \text{ and } j \text{ are not connected.} \end{cases}$$

In the case of weighted networks, the elements $A_{i,j}$ would be a non-negative number, if nodes i and j were connected, which would indicate the weight and 0 if i and j were not connected.

3.1.3 Network Cliques

Generally, complete graphs are also referred to as **cliques**. A k-clique is formally defined as a sub-graph, namely a subset of of k vertices, where every vertex is connected to every other of the $k-1$ in the subset. Presence of cliques serves as an indication of patterns in the network, meaning the network is not random. For example, the following network in Figure 1 presents cliques.

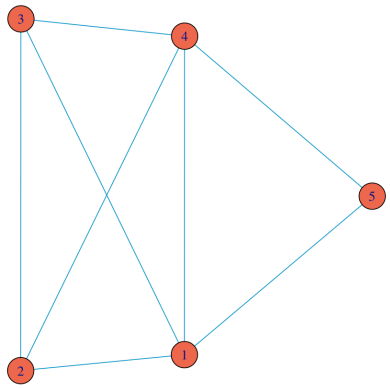


Figure 1: A toy-network featuring cliques

$k=$	1	2	3	4
Freq.	5	8	5	1

Table 1: Different possible k-cliques in the network of Figure 1 and their frequency.

Naturally, $k=1$ refers to the number of vertices in the network and $k=2$ refers to the number of edges. There are five sets of 3 vertices that are all connected to each other and only one set of vertices that form a complete sub-graph. **Maximal Clique** refers to the clique that is not subset of a larger clique. For instance, in the Figure 1 network, the 3 - clique of $\{3,4,2\}$ is **not** a maximal clique, as it is a subset of the 4 - clique of nodes $\{1,2,3,4\}$. The latter, however, is a maximal clique. In fact, the 3 - clique of nodes $\{1,4,5\}$ is also a maximal clique, as it is part of no larger clique. However trivial it may seem to detect the maximal clique in such a small network, it is by no means a straightforward task in larger networks. The Bron & Kerbosch (1973) algorithm for detecting all cliques in a graph has been proposed. A similar, but fundamentally different, definition is that of **maximum clique**, which is the clique with the greatest number of vertices. In Figure 1, the maximum clique is that of nodes $\{1,2,3,4\}$.

3.2 Special Cases of Networks

Networks can take any possible form, since they represent real-life situations and systems. However, there are some types of networks which are frequently encountered. Firstly, the concept of vertex degree

must be defined: Degree is the total number of connection for a vertex/node.

3.2.1 Some Special Cases

- Complete Network: Each node/vertex is connected to every other. Alternatively, every node/vertex has the same degree and equal to $|V| - 1$.
- Ring Network: Every node/vertex is connected only to the two adjacent ones, thus forming a ring. Similar to complete network, every vertex/node has the same degree, but now equal to 2.
- Tree Network: It is defined as a connected graph with no cycles, by Kolaczyk & Csárdi (2014). The vertex/node from which there is a directed path to every other vertex is defined as the root. For a vertex lower than the root, the next vertices are called descendants and the previous ones are called ancestors. The immediately next are called children and the immediately previous is called the parent.
- Star Network: All nodes/vertices are connected to one, but not to each other. This means that one vertex has degree equal to $|V| - 1$ and all others have a degree of 1.

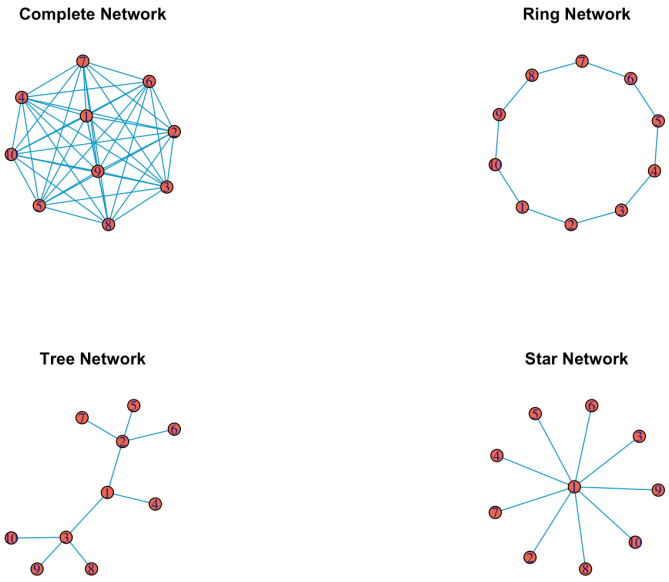


Figure 2: Visualization of the special network cases.

An example selection of the special types of networks is presented at Figure 2, all featuring 10 vertices.

3.2.2 Bipartite Networks

Bipartite are special cases of networks, whose vertex/node set V can be separated into two subsets of vertex sets V_1 and V_2, which are disjoint, i.e. $V_1 \cap V_2 = \varnothing$. Then, every edge/link in the network connects a vertex $v_i \in V_1$ to a vertex $v_j \in V_2$, and no vertices in the **same** subset are connected to each other.
Examples for bipartite networks vary, including the following:

- A network portraying actors as vertices of V_1 and movies as vertices of V_2 and edges symbolize participation of an actor in a movie. Kolaczyk & Csárdi (2014)
- A network where vertices of V_1 depict certain species and vertices of V_2 depict areas, having edges symbolize sightings of a species in a specific area.
- A network portraying students as vertices of V_1 and university courses as vertices of V_2, edges symbolizing enrollment to a course.
- A network whose vertices of V_1 depict merchants, vertices of V_2 depict banks and edges symbolizing loan issuing.

The last example is visualized in the following Figure 3.

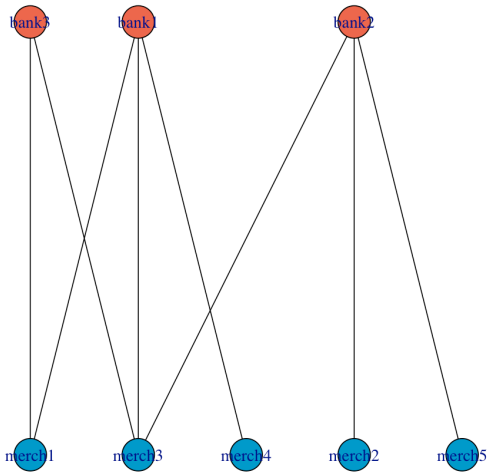


Figure 3: A simple bipartite network.

Naturally, bipartite networks can simultaneously be either directed, or weighted, or both.

3.3 Network Descriptive Analytics

The structure and properties of networks and their vertices are described and summarized by a series of descriptive statistics. While network density, clustering coefficient and average path length regard the whole network, vertex degree and centrality measures refer to individual nodes.

3.3.1 Vertex Degree and Strength

The degree of a vertex is the number of its connections. Accordingly, for weighted networks, an alternative statistic is defined, the vertex strength. The latter is defined as the sum of all weights of the vertices connection. Mathematically:

$$\text{degree}_i = \sum_{j \in V \setminus i} A_{i,j} \qquad\qquad \text{strength}_i = \sum_{j \in V \setminus i} w_{i,j}$$

These could be used as an indication of the importance of a node in a social network. Consider the case of a weighted network of social interactions in a small town within a year. Vertices are people, edges are social interactions and the weights are the time spent together. A person with higher strength could accelerate the spread of misinformation, more than someone with lower strength.
In directed networks, in-degree and out-degree of a vertex i can be defined as number of connections inward to i and number of connections outward from i respectively.

3.3.2 Network Density

Network density refers to the proportion of realized, or observed, edges in a network over the maximum possible number of edges. Given that any (undirected) network can be adequately described through an adjacency matrix, like the following:

$$A = \begin{bmatrix} 0 & A_{1,2} & A_{1,3} & \dots & A_{1,|V|} \\ A_{2,1} & 0 & A_{2,3} & \dots & A_{2,|V|} \\ A_{3,1} & A_{3,2} & 0 & \dots & A_{3,|V|} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{|V|,1} & A_{|V|,2} & A_{|V|,3} & \dots & 0 \end{bmatrix}$$

which is a symmetrical matrix in the case of undirected networks and asymmetric in case of directed (where $y_{i,j} \neq y_{j,i}$), of dimensions $|V| \times |V|$, where $|V|$ is the number of vertices in the network, with zeros in the diagonal. Thus, the maximum number of edges would be if $A_{i,j} = 1, \forall i, j$, meaning every node connected to every other. Then, the network would portray $|E|_{max} = \frac{|V| \cdot (|V|-1)}{2}$ edges. In conclusion, the network density is defined as:

$$density(G) = \frac{|E_G|}{|V_G| \cdot (|V_G| - 1)/2}$$

In comparison to the descriptive statistics vertex degree (or strength), which correspond to individual vertices, network density describes the entire network. Intuitively, for the simple network of Figure 1 the network density would be calculated equal to 0.8.

3.3.3 Vertex Centrality Measures

Briefly mentioned before, a goal in social network analysis is to determine the influential actors. Examples for this cause vary, including the following:

1. Epidemiological Networks monitoring the spread of HIV, where the most influential persons are those with many contacts.

2. Sport Networks, consider football games, where the most influential player is the one who is most fed the ball.

3. Professional Networks, where the most influential professional is the one who cooperates with most others.

Alternatively, a vertex may be considered influential if it is central in the network, not necessarily having the most connections/edges. Consequently, vertex centrality measures are defined.

- Closeness Centrality:

To define closeness centrality, firstly the idea of **geodesic distance** should be introduced. Geodesic distance, $d(x_i, x_j), i, j = 1, \ldots, |V|$, refers to the shortest path of steps from one node to another. In the case of the network of Figure 1, the geodesic distance between nodes 5 and 3 is: $d(5, 3) = 2$. A relevant quantity is **network diameter**, which is defined as the largest geodesic distance in the network. Intuitively, the shortest path between the most distant nodes. Therefore, closeness centrality is defined as:

$$c_{cl}(v_i) = \frac{1}{\sum_{j \in V \setminus i} d(v_i, v_j)}, i, j = 1, \ldots, |V|$$

This means, that a node is considered "central", when it has the lowest sum of (geodesic) distances to all the other nodes. Contrary to the vertex degree statistic, for a node to be considered central, it does not necessarily has to have the most connections, rather be the closest to every other node in the network.

- Betweenness Centrality:

Betweenness centrality statistic aims to quantify how much a vertex is located between other vertices. It builds on the notion of geodesic path, similar to the closeness centrality, and is defined as:

$$c_b(v) = \sum_{s \neq t \in V \setminus v} \frac{\sigma(s, t|v)}{\sigma(s, t)}$$

The numerator quantity $\sigma(s, t|v)$ denotes the total number of geodesic (i.e. shortest) paths between various nodes $s \neq t$ in the set $V \setminus v$ that pass through v. The denominator quantity $\sigma(s, t)$ denotes the total number of geodesic paths between nodes $s \neq t$ in $V \setminus v$. The set V contains all vertices of a network, which namely in Figure 1 is: $\{1, 2, 3, 4, 5\}$.

- PageRank Centrality:

PageRank is an algorithm developed and used by Google to assign a centrality score P_i to every webpage. It is based on the idea that important websites should have inbound links from many other important, but link parsimonious, websites. Naturally, therefore, it is defined for directed networks but can also be implemented in undirected, under the notion that every edge is mutual. Considering the adjacency matrix $A = \{a_{i,j}\}, i, j = 1, \ldots, |V|$, PageRank centrality is defined as:

$$P_i = \beta + \alpha \cdot \sum_{k \in V \setminus i} \frac{a_{k,i}}{d_k} \cdot P_k$$

where β and α are constants and d_k is the out-degree of node k (number of outward connections of node k in a directed graph). This constitutes a system of $|V|$ linear equations with $|V|$ unknown P_i scores, whose solution could be iterative, starting from initial values $P_i = \frac{1}{|V|}, i = 1, \ldots, |V|$.

3.3.4 Clustering Coefficient

A straightforward clustering coefficient mentioned by Kolaczyk & Csárdi (2014) is based on the ratio of triangles over triples in a network:

$$c_{|T}(G) = \frac{3 \cdot \tau_{\triangle}(G)}{\tau_3(G)}$$

where $\tau_{\triangle}(G)$ denotes the number of triangles and $\tau_3(G)$ denotes the number of triples in a network G. A triangle is defined as a set of three vertices/nodes $\{v_1, v_2, v_3\}$ which form the edges

$$\{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_1\}\}$$

and triple is a set of three vertices/nodes $\{v_1, v_2, v_3\}$ that form the edges $\{\{v_1, v_2\}, \{v_2, v_3\}\}$.
A naive approach in calculating the quantity $\tau_{\triangle}(G)$ would be to check all the possible triads of vertices $\{v_1, v_2, v_3\}$ and investigate if they form the edges $\{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_1\}\}$. This approach, however, would involve investigating all $\binom{|V|}{3} = \frac{|V|!}{3! \cdot (|V|-3)!}$ combinations of vertices, which simplifies to $\frac{|V| \cdot (|V|-1) \cdot (|V|-2)}{6}$, where once again $|V|$ is the total number of vertices. For relatively small networks, e.g. $|V| = 8$, this process would involve inspecting 56 triads of vertices.

3.3.5 Average Path Length

Average Path Length is directly based on the notion of geodesic distances, and quantifies the expected number of steps between two nodes. It is defined as:

$$L_G = \frac{1}{|V| \cdot (|V| - 1)} \cdot \sum_{i \neq j \in V} d(v_i, v_j)$$

It is directly related to the concept of **network diameter**, as it aims to capture how wide the network spans. A practical example for Average Path Length would be how many social contacts we expect, before a infectious disease is transmitted from an infected person to a susceptible one.

3.4 Visualization Layouts

Network visualization is a vital, but far from straightforward task. All methods aim to assign a spot for every vertex/node in a two-dimensional plane and represent edges by straight lines connecting the spots. The challenge is that there is neither a unique nor an inherently correct way to place vertices in a two-dimensional plane, such as when visualizing bi-variate relationships, e.g. income - expenditure or exposure to sunlight - growth in cm.
The aim of this section is to discuss two predominant algorithms of **force-directed graph drawing**, which exploit concepts from physics, such as Hooke's Law for spring forces and Coulomb's Law for electrically charged particles' repulsive forces, to produce aesthetically pleasing and accurate network representations. These are applicable for undirected and either weighted or unweighted networks.

3.4.1 Kamada - Kawai Algorithm

The Kamada & Kawai (1989) algorithm considers the layout of vertices in the two - dimensional plane to be a dynamical system, where edges connecting the vertices are springs. Hence, the optimal layout is

the one that minimizes the total spring energy of the system. Firstly, it is of use to review some physics concepts that apply in the Kamada - Kawai algorithm. The potential energy of a spring is given by the formula $U_s = \frac{1}{2} \cdot \kappa \cdot (\Delta_x)^2$, where κ is the spring constant (or strength) and $\Delta_x = (l_0 - x)$ is the spring displacement, namely the difference between the length of the spring in its relaxed state/equilibrium (l_o) and its length when stretched or compressed (x). It is apparent that compression or stretching of a spring by equal lengths results in the same potential energy.
Since we consider edges connecting nodes in the two - dimensional plane to be springs, the total potential energy formula summing all spring compression or stretching must be adapted accordingly:

$$U_{total} = \sum_{i=1}^{|V|-1} \sum_{j=i+1}^{|V|} \frac{1}{2} \cdot k_{i,j} \cdot (|V_i - V_j| - l_{i,j})^2$$

Naturally, the constant of a spring would depend on its material and manufacturing and for the case of networks will be defined as $k_{i,j} = \frac{K}{d_{i,j}^2}$, where K is a constant. Quantity $|V_i - V_j|$ denotes the Euclidean distance between the points of vertices i and j in the two - dimensional plane. Springs equilibrium length, $l_{i,j}$, is defined as: $l_{i,j} = L \cdot d_{i,j}$, where L corresponds to the preferred distance of points in the two - dimensional plane. Lastly, $d_{i,j}$ is the geodesic distance (shortest path) between vertices i and j in the network.
By defining

$$(x_1, y_1), (x_2, y_2), \ldots, (x_{|V|}, y_{|V|})$$

to be the (unknown) positions of vertices $V_1, V_2, \ldots, V_{|V|}$ in the two - dimensional plane. Substituting and expanding the square of differences, it results in :

$$U_{total}(\mathbf{x},\mathbf{y}) = \sum_{i \neq j} \frac{1}{2} \cdot k_{i,j} \cdot ((x_i - x_j)^2 - (y_i - y_j)^2 + 2 \cdot l_{i,j} \cdot \sqrt{(x_i - x_j)^2 - (y_i - y_j)^2} + l_{i,j}^2)$$

The aim is to find the two - dimensional coordinates that minimize $U_{total}(\mathbf{x},\mathbf{y})$. The achievement of a global minimum is not guaranteed, therefore a local minimum will suffice. The condition for a local minimum to exist is:

$$\frac{\partial U_{total}(\mathbf{x},\mathbf{y})}{\partial x_m} = \frac{\partial U_{total}(\mathbf{x},\mathbf{y})}{\partial y_m} = 0$$

The layout of vertices in the two - dimensional plane with coordinates $(x_i, y_i), i = 1, \ldots, |V|$ that satisfy this equation corresponds to a dynamic system where all forces of springs all balanced.

The following set of $2 \cdot |V|$ non - linear equations is produced by applying partial derivatives:

$$\frac{\partial U_{total}(\mathbf{x},\mathbf{y})}{\partial x_m} = \sum_{i \neq m} k_{m,i} \cdot ((x_m - x_i) + \frac{l_{i,m} \cdot (x_m - x_i)}{\sqrt{(x_i - x_j)^2 - (y_i - y_j)^2}})$$

and

$$\frac{\partial U_{total}(\mathbf{x},\mathbf{y})}{\partial y_m} = \sum_{i \neq m} k_{m,i} \cdot ((y_m - y_i) + \frac{l_{i,m} \cdot (y_m - y_i)}{\sqrt{(x_i - x_j)^2 - (y_i - y_j)^2}})$$

Then, it is subject of numerical methods to obtain the local minimum.

3.4.2 Fruchterman - Reingold Algorithm

Fruchterman & Reingold (1991) algorithm , similarly to the Kamada - Kawai algorithm, considers an imaginary physical system and works towards producing aesthetically pleasing but accurate visualizations through simple simulations. In contradiction to the Kamada - Kawai approach, the Fruchterman - Reingold considers both attractive (f_a) and repulsive (f_r) forces to be exercised between vertices. A physical reference would be to envision nodes as celestial bodies (e.g. planets) exercising both attractive and repulsive forces to each other. It performs on two basic principles:

- Vertices/nodes that are neighbors should be drawn close to one another.
- Vertices/nodes should **not** be drawn too close to one another.

The forces considered to be acting upon the vertices are the following:

$$f_a = \frac{d^2}{k}, f_r = -\frac{k^2}{d}$$

where k is defined as $k = C \cdot \sqrt{\frac{area}{|V|}}$ and denotes the optimal distance between vertices, where C is a constant optimally determined through experimentation. Term area refers to the area of the plot. Distance d is the Euclidean distance between the vertices in the two - dimensional plane, as defined in Kamada - Kawai algorithm.

Practically, the algorithm iteratively adjusts the position of nodes in the two - dimensional plane, until the forces enacted upon them bring the system into an equilibrium. Contrary to the Kamada - Kawai algorithm, in the Fruchterman - Reingold approach, borders must be set as line segments in the plot, so that the repulsive and attractive forces simulated upon the vertices do not push them out of bounds. Lastly, an issue both approaches faced was that, due to the physical forces imagined to be acting upon vertices, unconnected vertices have no forces pulling them close to the other.

3.4.3 Visualization Enrichment

As presented thoroughly by Kolaczyk & Csárdi (2014), there are various ways to visually enhance network layouts, and at the same time incorporate additional useful information. In particular, these refer to modifications that can be made after the layout has been established through the previously discussed methods of Kamada - Kawai or Fruchterman - Reingold, or any other.

The three main attributes that can be influenced in network visualization layouts are: **vertex size, color and shape** as well as **edge width and color**. It merely depends on what information is to be depicted. The vertex size could be dependent on a numeric attribute of the nodes. This attribute can be inherent to the network, for example vertex degree or strength, or independent. Similarly, vertex shape and color could be manipulated to encompass information from two categorical variables that regard the nodes. On the other hand, edge color could be manipulated to highlight connections between nodes of different subgroups (highlighted either by different vertex shapes or colors). Edge width typically encompasses information about the edge weights on a weighted network.

At this point, it would be useful to employ a practical example to illustrate these manipulations. Consider an imaginary network depicting financial transactions between major European and North American

financial institutions. Financial institutions can be partitioned into commercial banks, investment banks and central banks and a feature of interest is their vertex strength (total transactions).

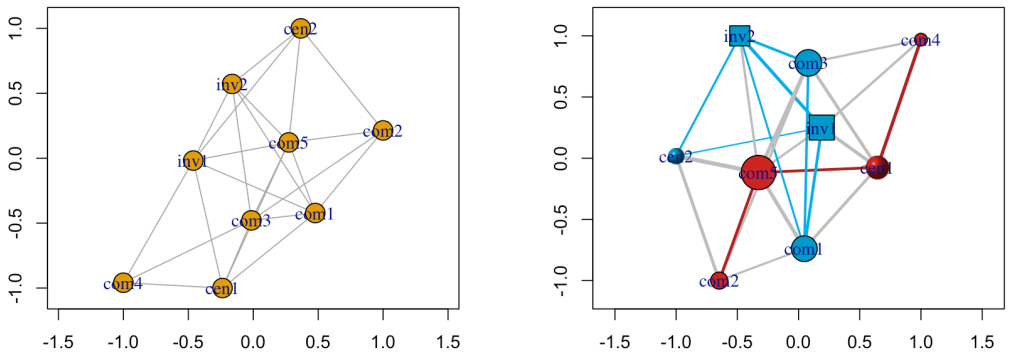


Figure 4: A comparison of enriched versus generic visual.

Figure 4 contains two plots of the same network. The plot on the left is the visualization of the example network through the Kamada - Kawai layout, without any visual enhancements. The plot on the right depicts the same network, again through the Kamada - Kawai layout, but with visual decorations. In particular, vertex color represents the continent, blue for Europe and red for North America. Vertex shape corresponds to the financial institution type, square for investment banks, circle for commercial banks and sphere for central banks. Vertex size is set according to vertex strength. Edge width represents the edge weights, scaled, and edge color aims to differentiate inter - and intra - continental transactions. It is now apparent, how much useful information could have been missed, if appropriate layout visual enrichment was not utilized.

4 Some Statistical Models

4.1 Elements of Bayesian Statistics

Due to the complexity of some of the models that will be discussed in this chapter, the Bayesian approach to estimation constitutes the most effective way forward. Bayesian statistics allows for a more flexible way of statistical inference, as parameters are not considered unknown quantities for estimation, rather random variables with probability distributions. Moreover, it facilitates for the integration of previous information or personal beliefs in the estimation process through the prior distribution.

The entire theory is based on Bayes' Rule:

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(B|A) \cdot \mathbb{P}(A)}{\mathbb{P}(B)}$$

and the Law of Total Probability:

$$\mathbb{P}(B) = \sum_{A_i \in \Omega} \mathbb{P}(B|A_i) \cdot \mathbb{P}(A_i)$$

where Ω denotes the sample space and A_i constitute a partition of the latter, as in $\mathbb{P}(\bigcup_{A_i \in \Omega} A_i) = \Omega$.

Through classical statistics, the standard approach is to construct a likelihood function through a probability model (distribution), denoted as $f(\mathbf{x}|\theta)$, which expresses the joint probability of observing the data $\mathbf{x}$ given the model parameters θ. The goal is to specify the parameter vector θ, that maximizes the probability of observing the data at hand, hence the term **Maximum Likelihood Estimation**. Depending on the complexity of the likelihood function, this can be straightforward or require numerical methods such as Fisher's Scoring or Newton - Raphson.

The Bayesian approach combines personal beliefs in the form of the prior (or a - priori) distribution, $\pi(\theta)$, along with data, in the likelihood function, $f(\mathbf{x}|\theta)$, through Bayes' Rule which results in the posterior (or a - posteriori) distribution of the parameter vector:

$$f(\theta|\mathbf{x}) = \frac{f(\mathbf{x}|\theta) \cdot \pi(\theta)}{f(\mathbf{x})}$$

Then, the Law of Total Probability is utilized to express the normalizing constant $f(\mathbf{x}) = \int_\theta f(\mathbf{x}|\theta) \cdot \pi(\theta)$. The denominator (normalizing constant), ensures that probabilities $f(\theta|\mathbf{x})$ integrate to 1. In most practical cases, numerical computation of the latter is unfeasible, thus hindering the retrieval of the posterior distribution impossible. In such cases, **Markov Chain Monte Carlo** (MCMC) methods are employed, to obtain a sample of values for θ from the posterior distribution. Two established MCMC methods are Metropolis - Hastings and Gibbs Sampling, which are not exclusively used in Bayesian Statistics, but any setting which requires the study of complex probability distributions.

The posterior distribution is often alternatively expressed as $f(\theta|\mathbf{x}) \propto f(\mathbf{x}|\theta) \times \pi(\theta)$, or:

$$posterior \propto prior \times likelihood$$

because the normalizing constant, $f(\mathbf{x})$, contains no information about the parameters θ. In empirical cases where either there is no prior information on parameters, or the researcher aims to avoid influencing the analysis with subjective beliefs, **uninformative** prior distributions are used. For example, a uniform of $Beta(1,1)$ distribution.

4.1.1 Metropolis - Hastings Algorithm

In MCMC, the term "Monte - Carlo" refers to the generation of random values and the "Markov - Chain" process of generating each value based on the previous, as per the Markov Property:

$$\mathbb{P}(X_{t+1}|X_t, X_{t-1}, \ldots, X_1) = \mathbb{P}(X_{t+1}|X_t)$$

The Metropolis - Hastings Algorithm, as presented by Robert & Casella (2004), is a way to obtain a random sample from the posterior distribution without the need to compute the normalizing constant. The algorithm proceeds as following:

- Assign an initial value to the parameter, $\theta^{(0)}$.
- Generate a random value $\theta^{(1)}$ from the proposal distribution $q(\theta^{(0)})$

For indices $i = 0, 1, \ldots, B$:

- Generate a random value $\theta^{(i+1)}$ from the proposal distribution $q(\theta^{(i)})$
- Accept the new value, $\theta^{(i+1)}$, with probability:

$$r = min(1, \frac{f(\theta^{(i+1)}|\mathbf{x})}{f(\theta^{(i)}|\mathbf{x})})$$

- Repeat a sufficient number of iterations, e.g. $B = 10000$

The result is a random sample $(\theta^{(0)}, \theta^{(1)}, \ldots, \theta^{(B)})$ of parameter values from the posterior distribution.

In the acceptance stage, quantity r is defined as the **Hastings Ratio**. The algorithm will, by definition, accept the newly proposed value, $\theta^{(i+1)}$, if it achieves a higher posterior probability than the previous, $\theta^{(i)}$. Otherwise, it draws a value u from a uniform distribution $U(0,1)$ and compares it to the Hastings Ratio, r. If $u < r$, it accepts the newly proposed value $\theta^{(i+1)}$. In the Hastings Ratio, there is no issue if the normalizing constant is intractable, as it cancels out.

An apparent issue with the Metropolis - Hastings Algorithm is that the sample is dependent on the initial value, $\theta^{(0)}$. Therefore, an initial batch of values called the **burn - in** is discarded until the Markov - Chain stabilizes, the size of which becomes irrelevant, as the total number of algorithm iterations (denoted B) increases. Chain stabilization can be monitored in the **trace - plot**, which captures the sampling process.

4.1.2 Gibbs Sampling

The Gibbs Sampling (also referred to as alternating conditional sampling by Gelman et al. (1995)) is a MCMC method, with similar approach to the Metropolis - Hastings algorithm. However, it provides for a more versatile way to obtain a random sample from a complex **multivariate** (posterior) distribution.

One key point of difference between these MCMC approaches it that the Gibbs Sampling involves no proposal and acceptance step, but samples straight from the conditional distributions.

Consider a d - dimensional parameter vector $\theta = (\theta_j), j = 1, \ldots, d$. The notation θ_{-j}, refers to all ($d - 1$ in total) parameters, apart from θ_j. For instance, if $d = 3$ and, thus, the parameter vector is $(\theta_1, \theta_2, \theta_3)$, then θ_{-1} would concern (θ_2, θ_3). Once again, Gibbs Sampling is applicable also in other settings, except the Bayesian. However, in the Bayesian context the algorithm is as described below, first with two parameters, and next the general case.

In the case of two parameters:

- Assign initial values to the parameters: $(\theta_1^{(0)}, \theta_1^{(0)})$

For indices $t = 1, \ldots, B$

- Obtain sample value $\theta_1^{(t)}$ from conditional posterior distribution $f(\theta_1^{(t-1)}|\mathbf{y}, \theta_2^{(t-1)})$. The Markov Chain is at state $(\theta_1^{(t)}, \theta_2^{(t-1)})$.
- Obtain sample value $\theta_2^{(t)}$ from conditional posterior distribution $f(\theta_2^{(t-1)}|\mathbf{y}, \theta_1^{(t)})$. The Markov Chain is now at state $(\theta_1^{(t)}, \theta_2^{(t)})$.
- Repeat a sufficient number of iterations, say $B = 10000$.

The result is a random sample $(\theta_1^{(t)}, \theta_2^{(t)}), t = 1, \ldots, B$, from the **full** posterior distribution.

In the general case of d parameters $(\theta_1, \theta_2, \ldots, \theta_d)$, the idea is the same:

- Assign initial values $(\theta_1^{(0)}, \theta_2^{(0)}, \ldots, \theta_d^{(0)})$.

For indices $t = 1, \ldots, B$:

- Obtain a sample value $\theta_1^{(t)}$ from conditional posterior distribution $f(\theta_1^{(t-1)}|\mathbf{y}, \theta_2^{t-1}, \ldots, \theta_{d-1}^{t-1}, \theta_d^{t-1})$
- Obtain a sample value $\theta_2^{(t)}$ from conditional posterior distribution $f(\theta_2^{(t-1)}|\mathbf{y}, \theta_1^{t}, \ldots, \theta_{d-1}^{t-1}, \theta_d^{t-1})$

$$\vdots$$

- Obtain a sample value $\theta_d^{(t)}$ from conditional posterior distribution $f(\theta_d^{(t-1)}|\mathbf{y}, \theta_1^{t}, \theta_2^{t}, \ldots, \theta_{d-2}^{t}, \theta_{d-1}^{t})$
- Repeat a sufficient number of iterations.

The result would be a random sample from the d - dimensional posterior distribution, $(\theta_1^t, \theta_2^t, \ldots, \theta_d^t)$ for large number of algorithm iterations $t = 1, \ldots, B$.

Similar to the Metropolis - Hastings Algorithm, the Gibbs Sampling Algorithm is also heavily dependent on the initial values. Therefore, an initial **burn - in** batch of sample vectors would need to be discarded, until the point where the Markov - Chain stabilizes.

4.2 Erdős–Rényi Model

The Erdős–Rényi is the simplest statistical model that can describe a network, comprehensively presented in the survey by Goldenberg et al. (2010). It is denoted as $G(n, p)$, where $n = |V|$ is the number of vertices. It is based on the concept that any two vertices v_i and v_j of the vertex set V have the same probability of forming a link, equal to p, and assumes that creation of edges is independent. It can support either directed or undirected networks, but not weighted.

The likelihood function of the model is constructed based on the Binomial Distribution, specifically:

$$\mathbb{P}[Y = y] = \prod_{i,j} p^{y_{i,j}} \cdot (1-p)^{1-y_{i,j}}$$

Naturally, quantities $y_{i,j}$ are equal to 1 if there is a link between nodes v_i and v_j and 0 if there is not. In the case of directed networks, the product involves all $i \neq j$ vertices, because $y_{i,j} \neq y_{j,i}$. However, in the case of undirected networks, the product is based on $i < j$ terms. The goal is to estimate the probability p of any two vertices forming a link, which can be achieved through Maximum Likelihood Estimation.

4.2.1 MLE Estimation

The estimation of p will be based on the log - likelihood function, as:

$$\arg\max_p L(Y|p) = \arg\max_p \ell(Y|p)$$

Below is presented the steps to obtain the maximum likelihood estimate of link probability p:

$$\begin{aligned}
\ell(Y|p) = \log L(Y|p) &= \log(\prod_{i,j} p^{y_{i,j}} \cdot (1-p)^{1-y_{i,j}}) \\
&= \sum_{i,j} \log(p^{y_{i,j}} \cdot (1-p)^{1-y_{i,j}}) \\
&= \log(p) \cdot \sum_{i,j} y_{i,j} + \log(1-p) \cdot \sum_{i,j} (1-y_{i,j})
\end{aligned}$$

The next step is to derive the score of the log - likelihood function and equalize it to zero:

$$\begin{aligned}
U(p) = 0 &\Rightarrow \frac{dl(Y|p)}{dp} = 0 \\
&\Rightarrow \frac{\sum_{i,j} y_{i,j}}{p} - \frac{\sum_{i,j}(1-y_{i,j})}{1-p} = 0 \\
&\Rightarrow \frac{\sum_{i,j}(y_{i,j}) - p \cdot \sum_{i,j}(y_{i,j}) - p \cdot \sum_{i,j}(1) + p \cdot \sum_{i,j}(y_{i,j})}{p \cdot (p-1)} = 0 \\
&\Rightarrow \hat{p}_{MLE} = \frac{\sum_{i,j}(y_{i,j})}{\sum_{i,j}(1)}
\end{aligned}$$

The term $\sum_{i,j}(y_{i,j})$ expresses the total number of connections. Similarly, term $\sum_{i,j}(1)$ expresses total number of **possible** connections, which is a quantity that differs between directed and undirected networks. In undirected networks this is equal to the combinations $\binom{|V|}{2} = \frac{|V|!}{2!\cdot(|V|-1)!} = \frac{|V|\cdot(|V|-1)}{2}$. However, in undirected networks where $y_{i,j}$ (link from node v_i to v_j) differs from $y_{j,i}$ (link from node v_j to v_i), this sum is equal to $|V| \cdot (|V| - 1)$, as the adjacency matrix with zeros in the diagonal is not symmetrical.

4.2.2 Properties of Erdős–Rényi

There are two interesting properties arising from the simplistic nature of the Erdős–Rényi statistical model. Firstly, the total expected number of connections in an undirected network is $\binom{|V|}{2} \cdot \hat{p}_{MLE}$ and in a directed network $|V| \cdot (|V| - 1) \cdot \hat{p}_{MLE}$.

As presented by Garlaschelli (2009), under the Erdős–Rényi model, the degree distribution for every node is $Binom(|V| - 1, p)$ with probability density function:

$$\mathbb{P}(degree_i = k) = \binom{|V| - 1}{k} \cdot p^k \cdot (1 - p)^{(|V|-1)-k}$$

Based on this, we can infer that for every node in the network, the expected degree (total number of connections) is $(|V| - 1) \cdot \hat{p}_{MLE}$.

Having estimated p, the model can be used to produce sample random graphs, which by definition do not display a specific structure. For example, the clustering coefficient in such a network would be small, as triangles serve as an indication of structure. Although straightforward and intuitive, this model lacks applicability in any real - world network setting, as few if any observed networks present such simplistic structure. Nevertheless, it serves as the building block for more complex models that aim to better capture the networks' structure and characteristics.

4.3 The p1 model

The p_1 model builds on the bases of the Erdős–Rényi model, for unweighted and directed networks. Its construction allows for the capture and quantification of **reciprocity**. Reciprocity refers to the effect of observing both the edge $v_i \rightarrow v_j$ and the edge $v_i \leftarrow v_j$, i.e. the tendency of edges to be mutual. Contrary to the Erdős–Rényi which had one parameter (p), the p_1 model has $2 + 2 \cdot |V|$ parameters. The general form of the probability parametrization is:

$$logit(p_{i,j}) = \mu + a_i + b_j + \rho \cdot Y_{j,i} \Leftrightarrow p_{i,j} = \frac{e^{\mu + a_i + b_j + \rho \cdot Y_{j,i}}}{1 + e^{\mu + a_i + b_j + \rho \cdot Y_{j,i}}}$$

Parameters $a_i, i = 1, \ldots, |V|$ express nodes' v_i tendency to produce outgoing edges, $b_j, j = 1, \ldots, |V|$ express node's v_j tendency to receive incoming edges, parameter ρ encompasses the network - wide effect of mutuality or reciprocation and, lastly, μ is the baseline rate of edge creation. Naturally, if $a_i = b_j = \rho = 0, \forall i, j \in V$, then the p_1 model becomes the Erdős–Rényi. In the original paper by Holland & Leinhardt (1981), it is designated that the following restrictions must be set: $a_+ = \sum_{i=1}^{|V|} a_i = 0$ and $b_+ = \sum_{j=1}^{|V|} b_j = 0$, in order for parameters $a_i, i = 1, \ldots, |V|$ and $b_j, j = 1, \ldots, |V|$ to be identifiable,

or be uniquely estimable from the observed network. Therefore, the total number of parameters to be estimated is not $2 + 2 \cdot |V|$, but rather $2 \cdot |V|$.

The construction of the models' likelihood function is the following:

$$\begin{aligned} L(Y|\mu, a_i, b_j, \rho) &= \prod_{i \neq j} (p_{i,j})^{y_{i,j}} \cdot (1 - p_{i,j})^{1 - y_{i,j}} \\ &= \prod_{i \neq j} \left(\frac{e^{\mu + a_i + b_j + \rho \cdot Y_{j,i}}}{1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}} \right)^{y_{i,j}} \cdot \left(\frac{1}{1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}} \right)^{1 - y_{i,j}} \end{aligned}$$

By applying the natural logarithm to both sides, we can work with the log - likelihood function:

$$\begin{aligned} \ell(Y|\mu, a_i, b_j, \rho) &= \log \prod_{i \neq j} \left(\frac{e^{\mu + a_i + b_j + \rho \cdot Y_{j,i}}}{1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}} \right)^{y_{i,j}} \cdot \left(\frac{1}{1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}} \right)^{1 - y_{i,j}} \\ &= \sum_{i \neq j} y_{i,j} \cdot \log \frac{e^{\mu + a_i + b_j + \rho \cdot Y_{j,i}}}{1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}} + \sum_{i \neq j} (1 - y_{i,j}) \cdot \log \frac{1}{1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}} \\ &= \sum_{i \neq j} y_{i,j} \cdot (\mu + a_i + b_j + \rho \cdot y_{j,i}) - \sum_{i \neq j} y_{i,j} \cdot \log(1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}) \\ &- \sum_{i \neq j} \log(1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}) + \sum_{i \neq j} y_{i,j} \cdot \log(1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}) \\ &= \sum_{i \neq j} y_{i,j} \cdot (\mu + a_i + b_j + \rho \cdot y_{j,i}) - \sum_{i \neq j} \log(1 + e^{\mu + a_i + b_j + \rho \cdot y_{j,i}}) \end{aligned}$$

In directed networks, where the adjacency matrix is asymmetrical, the sum involving all $i \neq j$ terms coincides with the double sum $\sum_{i=1}^{|V|} \sum_{j=1}^{|V|}$, as terms $y_{i,i} = 0, \forall i \in V$. Thus, by focusing only on the first term of the log - likelihood, the following form is obtained:

$$\begin{aligned} &\ell(Y|\mu, a_i, b_j, \rho) \propto \mu \cdot \sum_{i=1}^{|V|} \sum_{j=1}^{|V|} y_{i,j} + \sum_{i=1}^{|V|} a_i \sum_{j=1}^{|V|} y_{i,j} + \sum_{j=1}^{|V|} b_j \sum_{i=1}^{|V|} y_{i,j} + \rho \cdot \sum_{i=1}^{|V|} \sum_{j=1}^{|V|} y_{i,j} \cdot y_{j,i} \Rightarrow \\ &\ell(Y|\mu, a_i, b_j, \rho) \propto \mu \cdot y_{+,+} + \sum_{i=1}^{|V|} (a_i \cdot y_{i,+}) + \sum_{j=1}^{|V|} (b_j \cdot y_{+,j}) + \rho \cdot n_{mutual} \end{aligned}$$

The quantity $\sum_{i=1}^{|V|} \sum_{j=1}^{|V|} y_{i,j} \cdot y_{j,i}$ is denoted as n_{mutual}, as it corresponds to the number or mutual (or reciprocal) edges in the network, since $y_{i,j} \cdot y_{j,i} = 1$ only if both $v_i \rightarrow v_j$ and $v_i \leftarrow v_j$ edges exist. Of course, estimation of parameters through this likelihood function is not straightforward, but an iterative algorithm has been proposed in the original paper by Holland & Leinhardt (1981). An asset of the p_1 model is that it can capture **degree heterogeneity**, which refers to the phenomenon of varying degrees between vertices.

4.4 Network Block Model

The Network Block Model is another model that expands the basic Erdős–Rényi model for social networks. In particular, it is assumed that every vertex $v_i \in V$ belongs to one of Q **known** classes, denoted $C_1, \ldots, C_Q$. The fundamental idea behind the model is, that the probability of two vertices, v_i and v_j, forming a connection, $p_{i,j}$, depends on their classes: q_i and q_j respectively.

What the term **known** labels means, is that they are derived either from a supplementary variable, or from some community detection method presented later. For instance, consider a network of residents in a rural city. For each resident (vertex) we know their education level, namely: school, university, doctorate. It would be useful to predict the probability of two residents forming a friendship based on their education level. This would form a probability matrix of dimensions $Q \times Q$, like the one presented further.

If it can be argued that a network has vertices that belong to classes, but whose class membership is **unknown**, the Stochastic Block Model is of use. Nevertheless, in the case of known class membership, the probability of a node of class $q_i = q$ forming a connection with a node of class $q_j = r$ is denoted: $\pi_{q,r}$ and must be estimated.

4.4.1 MLE Estimation

The estimation of the probabilities π_{q_i,q_j} is achieved through Maximum Likelihood Estimation. The number of parameters to be estimated is $\binom{Q}{2} + Q = Q \cdot (Q+1)/2$, seeing as the parameters correspond to the following:

- The probability of two nodes within **the same** class forming a connection, for each of the Q classes.
- The probability of a node in every class forming a link with a node **from any other** class.

The models' likelihood function is constructed in a similar way as the Erdős–Rényi model, based on the Binomial Distribution:

$$L(Y) = \mathbb{P}(Y = y) = \prod_{i \neq j | q_i = q, q_j = r} (\pi_{q,r})^{y_{i,j}} \cdot (1 - \pi_{q,r})^{1 - y_{i,j}}$$

The next step is applying the natural logarithm to both sides and working with the log - likelihood:

$$\begin{aligned} \log \mathbb{P}(Y = y) &= \log \left(\prod_{i \neq j | q_i = q, q_j = r} (\pi_{q,r})^{y_{i,j}} \cdot (1 - \pi_{q,r})^{1 - y_{i,j}} \right) \\ &= \sum_{i \neq j | q_i = q, q_j = r} y_{i,j} \cdot \log(\pi_{q,r}) + \sum_{i \neq j | q_i = q, q_j = r} (1 - y_{i,j}) \cdot log(1 - \pi_{q,r}) \end{aligned}$$

To obtain the MLE estimate of $\pi_{q,r}$, the score function must be equalized to zero:

$$
\begin{aligned}
U(\pi_{q,r}) &= \frac{\partial \log \mathbb{P}(Y=y)}{\partial \pi_{q,r}} = 0 \\
&\Rightarrow \frac{\sum\limits_{i\neq j|q_i=q,q_j=r} y_{i,j}}{\pi_{q,r}} - \frac{\sum\limits_{i\neq j|q_i=q,q_j=r} (1-y_{i,j})}{1-\pi_{q,r}} = 0 \\
&\Rightarrow \frac{1}{\pi_{q,r}\cdot(1-\pi_{q,r})} \cdot \Bigg(\sum_{i\neq j|q_i=q,q_j=r} y_{i,j} - \pi_{q,r} \cdot \sum_{i\neq j|q_i=q,q_j=r} y_{i,j} \\
&\quad + \pi_{q,r} \cdot \sum_{i\neq j|q_i=q,q_j=r} (1) + \pi_{q,r} \cdot \sum_{i\neq j|q_i=q,q_j=r} y_{i,j} \Bigg) = 0
\end{aligned}
$$

By eliminating second and fourth terms, and since $\pi_{q,r} \in (0,1)$, the MLE of $\pi_{q,r}$ is:

$$
\hat{\pi_{q,r}}_{MLE} = \frac{\sum\limits_{i\neq j|q_i=q,q_j=r} y_{i,j}}{\sum\limits_{i\neq j|q_i=q,q_j=r} (1)} = \frac{e_{q,r}}{|V_q|\cdot|V_r|}
$$

Therefore, the Maximum Likelihood Estimate of the probability of link formation between any two nodes of groups q and r is the observed proportion of realized edges between nodes of these classes, over the total possible number of nodes between these classes. This would produce an link - formation probability matrix:

$$
\begin{bmatrix}
\hat{\pi_{1,1}} & \hat{\pi_{1,2}} & \cdots & \hat{\pi_{1,Q}} \\
\hat{\pi_{2,1}} & \hat{\pi_{2,2}} & \cdots & \hat{\pi_{2,Q}} \\
\vdots & \vdots & \ddots & \vdots \\
\hat{\pi_{Q,1}} & \hat{\pi_{Q,2}} & \cdots & \hat{\pi_{Q,Q}}
\end{bmatrix}
$$

which is symmetric in undirected networks and asymmetric in directed ones. A significant advantage of the Network Block Model is that it can capture the phenomenon of **homophily** in networks. Homophily is the tendency of nodes with similar characteristics, in this case classes, to from connections more frequently than with nodes of different characteristics.

4.5 Latent Distance Model

The Latent Distance Model belongs to the broader category of Latent Network Models, which are models that aim to explain the link generating mechanism of networks based on latent, i.e. unobserved, quantities. These models include the Stochastic Block Model and more. The Latent Distance Model, thoroughly presented by Kaur et al. (2023), considers the probability of a connection between two vertices $v_i, v_j \in V$ to be dependent on their relative position in a latent d - dimensional Euclidean space, $\mathbb{R}^d$, among other pair - specific covariates.

The positions of the vertices in the latent space, which are vectors of dimensions $d \times 1$, are denoted $z_i, i = 1, \ldots, |V|$ and usually refer to the two dimensional space $\mathbb{R}^2$. If available, any pair - specific covariates are symbolized $X_{i,j,k}$ where $i, j = 1, \ldots, |V|$ and $k = 1, \ldots, p$. In a practical scenario, these could be the difference in age of two network actors, or whether two students attend the same university.

The latent distance model can, inherently, capture **homophily** in the network, as nodes closer in the latent space present higher link probability.

Mathematically, it can be viewed as an extension of the fundamental Erdős–Rényi model, by parameterizing the link probability $p_{i,j} = \mathbb{P}(Y_{i,j} = 1 \mid z_i, z_j, X_{i,j,k})$ as:

$$logit(p_{i,j}) = \log\left(\frac{p_{i,j}}{1-p_{i,j}}\right) = \mu + \sum_{k=1}^{p} \beta_k \cdot X_{i,j,k} - \mid\mid z_i - z_j \mid\mid$$

The strictly increasing logit transformation, $logit(x) : (0,1) \to \mathbb{R}$, is used because the right part can take real values ($\mathbb{R}$), while probability is bound in the interval $(0,1)$, similar to its use in the p_1 model. By β_k are denoted the coefficients of the pair - specific covariates and by $\mid\mid \cdot \mid\mid$ is the Euclidean Distance, defined in a d - dimensional space as:

$$\mid\mid x - y \mid\mid = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2}$$

For instance, it would be useful to estimate the latent positions of the five vertices in the toy - network of Figure 1 in the two - dimensional space:

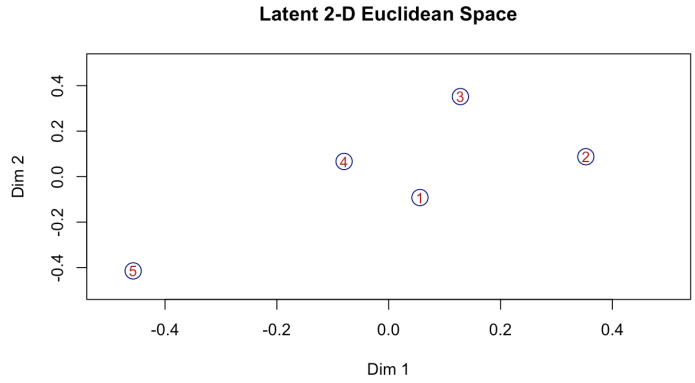


Figure 5: The latent positions of the vertices of Figure 1.

Naturally, the estimation of latent positions that correspond to the link structure of a network, also serve the purpose of visualization as they provide a methodological and not arbitrary way of placing vertices as points in a surface. In Figure 5, the nodes that formed the 4 - clique of Figure 1 (specifically 1,2,3 and 4) are positioned closer together, while at the same time the vertices 1 and 4 that had links with 5 are closer to it than nodes 3 and 2, which had not.

4.5.1 Metropolis - within - Gibbs Estimation

The Latent Distance Model requires estimation not only of the covariate coefficients $\beta_k, k = 1, \ldots, p$ and constant μ, but also of the latent vertex positions, $z_i, i = 1, \ldots, |V|$. It is only natural, therefore, that the Bayesian approach is employed and, in particular, the Metropolis - within - Gibbs MCMC algorithm.

The likelihood function of the model, denoted $f(Y|Z, \boldsymbol{\beta})$, is in the form:

$$f(Y \mid Z, \boldsymbol{\beta}) = \prod_{i \neq j \in V} p_{i,j}^{y_{i,j}} \cdot (1 - p_{i,j})^{1 - y_{i,j}}$$

where Z is a matrix of dimensions $2 \times |V|$ whose columns are the vertices' latent positions and $\boldsymbol{\beta}$ is a vector of dimensions $(k+1) \times 1$ including the constant and covariate coefficients. We specify prior distributions for the model parameters:

$$\boldsymbol{z_i} \overset{\text{iid}}{\sim} N_2(0, \Sigma)$$
$$\boldsymbol{\beta_k} \overset{\text{iid}}{\sim} N_1(0, \sigma^2)$$

Therefore, using Bayes Rule, we can express the joint posterior distribution of the model parameters as following:

$$f(Z, \boldsymbol{\beta} \mid Y) = \frac{f(Y \mid Z, \boldsymbol{\beta}) \cdot f(Z) \cdot f(\boldsymbol{\beta})}{f(Y)}$$

or, by neglecting the denominator and splitting the joint prior distributions:

$$f(Z, \boldsymbol{\beta} \mid Y) \propto f(Y \mid Z, \boldsymbol{\beta}) \cdot f(z_1) \cdot \cdots \cdot f(z_{|V|}) \cdot f(\beta_0) \cdot \cdots \cdot f(\beta_k)$$

The denominator is the normalizing constant, that ensures posterior probabilities sum to 1. All model parameters are considered to be independent to one another.

In the Metropolis - within - Gibbs MCMC algorithm, each parameter is sampled through its conditional distribution using a proposal distribution. The formula for a conditional distribution is $f(x \mid y) = \frac{f(x,y)}{f(y)}$, or joint over marginal. The conditional distribution of the parameters $\beta_k, k = 0, \ldots, p$ and $z_i, i = 1, \ldots, |V|$, hence, are:

$$f(z_i \mid Y, \boldsymbol{\beta}, z_{-i}) \propto f(Y \mid Z, \boldsymbol{\beta}) \cdot f(z_i)$$
$$f(\beta_k \mid Y, Z, \beta_{-k}) \propto f(Y \mid Z, \boldsymbol{\beta}) \cdot f(\beta_k)$$

By the notation z_{-i} and β_{-k}, are referred to all the parameters besides z_i and β_k respectively. The MCMC procedure to obtain a sample from the posterior distribution is:

- Assign arbitrary initial values to the parameters: $z_i^{(0)}, i = 1, \ldots, |V|$ and $\beta_k^{(0)}, k = 0, \ldots, p$.

For iterations $j = 1, \ldots, B$ **repeat:**

For vertex indices $i = 1, \ldots, |V|$:

- Sample a candidate value $z_i^{(j)}$ from the proposal distribution $N_2(z_i^{(j-1)}, \Sigma)$.
- Calculate the Hastings Ratio $r = \frac{f(z_i^{(j+1)} | Y, z_{-i}, \boldsymbol{\beta})}{f(z_i^{(j)} | Y, z_{-i}, \boldsymbol{\beta})}$.
- Accept the new value with probability $\min(r, 1)$.

For coefficient indices $k = 0, \ldots, p$:

- Sample a candidate value $\beta_k^{(j)}$ from the proposal distribution $N_1(\beta_k^{(j-1)}, \sigma^2)$.
- Calculate the Hastings Ratio $r = \frac{f(\beta_k^{(j+1)}|Y,Z,\beta_{-k})}{f(\beta_k^{(j)}|Y,Z,\beta_{-k})}$.
- Accept the new value with probability $\min(r, 1)$.

This concludes a Markov Chain going through the following states in each iteration:

$$(z_1^{(j)}, z_2^{(j)}, \ldots, z_{|V|}^{(j)}, \beta_0^{(j)}, \beta_1^{(j)}, \ldots, \beta_k^{(j)}) \to (z_1^{(j+1)}, z_2^{(j)}, \ldots, z_{|V|}^{(j)}, \beta_0^{(j)}, \beta_1^{(j)}, \ldots, \beta_k^{(j)}) \to \cdots \to$$
$$(z_1^{(j+1)}, z_2^{(j+1)}, \ldots, z_{|V|}^{(j+1)}, \beta_0^{(j)}, \beta_1^{(j)}, \ldots, \beta_k^{(j)}) \to (z_1^{(j+1)}, z_2^{(j+1)}, \ldots, z_{|V|}^{(j+1)}, \beta_0^{(j+1)}, \beta_1^{(j)}, \ldots, \beta_k^{(j)}) \to$$
$$(z_1^{(j+1)}, z_2^{(j+1)}, \ldots, z_{|V|}^{(j+1)}, \beta_0^{(j+1)}, \beta_1^{(j+1)}, \ldots, \beta_k^{(j)}) \to \cdots \to (z_1^{(j+1)}, z_2^{(j+1)}, \ldots, z_{|V|}^{(j+1)}, \beta_0^{(j+1)}, \beta_1^{(j+1)}, \ldots, \beta_k^{(j+1)})$$

The result is a random sample from the joint posterior distribution $f(Z, \boldsymbol{\beta} \mid Y)$ of values:

$$(z_1^{(j)}, z_2^{(j)}, \ldots, z_{|V|}^{(j)}, \beta_0^{(j)}, \beta_1^{(j)}, \ldots, \beta_k^{(j)}), j = 1, \ldots, B$$

for a sufficient number of iterations, e.g. $B = 10000$. A burn - in batch of sample values should be discarded, whose size is irrelevant is B is large enough, to neutralize the effect of the initial parameter values.

4.6 ERGM

ERGMs or Exponential Family Random Graph Models, often denoted p^*, are models that consider several dependencies of networks, and are thoroughly presented by van Der Pol (2019). Dependencies that the ERGM could capture include transitivity, reciprocity and homophily. **Transitivity** is the tendency of edge $v_i \leftrightarrow v_k$ to form, if edges $v_i \leftrightarrow v_j$ and $v_j \leftrightarrow v_k$ exist, often playfully referred to as "a friend of my friend is also my friend". ERGMs are applicable both in directed and undirected networks, in either case the dependent variable is the entire network. If the adjacency matrix of dimensions $|V| \times |V|$ that can adequately describe a network is denoted Y, and y is the adjacency matrix of the **observed** network, then the ERGM can be expressed as:

$$\mathbb{P}(Y = y) = \frac{1}{c} \cdot \exp\left(\sum_k \theta_k \cdot G_k(y)\right)$$

This is also the likelihood of the model, i.e. the probability of observing the network as a whole given the set of parameters. Quantities $G_k(y)$ can be either node - or edge - level characteristics, or structural descriptive statistics of the network and θ_k are their coefficients. Some examples for the former could be:

- **Number of edges**: $\sum_{j=i+1}^{|V|} \sum_{i=1}^{|V|-1} y_{i,j}$ for undirected networks with symmetrical adjacency matrices or $\sum_{j=1}^{|V|} \sum_{i=1}^{|V|} y_{i,j}$ for directed ones with asymmetrical adjacency matrices. Alternatively, network density can be used.

- **Reciprocity of the Network**: $\sum_{j=1}^{|V|}\sum_{i=1}^{|V|} y_{i,j} \cdot y_{j,i}$ naturally defined for directed networks.
- **Number of triangles**: $\sum_{i=1}^{|V|}\sum_{j=1}^{|V|}\sum_{k=1}^{|V|} y_{i,j} \cdot y_{j,k} \cdot y_{k,i}$, which is also referred to as **transitivity**.
- **Number of $k-$ Stars**, as presented in Figure 2.

In the denominator of the likelihood function, $c = \sum_{y} \exp\left(\sum_{k} \theta_k \cdot G_k(y)\right)$, is the normalizing constant which sums for all possible network configurations y. This constitutes the issue with the estimation of the ERGM, as the normalizing constant is intractable for any network of realistic size. In particular, the number of possible networks is $2^{|V|\cdot(|V|-1)/2}$ for undirected networks and $2^{|V|\cdot(|V|-1)}$ for directed.

4.6.1 Probability of an edge forming

Since the model is defined from its likelihood, which is in the form of the Exponential Family of Distributions likelihood, a method needs to be devised to move from the probability of observing the whole network $\mathbb{P}(Y = y)$, to the probability of an edge between two vertices forming, $\mathbb{P}(y_{i,j} = 1 \mid G_k(y))$.

The first step is denoting y the adjacency matrix of the network with the edge between v_i and v_j present ($y_{i,j} = 1$) and y^- the adjacency matrix where the edge between v_i and v_j is absent ($y_{i,j} = 0$). Then, the odds of the probability of observing the network with edge present over the probability of observing the network with the edge absent is:

$$\begin{aligned} odds = \frac{\mathbb{P}(Y = y)}{\mathbb{P}(Y = y^-)} &= \frac{\frac{1}{c} \cdot \exp\left(\sum_{k} \theta_k \cdot G_k(y)\right)}{\frac{1}{c} \cdot \exp\left(\sum_{k} \theta_k \cdot G_k(y^-)\right)} \\ &= \exp\left(\sum_{k} \theta_k \cdot G_k(y) - \sum_{k} \theta_k \cdot G_k(y^-)\right) \\ &= \exp\left(\sum_{k} \theta_k \cdot (G_k(y) - G_k(y^-))\right) \end{aligned}$$

The normalizing constant, c, is eliminated from the fraction. By applying the natural logarithm to both sides, the log - odds can be obtained:

$$\log(odds) = \log\left(\frac{p_{i,j}}{1 - p_{i,j}}\right) = \sum_{k} \theta_k \cdot (G_k(y) - G_k(y^-))$$

Therefore, given estimates for the coefficients $\hat{\theta_k}$, the log - odds of the probability of an edge forming between nodes v_i and v_j: $p_{i,j} = \mathbb{P}(y_{i,j} = 1 \mid G_k(y))$ can be calculated, which resembles the form of the logistic regression. Naturally, $G_k(y)$ are the descriptive statistics of the network with the edge $v_i \leftrightarrow v_j$ and $G_k(y^-)$ are the descriptive statistics of the network without it.

4.6.2 MCMC - MLE Estimation

Obtaining the MLE for the parameters θ_k is no straightforward task, as the intractable normalizing constant c forbids any analytical calculations in the likelihood. Therefore, obtaining $\hat{\theta_k} = \arg\max_{\theta} \mathbb{P}(Y = y)$ must be approached through simulation. The problem is presented in the paper of Hunter & Handcock (2006).

Firstly, the likelihood function of the ERGM can be presented in the form of the exponential family:

$$L_y(\theta) = \mathbb{P}_\theta(Z = z) = exp\bigg(\theta^T \cdot g(z) - \psi(z)\bigg)$$

where, for the ERGM $\theta^T = \bigg(\theta_1, \ldots, \theta_k\bigg)$, also $g(z) = g(y) = \bigg(G_1(y), \ldots, G_k(y)\bigg)$ is the vector of network statistics and, lastly, $\psi(z) = \psi(y) = \log \sum_y exp\bigg(\sum_k \theta_k \cdot G_k(y)\bigg)$ is the normalizing constant. Subsequently, the log - likelihood function is obtained by applying the natural logarithm to both sides of the equation:

$$\ell_y(\theta) = \log L_y(\theta) = \theta^T \cdot g(z) - \psi(z)$$

Obtaining the derivative of the log - likelihood for parameter vector θ is straightforward:

$$\frac{\partial \ell_y(\theta)}{\partial \theta} = \frac{\partial}{\partial \theta}\bigg(\theta^T \cdot g(y) - \psi(y)\bigg) = g(y) - \mathbb{E}_\theta(g(Y))$$

and $\frac{\partial \psi(y)}{\partial \theta} = \mathbb{E}_\theta(g(Y))$, because:

$$\begin{aligned}
\frac{\partial \psi(y)}{\partial \theta} &= \frac{\partial}{\partial \theta} \log \sum_y \exp\bigg(\sum_k \theta^T \cdot g(y)\bigg) = \frac{\frac{\partial}{\partial \theta}\sum_y \exp\bigg(\theta^T \cdot g(y)\bigg)}{\sum_y \exp\bigg(\theta^T \cdot g(y)\bigg)} \\
&= \frac{\sum_y g(y) \cdot \exp\bigg(\theta^T \cdot g(y)\bigg)}{\sum_y \exp\bigg(\theta^T \cdot g(y)\bigg)} = \sum_y g(y) \cdot \frac{\exp\bigg(\theta^T \cdot g(y)\bigg)}{\sum_y \exp\bigg(\theta^T \cdot g(y)\bigg)} \\
&= \sum_y g(y) \cdot \mathbb{P}_\theta(Y = y) = \mathbb{E}_\theta(g(Y))
\end{aligned}$$

where $\mathbb{P}_\theta(Y = y)$ is the probability under the ERGM. The estimation of the ERGMs' parameters is a combination of Metropolis - Hastings and gradient ascent algorithms. In particular, the Metropolis - Hastings algorithm is utilized to alter an edge at random in the observed network. The entire estimation procedure is as follows:

- Assign initial values to the parameters $\bigg(\theta_1^{(0)}, \ldots, \theta_k^{(0)}\bigg)$

Metropolis - Hastings Part

- Based on the last parameter values $\left(\theta_1^{(j)}, \ldots, \theta_k^{(j)}\right)$
- Alter an edge in the network $y^{(t)}$ at random, resulting in $y^{(t+1)}$, which means:
    1. If the edge exists, remove it.
    2. If the edge does not exist, add it to the network.
- Calculate the network statistics after the alteration, $\left(G_1(y), \ldots, G_k(y)\right)$
- Accept the change with probability $\min\left(1, \mathbb{P}_\theta(Y = y^{(t+1)})/\mathbb{P}_\theta(Y = y^{(t)})\right)$, which is the ratio of the likelihood functions.
- Repeat a significant number of times, say $B = 1000$.

Gradient Ascent Part

- Calculate the **gradient**, i.e. the derivative of the log - likelihood function, which is $g(y) - \mathbb{E}_{\theta^{(j)}}(g(Y))$, the mean is calculated as the sample average of the network statistics $g(y^{(t)}), t = 1, \ldots, B$ from the simulated values of the Metropolis - Hastings part.
- Update the parameter estimates as: $\hat{\theta}^{(j+1)} = \hat{\theta}^{(j)} + \alpha \cdot \left(g(y) - \mathbb{E}_{\theta^{(j)}}(g(Y))\right)$
- Keep returning to the Metropolis - Hastings part until the change in the parameter estimates is negligible.

Quantity α is referred to as "step size" or "learning rate" and should be set as a small number, e.g. $\alpha = 0.01$. This algorithm aims to maximize the likelihood through gradient ascent, but approaches the (intractable) gradient through MCMC simulation. Gradient Ascent is susceptible to being trapped in local maxima, this issue however can be addressed by repeating the estimation process several times and comparing the parameter estimates.

Interestingly, both the Erdős–Rényi and network block model can be expressed in the ERGM form. For the former, the only network descriptive statistic $G(y)$ is the edge count $|E|$ and for the latter, the necessary network statistics $G(k)$ are the number of edges within and between blocks, earlier denoted as $e_{q,r}$.

5 Community - Detection Methods

Community - detection essentially refers to the application of appropriate clustering methods to network data. Clustering is a group of unsupervised statistical methodologies that aim at finding groups of data points. Generally, an application of clustering is successful if data points within the same cluster are more similar and data points of different clusters are dissimilar. It is a modern approach that can address a variety of real - world scenarios. The following could be some use - cases of community - detection in social networks:

- **Social Media**: Clustering of users based on their common connections could improve the group, page or friend suggestion.
- **Epidemiology and Public Health**: Detecting clusters of persons in an area based on their common encounters could help identifying susceptible individuals and containing the spread, when a case of an infectious disease is recorded.
- **Marketing**: Detecting communities of consumers with the goal of personalizing commercial campaigns, based on what neighbors (close friends or friends - of - friends) of a consumer are buying.

The practical advantages of community detection methods are under no circumstances negligible. Firstly, clustering improves the network visualization, as nodes of different clusters could be assigned different colors or shapes, making them distinctive. Moreover, the clusters that result from these methods can be used as the blocks to estimate a Network Block Model, helping a link - prediction effort if no other covariates (such as age groups, education level, political affiliations etc.) are available. Generally, they serve as a way to obtain and examine a more coarse, general structure from large - scale networks.

Of the three methods that will be presented, Hierarchical Clustering is a "hard - clustering" method. It results in vertices having a strict, single - cluster membership. On the other hand, Stochastic Block Model and the two - step approach constitute "soft - clustering" methods. This means that the node membership to each community is supplemented with a probability, thus introducing uncertainty to the problem.

5.1 Hierarchical Clustering

Hierarchical clustering is a distance - based clustering method. It is considered a significantly versatile method, as given the choice of an appropriate distance or similarity metric, it can be used to cluster various or mixed data types. A major difference compared to other methods, is that through it, a hierarchy of clusters is obtained. This means that the choice of cluster number, G, is made post - hoc, as all the possible configurations of clusters can be obtained. The result of the clustering algorithm is visualized through the dendrogram, in Figure 6.

For instance, by obtaining $G = 2$ clusters, these would be $\{1, 4\}$ and $\{2, 3, 5\}$. Alternatively, by obtaining $G = 3$ clusters would result in $\{1, 4\}$, $\{2, 5\}$ and $\{3\}$.

Hierarchical clustering algorithms are divided into two categories: the **agglomerative** and the **divisive**. The agglomerative, also called "bottom - up" approach, begins by considering each data point (in the case of network data, each node) as a separate cluster. Then, it iteratively merges the two closest or most similar clusters in each step. It continues, until all data points are contained within one cluster. The

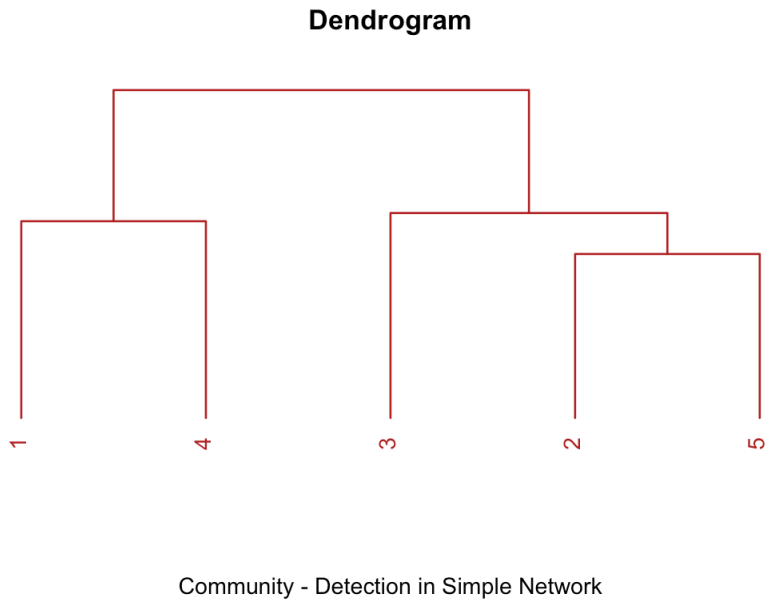


Figure 6: The dendrogram of hierarchical clustering in the network of Figure 1.

divisive, also referred to as "top - down" approach, begins by considering all data points to belong in one cluster. It iteratively divides clusters in each step until each data point is its own cluster.

The whole process is based on the information contained in a distance, or dissimilarity, matrix. Therefore, the only specifications necessary before the initialization of the algorithm is the distance metric and the linkage criterion. The **linkage criterion** refers to the way the distance between two clusters is calculated.

The simplest three choices of linkage criterion are the following:

- **Complete Linkage**: Also called the "furthest - neighbor" method, considers the dissimilarity between two clusters (C_1 and C_2) to be the dissimilarity of their two most dissimilar data points.

$$d(C_1, C_2) = \max_{v_i \in C_1, v_j \in C_2} \Big[d(v_i, v_j) \Big]$$

- **Single Linkage**: Also called the "nearest - neighbor" method, considers the dissimilarity between two clusters (C_1 and C_2) to be the dissimilarity of their two least dissimilar data points.

$$d(C_1, C_2) = \min_{v_i \in C_1, v_j \in C_2} \Big[d(v_i, v_j) \Big]$$

- **Average Linkage**: Considers the dissimilarity of two clusters to be the average dissimilarity of all pairs of data points belonging in the different clusters.

$$d(C_1, C_2) = \frac{1}{|C_1| \cdot |C_2|} \cdot \sum_{v_i \in C_1} \sum_{v_j \in C_2} \Big[d(v_i, v_j) \Big]$$

A drawback of the Hierarchical Clustering method is its computational requirements. Specifically, the

distance matrix that is required contains the distances / dissimilarities between every pair of network nodes. Therefore, its dimensions are $|V| \times |V|$, making it unfeasible to calculate and store for large - scale networks of several thousand nodes. An application of the hierarchical clustering method in network data is comprehensively presented by Liao et al. (2019).

Their suggestion of similarity metric is the **Jaccard Similarity**, applied in the adjacency matrix, which is a quantity proportional to their number of common neighbors. The set of the neighbors of a node v_i can be denoted as $N(v_i)$. Then, the Jaccard Similarity is simply computed as:

$$S_{Jaccard}(v_i, v_j) = \frac{|N(v_i) \cap N(v_j)|}{|N(v_i) \cup N(v_j)|} = \frac{|N(v_i) \cap N(v_j)|}{|N(v_i)| - |N(v_i) \cap N(v_j)| + |N(v_j)|}$$

The notation $|\cdot|$ refers to the size of the neighbor set of a vertex. As an example, in the simple network of Figure 1, which features cliques, the Jaccard Similarity between vertices 3 and 5 will be calculated.

- The set of neighbors of vertex 3 is $N(v_3) = \{1, 2, 4\}$ and its size is $|N(v_3)| = 3$. Similarly, the neighbor set of vertex 5 is $N(v_5) = \{1, 4\}$ and its size $|N(v_5)| = 2$.
- The Jaccard similarity is calculated as following, considering their number of common neighbors of the two nodes are two:

$$S_{Jaccard}(v_3, v_5) = \frac{|N(v_3) \cap N(v_5)|}{|N(v_3)| - |N(v_3) \cap N(v_5)| + |N(v_5)|} = \frac{2}{3 - 2 + 2} = 2/3$$

- To transform the similarity metric to a distance metric, it is subtracted from 1:

$$d_{Jaccard}(v_3, v_5) = 1 - S_{Jaccard}(v_3, v_5) = 1/3$$

Therefore, the Jaccard Similarity is a measure of how similar the neighbor sets of two vertices are. A hierarchical community - detection approach based on the Jaccard Similarity will identify clusters of nodes with similar connections to each other, and dissimilar connection with nodes of other clusters.

The agglomerative algorithm can be explained as:

1. Calculate the $|V| \times |V|$ distance matrix, D, based on the Jaccard Similarity and the adjacency matrix, A. Consider each one of the $|V|$ nodes as a separate cluster $C_i, i = 1, \ldots, |V|$.
2. Merge the two nearest (most similar, based on the Jaccard Similarity) clusters, according to the chosen linkage criterion. Recalculate the distance matrix, D, after the merge.
3. Repeat this process until all nodes are contained within one cluster. Then, the results of the process can be studied through the dendrogram.

Alternatively, the $|V| \times |V|$ distance matrix based on the geodesic distances could be used for community detection. However, this approach has a practical implication. Geodesic distance is the shortest number of steps between two vertices, in observed networks many geodesic distances would have the minimum value of one. Therefore, the algorithm would have to be adjusted to cluster them in random order. Also, in the case of disconnected nodes the geodesic distance tends to infinity.



Figure 7: A comparison of Jaccard versus geodesic distances in clustering of Figure 1 network.

Naturally, the chosen distance metric reflects the purpose of the clustering and does not yield the same result, as presented in Figure 7. In the left, nodes were clustered based on the similarity of their neighbor sets. In the right, nodes were clustered based on how "close" to each other they are in the network.

In practical scenarios, a straightforward way to assess clustering success or compare possible values of G, is through **silhouette values**. The silhouette value, $S_i, i = 1, \ldots, |V|$ is a coefficient with values in the interval $-1 \leq S_i \leq +1$. It quantifies how well a specific vertex, v_i, is placed in a cluster by comparing its similarity to vertices of the same cluster and dissimilarity with vertices of other clusters. For any vertex $v_i, i = 1, \ldots, |V|$, it is calculated as:

$$S_i = \frac{b_i - a_i}{\max(a_i, b_i)}$$

Quantity a_i denotes the average distance / dissimilarity of vertex v_i to other vertices of the same cluster and b_i denotes the average distance or dissimilarity of v_i to vertices of the **closest** cluster to its own. By considering vertex $i = 1, \ldots, |V|$ to belong to cluster $g = 1, \ldots, G$ as $v_i \in C_g$, these quantities can be expressed mathematically as:

$$a_i = \frac{1}{|C_g| - 1} \cdot \sum_{j \in C_g \setminus i} d(v_i, v_j)$$

$$b_i = \min_{g' \neq q} \left\{ \frac{1}{|C_{g'}|} \sum_{j \in C_{g'}} d(v_i, v_j) \right\}$$

where $|C_g|$ refers to the size of the cluster $g = 1, \ldots, G$ and $d(v_i, v_j)$ is the distance of nodes v_i and v_j based on the chosen similarity metric. The closer S_i is to $+1$, means that vertex v_i is appropriately placed in its cluster.

Finally, comparison of different values of G equates to comparing the mean silhouette value:

$$\bar{S} = \frac{1}{|V|} \cdot \sum_{i=1}^{|V|} S_i$$

Attention should be given to the fact that, considering the silhouette coefficient is a metric based on distances / dissimilarities, a comparison is valid between two clustering attempts using the same similarity

metric. For instance, comparison of the two approaches in Figure 7 is inappropriate.

5.2 Stochastic Block Model

The Stochastic Block Model, as presented by Kolaczyk & Csárdi (2014), follows the same principle as the network block model, specifically modeling the probability of a link between two nodes to depend exclusively on their block memberships. In contrast to the block model, where the blocks come from a supplementary variable or from the results of a community detection algorithm, in the stochastic block model the membership of each vertex to a block, as well as the nature of the blocks themselves are unknown and must be inferred though data, thus making it a method of node clustering.

The only specification regards the number of blocks, denoted as Q. Then, both the probabilities of every vertex belonging in each block, as well as the link probabilities between two vertices must be estimated. The above parameters, for $i = 1, \ldots, |V|$ and $q, r = 1, \ldots, Q$, are denoted respectively:

$$p_{i,q} = \mathbb{P}(z_i = q|Y)$$
$$\pi_{q,r} = \mathbb{P}(Y_{i,j} = 1 \mid z_i = q, z_j = r)$$

Naturally the block membership probabilities for a specific vertex sum to one: $\sum_{q=1}^{Q} p_{i,q} = 1$ This makes the total number of parameters $2 \cdot |V| + \frac{Q \cdot (Q+1)}{2}$, where the first term encompasses the block membership probabilities ($p_{i,q}$) and the second term refers to the link probabilities ($\pi_{q,r}$), in the case of an undirected network where $\pi_{q,r} = \pi_{r,q}$ for any $q, r = 1, \ldots, Q$. In the case of directed networks, the total number of parameters to be estimated would be $2 \cdot |V| + Q^2$, as $\pi_{q,r} \neq \pi_{r,q}$.

A practical consideration is to identify the optimal number of blocks, or clusters, Q. Since the Stochastic Block Model is a model - based approach, goodness - of - fit mathematical tools based on the likelihood are available. Indicatively:

$$AIC = -2 \cdot \log L(Y) + 2 \cdot d$$
$$BIC = -2 \cdot \log L(Y) + d \cdot \log |V|$$

A specification of Q that yields the smallest AIC or BIC suggests the optimal number of blocks or clusters in the network. The Stochastic Block Model constitutes a soft clustering approach, thus the result is a vector of block membership probabilities for each vertex. Therefore, vertices can be assigned to clusters, e.g. for visualization or link prediction purposes, based on the maximum a - posteriori probability (MAP) criterion, practically belonging to the block for which the membership probability is the highest.

5.3 Latent Position Cluster Model

The latent position cluster model is closely related to the latent distance model, but extends it to capture clusters of nodes in networks. The key assumption is that the latent positions z_i, for $i = 1, \ldots, |V|$, source from a finite mixture of normal distributions.

$$z_i \sim \sum_{k=1}^{K} \pi_k \cdot N_d(\mu_k, \Sigma_k)$$

The above denotes a finite mixture of d - variate Normal Distributions, where d refers to the specified dimensions of the latent Euclidean space. Quantity π_k is the **mixing proportion** of component k, which expresses the prior probability of a vertex belonging in the component, i.e. cluster, k and naturally $\sum_{k=1}^{K} \pi_k = 1$. Parameters μ_k and Σ_k are the mean vector and covariance matrix of the k mixture component.

Naturally, estimating a mixture means estimating $K - 1$ parameters for π_k, $K \cdot d$ parameters for μ_k and $K \cdot \frac{d \cdot (d+1)}{2}$ parameters for Σ_k since the covariance matrix is symmetrical.

Two approaches have been proposed by Handcock et al. (2007), the two - step maximum likelihood estimation and the simultaneous Bayesian estimation. In the former, we first obtain estimates of the latent positions z_i and then perform model based clustering based on them. In the latter, latent positions and cluster membership is inferred simultaneously.

5.3.1 Two - Step Approach

In the two - step approach, the positions of the nodes in the d - dimensional latent Euclidean space, $z_i = (z_{1,i}, \ldots, z_{d,i})^T$ for $i = 1, \ldots, |V|$, which were estimated in a Bayesian manner, are considered fixed quantities. Then, model - based clustering using finite Gaussian mixture models is performed on them.

Specifically, under the mixture model, the probability density function (p.d.f.) of vertex v_i is:

$$f(z_i) = \sum_{k=1}^{K} \pi_k \cdot f_d(z_i \mid \mu_k, \Sigma_k)$$

Naturally, $f_d(\cdot \mid \mu_k, \Sigma_k)$ denotes the p.d.f. of the k-th component of the mixture and along with the set of parameters, specifically:

$$f(x \mid \mu_k, \Sigma_k) = \left((2 \cdot \pi)^d \cdot |\Sigma_k| \right)^{-\frac{1}{2}} \cdot \exp\left[\frac{1}{2} \cdot (x - \mu_k)^T \cdot \Sigma_k^{-1} \cdot (x - \mu_k) \right]$$

$$\mu_k = (\mu_{1,k}, \ldots, \mu_{d,k})^T \qquad \Sigma_k = \begin{pmatrix} \sigma_{1,k}^2 & \ldots & \sigma_{1,d,k} \\ \vdots & \ddots & \vdots \\ \sigma_{d,1,k} & \ldots & \sigma_{d,k}^2 \end{pmatrix}$$

Quantities π_k are also parameters and express the prior probability of a vertex belonging in mixture component $k = 1, \ldots, |V|$. The likelihood of the finite Gaussian mixture model can be constructed as the product of the p.d.f. of all the nodes:

$$L_Y(\pi_k, \mu_k, \Sigma_k) = \prod_{i=1}^{|V|} \left(\sum_{k=1}^{K} \pi_k \cdot f_d(z_i \mid \mu_k, \Sigma_k) \right)$$

Of course, this is a complex model and, nevertheless, the ML estimates could not be directly computed, as the component memberships are **unknown**. In such cases, the **E - M algorithm** (expectation - max-

imization) is of use. The E - M is an iterative algorithm, to maximize the likelihood in problems with missing variables. It alternates between updating the unknown quantities (E - step) and computing the ML estimates (M - Step).

The first step is to obtain a probability for each node to belong in each component, or cluster, combining the information about the cluster size (mixing proportions) and the information of its latent position. Through Bayes' Theorem:

$$\mathbb{P}(K_i = k \mid z_i) = \frac{\mathbb{P}(z_i \mid K_i = k) \cdot \mathbb{P}(K_i = k)}{\mathbb{P}(z_i)} = \frac{f(z_i \mid \mu_k, \Sigma_k) \cdot \mathbb{P}(K_i = k)}{f(z_i)}$$

The denominator can be expressed in a rather convenient form through the Law of Total Probability, which results in the final form of:

$$\mathbb{P}(K_i = k \mid z_i) = \frac{\pi_k \cdot f(z_i \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \cdot f_d(z_i \mid \mu_j, \Sigma_j)}$$

This quantity is called the **responsibility** and is denoted $w_{i,k}$ for $i = 1, \ldots, |V|$ and $k = 1, \ldots, K$. It essentially expresses the posterior probability of a node with latent position z_i belonging in component k.

The first step is to assign initial values for the parameters $\pi_k^{(0)}, \mu_k^{(0)}, \Sigma_k^{(0)}$ for $k = 1, \ldots, K$, which could be based on the results of a Hierarchical Clustering algorithm. Then, the iterative algorithm proceeds as following for iterations $t = 0, \ldots, B$:

E - Step: (for $i = 1, \ldots, |V|$ and $j = 1, \ldots, K$)

$$w_{i,j}^{(t+1)} = \frac{\pi_k^{(t)} \cdot f(z_i \mid \mu_k^{(t)}, \Sigma_k^{(t)})}{\sum_{j=1}^{K} \pi_j^{(t)} \cdot f_d(z_i \mid \mu_j^{(t)}, \Sigma_j^{(t)})}$$

M - Step: (for $k = 1, \ldots, K$)

$$\begin{aligned}
\pi_k^{(t+1)} &= \frac{\sum_{i=1}^{|V|} w_{i,k}^{(t+1)}}{|V|} \\
\mu_k^{(t+1)} &= \left(1 / \sum_{i=1}^{|V|} w_{i,j}^{(t+1)}\right) \cdot \sum_{i=1}^{|V|} w_{i,k}^{(t+1)} \cdot z_i \\
\Sigma_k^{(t+1)} &= \left(1 / \sum_{i=1}^{|V|} w_{i,k}^{(t+1)}\right) \cdot \sum_{i=1}^{|V|} w_{i,k}^{(t+1)} \cdot (z_i - \mu_k^{(t+1)}) \cdot (z_i - \mu_k^{(t+1)})^T
\end{aligned}$$

Essentially, in the **E - step** the responsibilities (posterior probabilities of component membership) are calculated, and then in the **M - step** the parameter estimates are updated. A consideration for the E - M algorithm is its tendency to become trapped in local maxima.

The two - step approach is based on a pre - selection of the number of components, K. Different configurations for K, e.g. $K = 3$ or $K = 4$, can be compared based on the likelihood and specifically AIC or BIC.

Parametrization of the Covariance Matrix:

Given the number of mixture components K and the specified dimensions of the latent Euclidean social space d, the number of parameters of the mixture model might be inflated. For instance, for the case of $K = 4$ and $d = 3$ the total number of parameters is 39. To address this, a parametrization of the components' covariance matrix can be introduced, which encompasses the most parameters.

$$\Sigma_k = \lambda_k \cdot D_k \cdot \Lambda_k \cdot D_k^T$$

Quantity λ_k is a scalar, which defines the cluster size. Matrix Λ_k is diagonal and defines the cluster shape, specifically how much it stretches in each dimension. Lastly, D_k defines the cluster orientation in the d - dimensional latent Euclidean space. This is the result of spectral decomposition of the covariance matrix.

Constraints can be imposed on these parameters, resulting in hypotheses about the size, shape and orientation of the clusters. Specifically, the constraint is denoted by three letters, each corresponding to:

Size	Shape	Orientation

Table 2: Cluster covariance matrix constraining.

Below, some specification along with the resulting cluster configurations, are presented as examples:

- VII: Variable cluster size, spherical shape, orientation is irrelevant.
- EVV: Equal cluster size, variable shapes and variable orientations.
- VEV: Variable cluster size, equal shapes (not necessarily spherical), variable orientation.

5.3.2 Bayesian Estimation

The Bayesian estimation approach is naturally more complex than the two step approach. It involves defining appropriate prior distributions for the parameters and then using MCMC sampling. This is an appropriate point to introduce the definition of **prior conjugacy**, as presented by Gelman et al. (1995), because it is exploited in this approach.

Conjugacy is the property where the prior distribution and the posterior distribution follow the same parametric form. For instance, a Beta prior combined with a Binomial likelihood (model) results in a Beta posterior. This is a rather convenient property, as it allows for a tractable posterior distribution. In the Latent Distance Model, for example, univariate and bivariate Normal parameter priors were combined with a likelihood function of the Binomial form, thus resulting in a joint posterior distribution of unknown form. For this reason, Metropolis - within - Gibbs was used, as sampling directly from the conditional posterior distributions was infeasible.

In the case of the Latent Position Cluster Model, the specification of prior distributions results in some conditional posterior distributions being in a known parametric form, and some being in an unknown. In the former, sampling directly from the conditional posterior is feasible. In the latter, sampling from the conditional posterior is done through a Metropolis - Hastings step, involving the acceptance step.

Firstly, a new variable K_i, which expresses the cluster membership for vertex $i = 1, \ldots, |V|$, is introduced. If vertex i belongs to cluster $k = 1, \ldots, K$, it is denoted as $K_i = k$. The Latent Distance Model parametrizes the link probability between vertices v_i and v_j as:

$$\mathbb{P}(Y_{i,j} = 1 | z_i, z_j, X_{i,j}) = p_{i,j} = logit^{-1}\Big(\beta_0^T \cdot X_{i,j} - \beta_1 \cdot ||z_i - z_j||\Big)$$

Latent positions of the nodes are considered to arise from a finite mixture of Normal distributions, which is denoted as following:

$$z_i \sim \sum_{k=1}^{K} \pi_k \cdot N_d(\mu_k, \sigma_k^2 \cdot I_{d \times d})$$

Naturally, d are the dimensions of the defined d - dimensional latent Euclidean social space. In the paper of Handcock et al. (2007), covariance matrices of the components were defined as spherical. Prior distributions are specified for the other parameters:

$$\begin{aligned}
\beta &\sim N_p(\xi, \Psi) \\
\pi &\sim Dirichlet(\nu) \\
\sigma_k^2 &\sim \sigma_0^2 Inv(\chi_\alpha^2) \\
\mu_k &\sim N_d(0, \omega^2 \cdot I_{d \times d})
\end{aligned}$$

Vector $\beta = (\beta_0^T, \beta_1)^T$ are the coefficients of the distance and pair - specific covariates in the link probability. Vector $\pi = (\pi_k), k = 1, \ldots, K$ are the mixing proportions and Dirichlet distribution is the multivariate generalization of the Beta distribution. Parameters σ_k and μ_k are the parameters of the $k = 1, \ldots, K$ component of the finite gaussian mixture and $I_{d \cdot d}$ is the identity matrix of $d \times d$ dimensions. Inverse chi - squared distribution is a distribution commonly used as a conjugate prior.

The next step is to derive the conditional posterior distributions of the parameters, some of which are in known parametric form as a result of prior conjugacy, while others are in unknown form.

$$\begin{aligned}
z_i \mid K_i = k, \beta, \pi, \mu_k, \sigma_k^2 &\propto f_d(z_i \mid \mu_k, \sigma_k^2) \cdot \mathbb{P}(Y | \beta, X, Z) \\
\beta | Z, \pi, \mu_k, \sigma_k^2, K &\propto f_p(\beta \mid \xi, \Psi) \cdot \mathbb{P}(Y | \beta, X, Z) \\
\pi \mid \beta, Z, \mu_k, \sigma_k^2, K &\sim Dirichlet(n + \nu) \\
\mu_k \mid \pi, \beta, Z, \sigma_k^2, K &\sim N_d\left(\frac{m_k \cdot \bar{z_k}}{m_k + \sigma_k^2/\omega^2}, \frac{\sigma_k^2}{m_k + \sigma_k^2/\omega^2} \cdot I_{d \cdot d}\right) \\
\sigma_k^2 \mid \beta, Z, \mu_k, \pi, K &\sim (\sigma_0^2 + d \cdot s_k^2) Inv(\chi^2_{\alpha + m_k \cdot d}) \\
\mathbb{P}(K_i = k \mid z_i, \beta, \mu_k, \sigma_k^2, \pi) &= \frac{\pi_k \cdot f_d(z_i \mid \mu_k, \sigma_k^2)}{\sum_{k=1}^{K} \pi_k \cdot f_d(z_i \mid \mu_k, \sigma_k^2)}
\end{aligned}$$

Term $\mathbb{P}(Y | \beta, X, Z)$ denotes the likelihood function of the model, which is in the Binomial Form, and $f_p(\cdot)$ denotes the probability density function of the p - variate Normal Distribution.

Quantity $n = (n_1, \ldots, n_K)^T$, is the vector whose elements are $n_k = \sum_{i=1}^{|V|} \mathbb{I}(K_i = k)$. Also, $\bar{z_k} =$

$\frac{1}{n_k} \cdot \sum_{i=1}^{|V|} \mathbb{I}(K_i = k) \cdot z_i$ and, lastly, $s_k^2 = \frac{1}{d} \cdot \sum_{i=1}^{|V|} \mathbb{I}(K_i = k)(z_i - \mu_k)^T \cdot (z_i - \mu_k)$. The z_i and μ_k are $d \cdot 1$ vectors, therefore $\bar{z_k}$ is a $d \cdot 1$ vector and s_k^2 is a scalar. As a reminder, d are the dimensions of the latent Euclidean social space.

Similar to the Latent Distance Model, z_i (latent positions) and β (coefficients of pair - specific covariates) must be sampled through a Metropolis - Hastings step because their conditional posterior distribution is in unknown form. The other parameters can be sampled through Gibbs sampling, because due to appropriate selection of priors, their conditional posterior is in known parametric form.
The MCMC algorithm is the following:

- Assign arbitrary initial values to all parameters.

For the latent positions $z_i^t, i = 1, \ldots, |V|$, update of parameters should be performed in random order of nodes as:

- Obtain a candidate value z_i^{t+1} from a multivariate Normal Proposal distribution $N_d(z_i^t, \alpha \cdot I_{d \times d})$
- Calculate the Hastings ratio $r_z = \frac{f_d(z_i^{t+1}|\mu_k, \sigma_k^2) \cdot \mathbb{P}(Y|\beta, X, Z^{t+1})}{f_d(z_i^t|\mu_k, \sigma_k^2) \cdot \mathbb{P}(Y|\beta, X, Z^t)}$.
- Accept the candidate value z_i^{t+1} with probability $\min(r_z, 1)$, otherwise $z_i^{t+1} \leftarrow z_i^t$.

For the pair - specific covariate coefficients β, update of parameters is accordingly:

- Obtain a candidate value β^{t+1} from a multivariate Normal Proposal distribution $N_p(\beta^t, \delta \cdot I_{p \times p})$.
- Compute the Hastings ratio $r_\beta = \frac{f_p(\beta^{t+1}|\xi, \Psi) \cdot \mathbb{P}(Y|\beta^{t+1}, X, Z)}{f_p(\beta^t|\xi, \Psi) \cdot \mathbb{P}(Y|\beta^t, X, Z)}$.
- Accept the candidate value with probability $\min(r_\beta, 1)$, else $\beta^{t+1} \leftarrow \beta^t$.

Regarding the parameters $K_i, \mu_k, \sigma_k^2, \pi_k$ for $i = 1, \ldots, |V|$ and $k = 1, \ldots, K$:

- Update them sequentially, directly through their conditional posterior distribution as in the Gibbs sampling framework.

The **hyper-parameters** of the proposal distributions, α and δ, are to be identified through experimentation. Of course, an initial batch of joint posterior samples, the "burn - in", should be discarded.

6 Application in real data

In this section, an application of some of the described methods in real data will be presented. Specifically, the data portray research collaborations between faculty members of Athens' major Statistics Departments, those of Athens University of Economics and Business (hereinafter ``AUEB'') and University of Piraeus (hereinafter ``UNIPI''), in form of published paper co-authorship. The data, therefore, constitutes an undirected and unweighted social network, with two distinct sub - networks. They were collected through publicly available data in Google Scholar as of February 10^{th} 2025, from the faculty members' pages.

Applying the methods in real data provides a different perspective, as interesting practical issues arise. For instance, the presence of disconnected vertices is a problem for some of the methods. Indicatively, the force - directed graph visualization techniques, like Kamada - Kawai, in which case disconnected vertices have no spring connections to other vertices and thus no supposed forces applied to them. Furthermore, the geodesic distance (shortest path) between a disconnected node and any other is infinite, complicating the calculation and interpretation of network descriptive statistics, such as Average Path Length and Network Diameter.

The statistical analysis is performed using the open source statistical computing software R. To facilitate the specialized techniques of social network analysis, several packages listed in the appendix were employed.

6.1 Descriptive Analytics

The social network comprises of $|V| = 43$ vertices (faculty members), $V_{AUEB} = 19$ belonging to AUEB and $V_{UNIPI} = 24$ belonging to UNIPI. The network potrays $|E| = 40$ edges in total. Force - directed graph layouts will not be employed, as there are many disconnected nodes present. Using spherical and circular layouts along with visualization enhancement techniques, the network can be visualized as following:

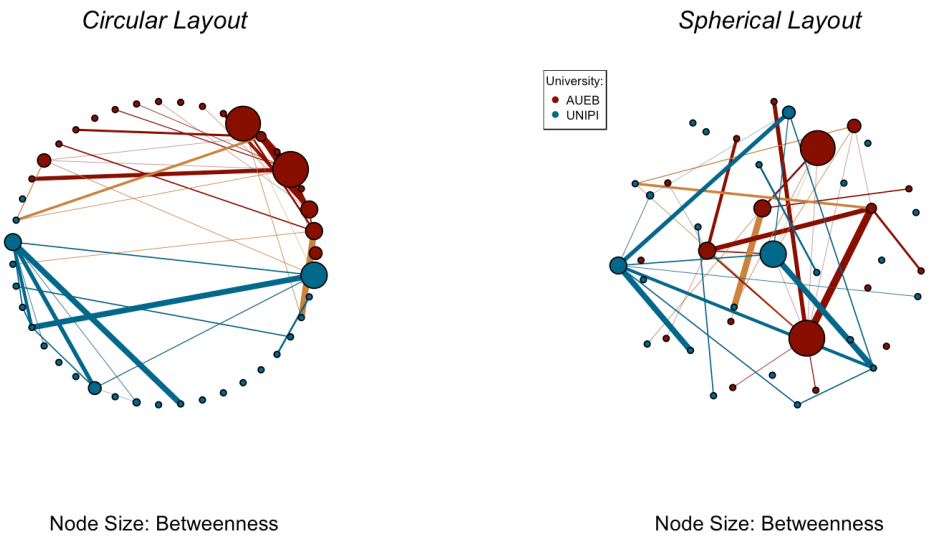


Figure 8: The network of academic collaborations between AUEB and UNIPI faculty members.

The red vertices are faculty members of AUEB, while the blue ones are of UNIPI. Red and blue edges symbolize intra - departmental academic collaboration of AUEB and UNIPI faculty members respectively, while orange edges symbolize inter - departmental collaboration. By edge width, the number of common publications between two nodes is visualized (weight), the wider an edge, the more collaborative publications. The circular layout helps more intuitively visualize the inter - departmental collaboration (orange edges), as vertices of the different departments are positioned anti-diametrically. However, it is not as easy to distinguish disconnected nodes compared to the spherical layout.

It is also of interest, to visually study the network of each university individually.

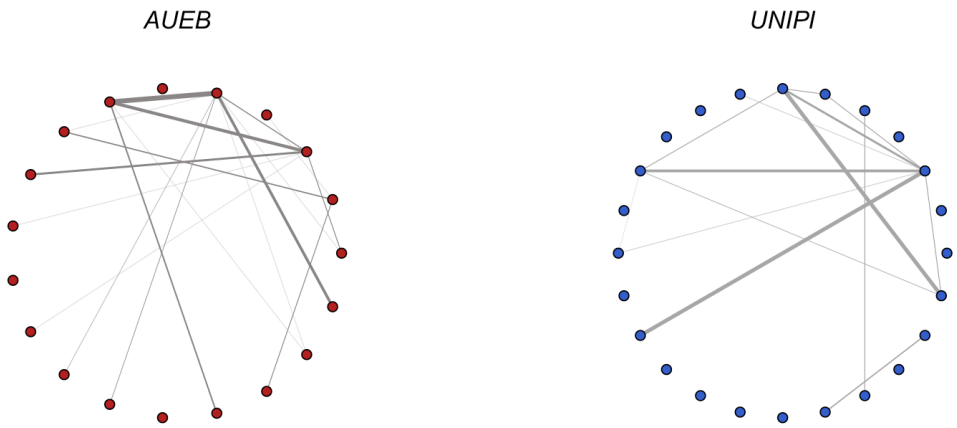


Figure 9: Layouts of the individual networks.

In UNIPI, there appear to be more disconnected nodes than in AUEB. This means that there is a portion of the faculty members that have not collaborated with their colleagues to publish papers. Additionally, the collaborations in AUEB appear to be denser.

Network Density is a descriptive statistic that quantifies the extent of the intra - departmental academic collaboration.

University	**AUEB**	**UNIPI**
Network Density	10.52%	5.07%
Network Transitivity	16.36%	47.36%

Table 3: Comparison of network densities and transitivities of two universities.

Network density (Table 3) is the ratio of realized academic collaborations over possible academic collaborations. A higher density suggests that faculty members collaborate in a greater extent with one another. AUEB's network density of $\approx 10.5\%$ is double than UNIPI's $\approx 5\%$. Therefore, faculty members of the former collaborate with each other more than faculty members of the latter. The overall network density is $\approx 4.42\%$, meaning **only** approximately 5% of the possible academic collaborations take place.

Clustering Coefficient, also referred to as transitivity (Table 3), quantifies the tendency of triads to form triangles. This is an indication that is professor A collaborates with professor B and professor B collaborates with professor C, then professor C might also chose to collaborate with A.

Apparently, in UNIPI the proportion of collaboration triangles, of the form $v_1 \leftrightarrow v_2, v2 \leftrightarrow v_3, v_3 \leftrightarrow v_1$, is significantly greater than in AUEB. The transitivity of the entire network is $\approx 29.32\%$, which means that approximately 30% of the connected triads are triangles.

The descriptive statistics of **network diameter** and **average path length** aim to quantify the size and connectivity of a network, both presented in Table 4.

Descriptive Statistic	**AUEB**	**UNIPI**	**Overall**
Diameter	28	25	39
Average Path Length	10.22	8.63	11.32

Table 4: Diameter and Average Path Length Descriptive Statistics.

A network's diameter is the longest geodesic distance, i.e. the greatest shortest path. For instance, in the overall network the two furthest away faculty members in terms of academic collaboration links are 39 ``steps'' away. The average path length can be interpreted differently. For AUEB, two faculty members are on average (approximately) 10 academic collaborations apart. For UNIPI, two faculty members are on average (approximately) 9 academic collaborations apart. In the overall network, two faculty members are expected to be approximately 11 academic collaboration ``links'' apart. However, due to the presence of disconnected vertices, these descriptive measures are **highly biased**, because the disconnected nodes must be omitted from the calculation.

All previous descriptive measures concern the network as a whole. The following ones concern the individual vertices, here faculty members, and aim to assess their significance and influence in the network. Firstly, the vertex degree empirical distributions will be studied:

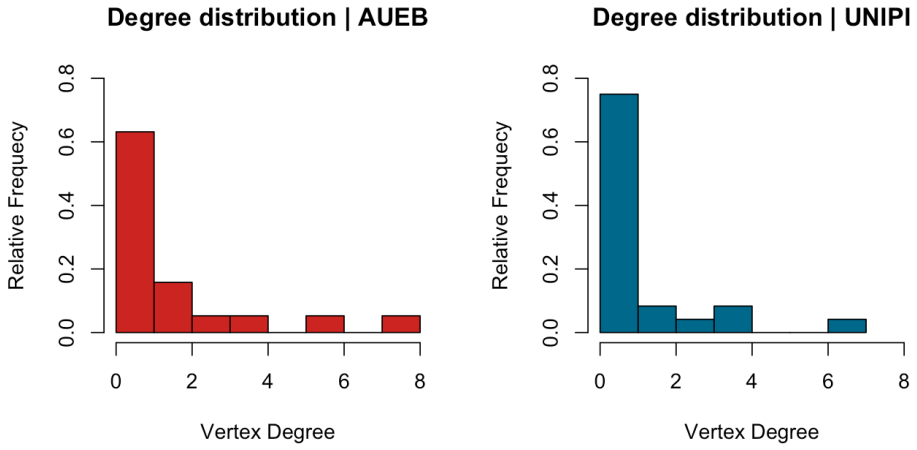


Figure 10: Empirical degree distributions of the vertices of the two networks.

Both department networks have a high presence of nodes with degree 0 or 1, specifically 63% for AUEB and 75% for UNIPI. This corresponds to faculty members without any inter - departmental academic collaboration links, or with only one collaboration link.

More importantly, though, the two departments' vertices appear to have the same form of degree distribution: a highly leptokutric and positively asymmetrical one. This distribution form points to one (or two) faculty member collaborating academically with many colleagues (like a **hub**), while most other have fewer collaborations, shown both in Table 5 and visually in Figure 10.

University	0	1	2	3	4	6	7	8
AUEB	0.16	0.47	0.16	0.05	0.05	0.05	--	0.05
UNIPI	0.50	0.25	0.08	0.04	0.08	--	0.04	--

Table 5: Vertex degree relative frequency table.

Since weight information is also available, in the form of number of collaborations, the empirical vertex strength distribution can also be studied:

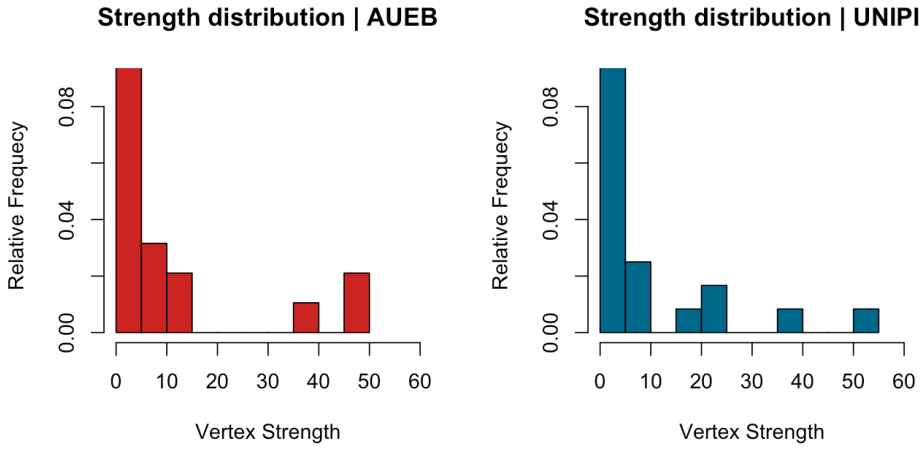


Figure 11: Empirical strength distributions of the vertices of the two networks.

Similarly, the empirical strength distribution presented in Figure 11 is highly leptokurtic and positively asymmetrical. Regardless of department, most faculty members have published $0 - 10$ papers collaborating with colleagues from the same department, while very few have published more.

However, a faculty member's influence in the academic collaboration network is not solely determined through his / hers number of collaborations. Therefore, different **vertex centrality measures** need to be studied, which assess a faculty member's influence based either on his / hers relative position or influence of collaborators.

Betweenness centrality of a node, v, is the proportion of all shortest paths between any two nodes, that pass through v. It quantifies how well positioned ``in the middle'' a vertex is. On the other hand, PageRank centrality consider vertices to be important, if they are connected to other highly connected, but link - parsimonious vertices. Its definition, therefore, mainly serves directed networks, as in undirected ones the in - degree and out - degree of a vertex match.

In this practical setting, betweenness centrality is of great significance. If an important faculty member according to betweenness centrality (a ``**middleman''**) seized to exist from the network (e.g. retired, took a position in another department etc.) the academic collaboration network would decompose into smaller sub - networks, isolated (in terms of academic collaboration) between each other.

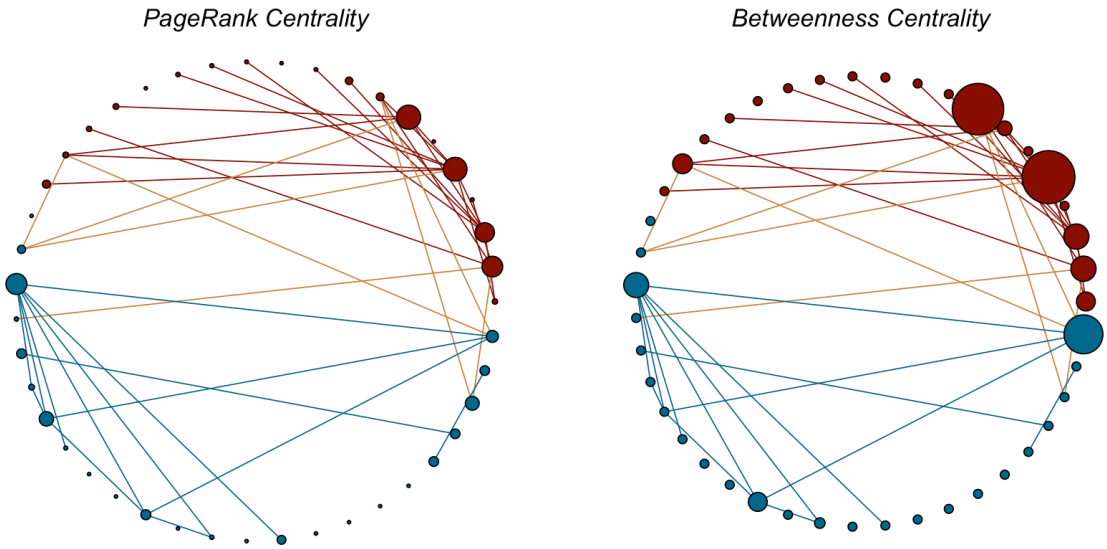


Figure 12: Network visualization encompassing centrality measures as node size.

Due to the presence of many disconnected nodes, the closeness centrality measure can not be examined. Indicatively, however, in Figure 12 there are two highly influential (according to betweenness centrality) faculty members in each department. If these were to disappear, faculty members connected through them would disconnect. In the same sense, it would be a problem in a server network for a node (server) with high betweenness centrality to malfunction, as this would disrupt the passage of information.

To give the opposite example, in epidemiological networks it would be highly beneficial to isolate an actor with high betweeness centrality. By doing so, the transmission of an infectious disease could be contained or halted.

6.2 Statistical Models

In this section, the goal is to estimate and interpret several appropriate models for the academic collaboration network. These models aim to explain the link - forming mechanism of the network.

6.2.1 Erdős–Rényi Model

The Erdős–Rényi model suggest constant network - wide probability of link forming. Hence, each faculty member would be assigned the same probability of collaborating with any other, regardless of department membership, relative position in the network or other characteristics. Conceptually, this is like using a linear regression with only the intercept, to predict the value of a numeric variable.

The MLE of the probability is realized edges, over number of possible edges, which is a quantity that matches the network density. For this network, it was computed as:

$$p = p_{i,j} = \mathbb{P}(Y_{i,j} = 1) = 0.0443$$

Essentially, predicting the academic collaboration between any two faculty members comes down to a Bernoulli trial with $p = 0.0443$. On a similar note, the expected vertex degree is $|V| \cdot p = 43 \cdot 0.0443 =$

1.905. This means that, under the Erdős–Rényi model, every faculty member would be expected to have approximately 2 academic collaborations.

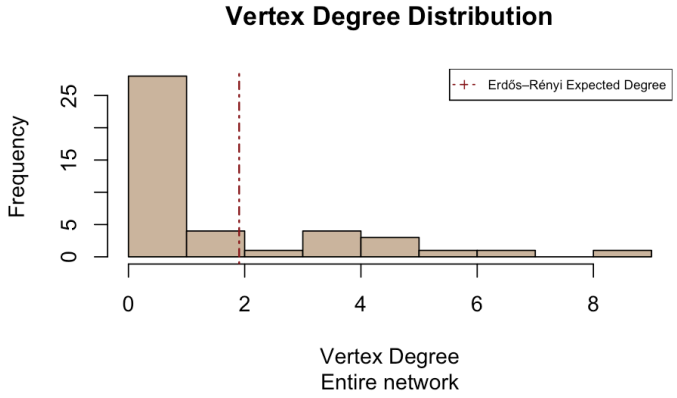


Figure 13: Comparing expected degree under the Erdős–Rényi with sample network data.

However, referring to Figure 13, most faculty members have 0 or 1 collaborations while specific ones have up to 8 or 9. The form of the distribution can also be compared by one of a simulated network under the Erdős–Rényi model with the same number of vertices, in Figure 14.

It is of interest to estimate the Erdős–Rényi model through the ERGM framework. In this case, the network statistic $G_1(y)$ is the number of edges. The estimated model is:

$$\mathbb{P}(Y = y) = \frac{1}{c} \cdot \exp\Big(-3.0715 \cdot G_1(y) \Big)$$

Quantity $c = \sum_y exp\Big[-3.0715 \cdot G_1(y) \Big]$ is the normalizing constant summing over all possible network configurations. The coefficient $\theta_1 = -3.0715$ means that a presence of more edges would reduces the probability of observing our network. Therefore, this is an indication that the network has **edge sparsity**.

It can be shown that the estimate of $\theta_1 = -3.0715$ through the ERGM framework is exactly equivalent to the MLE estimate of the probability $p = 0.0443$. In the ERGM chapter, it was proven that the probability of a specific link forming is:

$$\log\left(\frac{p_{i,j}}{1 - p_{i,j}}\right) = \theta_1 \cdot \Big(G_1(y^+) - G_1(y^-) \Big)$$

Naturally, since $G_1(y)$ refers to the number of edges, the quantity $G_1(y^+) - G_1(y^-)$ is equal to 1 for **any** edge. Therefore:

$$p_{i,j} = p = \frac{e^{\theta_1}}{1 + e^{\theta_1}} = 0.0443$$

A positive ERGM θ_1 coefficient would also be of interest. For example, a network of $|V| = 43$, equal to the one studied here, can be generated by an Erdős–Rényi model, but with $p = 0.8$. Then, the expected number of collaborations for any faculty member would be 34.4, approximately 35 collaborations. This, under no circumstance, constitutes **edge sparsity**, therefore we would expect that the θ_1 coefficient would be positive. Actually, the estimated ERGM is:

$$\mathbb{P}(Y = y) = \frac{1}{c} \cdot \exp\left(+ 1.3092 \cdot G_1(y) \right)$$

Here, the positive coefficient suggests that the network portrays denser edges. The degree distribution for the 1.000 networks simulated under the Erdős–Rényi model is fundamentally different in form as well, as presented in Figure 14

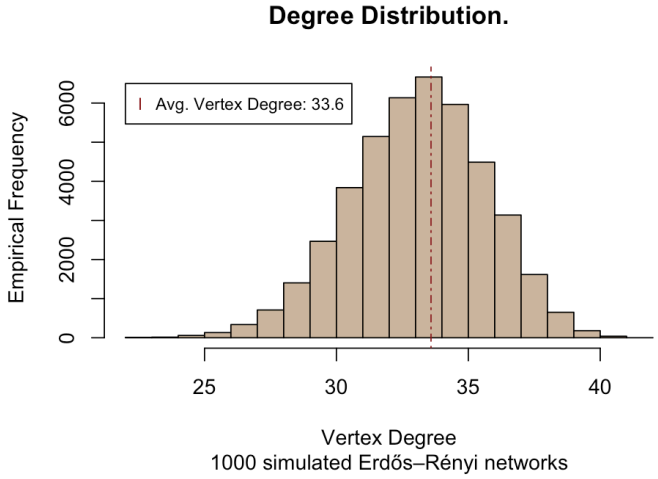


Figure 14: Empirical degree distribution for Erdős–Rényi simulated networks.

A network following the hypothesis of the Erdős–Rényi model, specifically the constant network - wide link probability, would be expected to have a symmetrical degree distribution around the expected degree.

6.2.2 Network Block Model

The network block model considers the link probability between two vertices, v_1 and v_j, to be dependent solely upon their **known** block membership, z_i and z_j. Here, the blocks examined are the academic department in which each faculty member belongs.

$$p_{i,j} = \mathbb{P}(Y_{i,j} = 1 \mid z_i = r, z_j = q) = p_{r,q}$$

As explained in network block model chapter, the MLE of $p_{r,q}$ is the proportion of realized edges between nodes of blocks q and r relative to the number of possible edges between the same blocks, specifically $|V_q| \times |V_r|$. In an undirected network $p_{q,r} = p_{r,q}$. These estimates produce a $Q \times Q$ block probability matrix, where $Q = 2$.

$$\begin{pmatrix} p_{AUEB \leftrightarrow AUEB} & p_{AUEB \leftrightarrow UNIPI} \\ p_{UNIPI \leftrightarrow AUEB} & p_{UNIPI \leftrightarrow UNIPI} \end{pmatrix} = \begin{pmatrix} 0.1053 & 0.0175 \\ 0.0175 & 0.0507 \end{pmatrix}$$

Naturally, the within - department link probabilities (diagonal elements) match the network densities of the corresponding sub - networks. The rows **do not** need to sum to 1, as there is also the probability of no connection, due to unrealized edges of the network. Similar to the Erdős–Rényi, any network block model can also be estimated through the ERGM framework.

$$\mathbb{P}(Y = y) = \frac{1}{c} \cdot \exp\Big(-4.025 \cdot G_1(y) + 1.096 \cdot G_2(y) + 1.885 \cdot G_3(y) \Big)$$

- $G_1(y)$: Number of edges in the network y.
- $G_2(y)$: Number of edges between nodes in UNIPI in network y.
- $G_3(y)$: Number of edges between nodes in AUEB in network y.

The probability of a specific link forming is derived as:

$$\log\left(\frac{p_{i,j}}{1-p_{i,j}}\right) = -4.02 \cdot \Big(G_1(y^+) - G_1(y^-)\Big) + 1.01 \cdot \Big(G_2(y^+) - G_2(y^-)\Big) + 1.88 \cdot \Big(G_3(y^+) - G_3(y^-)\Big)$$

Therefore, an collaboration between faculty members of different departments corresponds to $G_1(y^+) - G_1(y^-) = 1$ and $G_2(y^+) - G_2(y^-) = G_3(y^+) - G_3(y^-) = 0$. The log - odds of a collaboration between two faculty members increase by 1.01 if they both belong at UNIPI and by 1.88 if they both belong at AUEB. Therefore, if two faculty members both belong at AUEB constitutes a higher probability of collaboration compared to both belonging in UNIPI, or at different departments.

The MLE probabilities can be directly obtained from the ERGM formulation as in the case of the Erdős–Rényi. For example, deriving the probability $p_{i,j} = \mathbb{P}(Y_{i,j} = 1 \mid z_i = AUEB, z_j = AUEB) = p_{AUEB \leftrightarrow AUEB}$ is the following process:

$$\begin{aligned} p_{AUEB \leftrightarrow AUEB} &= \frac{e^{-4.02 \cdot (G_1(y^+) - G_1(y^-)) + 1.01 \cdot (G_2(y^+) - G_2(y^-)) + 1.88 \cdot (G_3(y^+) - G_3(y^-))}}{1 + e^{-4.02 \cdot (G_1(y^+) - G_1(y^-)) + 1.01 \cdot (G_2(y^+) - G_2(y^-)) + 1.88 \cdot (G_3(y^+) - G_3(y^-))}} \\ &= e^{-4.02 \cdot 1 + 1.88 \cdot 1} / 1 + e^{-4.02 \cdot 1 + 1.88 \cdot 1} = 0.1053 \end{aligned}$$

This stands, because for the probability of an edge between two AUEB faculty members, the difference in the total edge number statistic is 1, the difference in the total edge number between UNIPI vertices is 0 (did not change) and the difference in total edge number between AUEB vertices is, again, 1.

In terms of model fit, since both the network block model and the Erdős–Rényi lie on ML estimation, the AIC and BIC metrics can be compared. In particular the block model achieves a better fit in the network data compared to the Erdős–Rényi, as shown in Table 7.

Criterion	AIC	BIC
Erdős–Rényi	329.5	334.4
Block Model	312.4	326.8

Table 6: Comparing AIC and BIC criteria between Erdős–Rényi and block model.

It should be noted that, although the Erdős–Rényi model is indifferent to structural characteristics and special dependencies of the network, the network block model manages to capture the phenomenon of **homophily**. This means that faculty members have a greater probability of cooperating with colleagues of the same department, compared to colleagues from different departments.

6.2.3 Latent Distance Model

The Latent Distance Model considers the probability of academic collaboration $p_{i,j}$ between two faculty members, v_i and v_j, to be directly dependent on their positions in a latent (non observable) d-dimensional Euclidean space, z_i and z_j. Thus, $p_{i,j} = \mathbb{P}(Y_{i,j} = 1 \mid z_i, z_j)$.

Apart from a modern network modeling approach, the latent distance model provides a convenient visualization method. Frankly, there is neither a straightforward nor a methodologically correct way to place vertices in a 2-dimensional plot and connect them with edges. The force - directed visualization methods (Kamada - Kawai, Fruchterman - Reingold) try to produce a visually pleasing plot, but fail to highlight any structural information. The latent positions, $z_i, i = 1, \ldots, |V|$, however, contain information about the link structure of the network and especially in the case of $d = 2$ are easy to visualize.

The model is estimated in the Bayesian framework for $d = 2$ with an MCMC sample of 4000 after discarding a ``burn - in'' batch of 10.000 iterations until the Markov chain converges to the posterior distribution.

$$\log\left(\frac{p_{i,j}}{1 - p_{i,j}}\right) = 0.45552 - ||z_i - z_j||$$

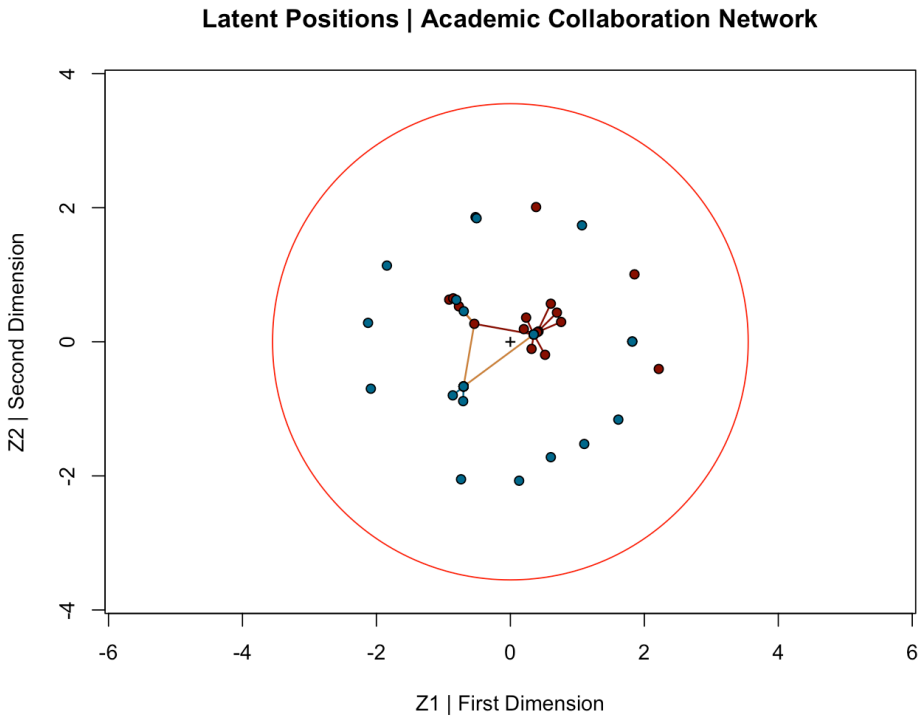


Figure 15: The latent positions of the academic collaboration network

The intercept of the model, $\mu = 3.6517$, as well as the latent positions $z_i = (z_{1,i}, z_{2,i})^T, i = 1, \ldots, |V|$ are the means of the posterior distributions sample. Consequently, for each entire replication of the Bayesian estimation procedure, they would differ.

The plot of the estimates of the latent positions provides interesting information. Firstly, it is convenient to detect the disconnected nodes. Furthermore, three **cores** of strong academic collaboration, positioned

in a way like in the peaks of a triangle. This connects to the subject of betweeness central nodes. These 3 **cores** connect to each other only through 4 specific faculty members. If these faculty members were to be removed from the network, these cores would then be disconnected. This insight, for instance, could not have been uncovered in a traditional visualization layout.

The latent distance model can facilitate node level supplementary variables. In this application, it would be of interest to study the department membership covariate. Once again, the model is based on an MCMC sample of size 4000 with a ``burn - in'' batch of 10.000 discarded. By denoting $K_i, i = 1, \ldots, |V|$ the department membership of node v_i the model can be expressed as:

$$\log\left(\frac{p_{i,j}}{1-p_{i,j}}\right) = -0.89574 + 2.16462 \cdot \mathbb{I}(K_i = K_j = AUEB) \\ + 1.17298 \cdot \mathbb{I}(K_i = K_j = UNIPI) - ||z_i - z_j||$$

As a result, regardless of the distance between their latent positions z_i and z_j, the log - odds of two faculty members, v_i and v_j, collaborating academically in publishing a paper increase by approximately 2.16 if they both belong at AUEB, or by approximately 1.17 if they both belong at UNIPI. This is totally in accordance with the interpretation of the (ERGM) network block model. Practically, the intra departmental collaboration of AUEB is stronger than of UNIPI, both of which are stronger than the inter departmental between faculty members of the two departments.

It is noteworthy that, when adding covariates, the latent positions change. In particular, in absence of covariates, the latent positions must encompass all the information of the network's link structure. If information is available through covariates, the latent positions need to capture less information.

6.2.4 Exponential family Random Graph Model

The ERGM models the network's structural characteristics and dependencies. It can include **homophily** terms, for example in the ERGM formulation of the network block model. It can also include **dyadic dependence** terms, for example number of triangles. Dyadic dependence is defined as the dependence of an edge between two vertices on their relative position in the network.

The ERGM can facilitate vertex characteristics, for instance age, gender etc. As presented by Kolaczyk & Csárdi (2014), these covariates (x) can be introduced to the model as:

$$G_k(y, x) = \sum_{i \neq j} y_{i,j} \cdot h(x_i, x_j)$$

where $h(x_i, x_j)$ is a symmetric function of x_i, x_j. If the goal is to capture the effect of a numeric variable, e.g. h - index, in the tendency of faculty members to collaborate $h(x_i, x_j) = x_i + x_j$ or $\frac{x_i + x_j}{2}$ can be used. On the other hand, if the aim is to capture homophily, i.e. assess the effect of equality in a variable in the probability of an edge, $h(x_i, x_j) = \mathbb{I}(x_i = x_j)$ is appropriate.

The goodness - of - fit in ERGMs can be assessed in two ways. One way is through AIC and BIC criteria based on the likelihood. The other way is to simulate a number of networks under the fitted ERGM and compare the average network statistics of the simulated networks with the network statistics of the observed one, as explained by Kolaczyk & Csárdi (2014).

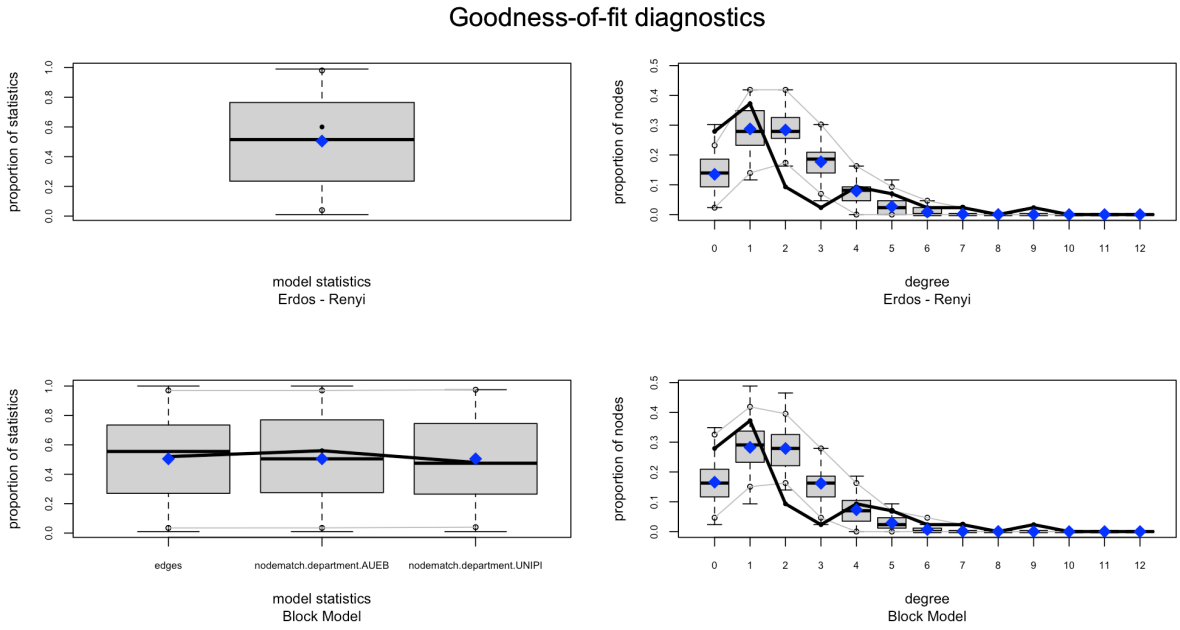


Figure 16: Simulation goodness - of - fit for the Erdős–Rényi and block model ERGMs.

For example, these are the goodness - of - fit assessments for the Erdős–Rényi and block model estimated through the ERGM framework. The boxplots are based on the network statistics of the simulated networks, while the black points and lines are the observed network's respective statistics. The blue rhombuses are the means of the network statistics for the simulated networks. On the left are the terms included in each ERGM and on the right is the degree distribution. None of the models succeeds to capture the network's degree distribution, due to the reason that the observed network potrays **degree heterogeneity** (Figure 13). The mean of the simulated networks of the block model ERGM for each of the statistics: number of edges, number of edges AUEB - AUEB and number of edges UNIPI - UNIPI is not that distant from the one of the observed academic collaboration network.

The ERGM can also be applied to directed networks, in which not all edges are mutual. For example, in a **citation network** Prof. A may cite Prof. B, however Prof. B may not cite Prof. A in his / hers work. In R, it is straightforward to simulate a citations network, N, with $|V| = 43$ researchers, like our academic collaboration network, and prespecified density of $density(N) = 0.15$.

Notice that, in simulating this network, there has not been a specification of **reciprocity**, i.e. proportion of edges that are mutual, nor transitivity. Therefore, the goal is to model them through the ERGM and obtain useful structural information.

$$\mathbb{P}(Y = y) = \frac{1}{c(\theta)} \cdot \exp\Big(\theta_1 \cdot G_1(y) + \theta_2 \cdot G_2(y)\Big), c(\theta) = \sum_y \exp\Big(\theta^T \cdot g(y)\Big)$$

In this ERGM, the included network statistic terms are:

- $G_1(y)$: Number of mutual edges.
- $G_2(y)$: Number of transitive vertex triads.

The simulated network can be studied visually in Figure 17, where the vertex size is defined according

to each vertices in - degree: $in - degree(v_i) = \sum_{j \in V \setminus i} y_{j,i}$ Thus, the size of a vertex suggest how many other researchers cite this specific researcher in their publications.

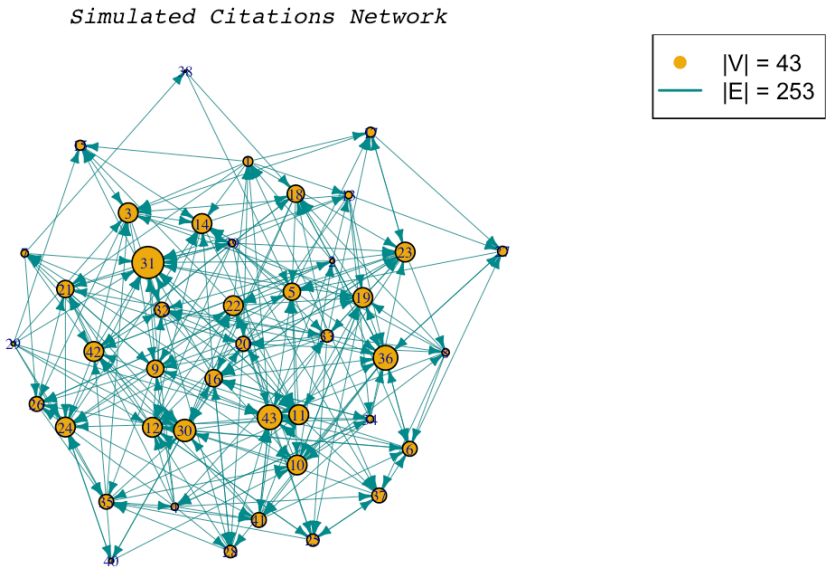


Figure 17: A simulated directed citations network.

The estimated model is the following:

$$\mathbb{P}(Y = y) = \frac{1}{c(\theta)} \cdot \exp\Big(-0.88299 \cdot G_1(y) - 0.57261 \cdot G_2(y) \Big)$$

or, expressed in the form of a logistic model for the probability of a specific edge forming:

$$\log\left(\frac{p_{i,j}}{1 - p_{i,j}}\right) = -0.88299 \cdot (G_1(y^+) - G_1(y^-)) - 0.57261 \cdot (G_2(y^+) - G_2(y^-))$$

As a result, the model detected that the network does not feature strong reciprocity, or strong transitivity. Practically, this means that only in few cases if Prof. A cites Prof. B, then Prof. B cites Prof. A back (reciprocity), or if Prof. A cites Prof. B and Prof. B cites Prof. C, then Prof. C cites Prof. A also (transitivity).

Probability of a reciprocal edge:

$$p = \frac{\exp(-0.88299)}{1 + \exp(-0.88299)} = 0.2925$$

Probability of an edge completing a transitive triad:

$$p = \frac{\exp(-0.57261)}{1 + \exp(-0.57261)} = 0.3606$$

Because the fitted ERGM does not take in to account any other structural terms, the probability of an edge that neither is reciprocal, nor completes a transitive triad is: $p = e^0/(1 + e^0) = 0.5$.

6.3 Community - Detection

The goal of this section is to try and uncover the latent / hidden community structure of the academic collaboration network. Consider that, the communities will not necessarily be the academic departments or only two.

6.3.1 Hierarchical Clustering

Sadly, the hierarchical approach is unsuitable for the network data at hand. The main issue is the presence of disconnected nodes, which disrupt the Jaccard similarity. For instance, suppose a disconnected node, v_1, where $degree(v_1) = 0$. Its similarity with any other (connected) vertex, v_j, is:

$$s(v_1, v_j) = \frac{|N(v_1) \cap N(v_j)|}{|N(v_1) \cup N(v_j)|} = 0 \Rightarrow d(v_1, v_j) = 1$$

Accordingly, the similarity of two disconnected nodes, v_1 and v_2, where $degree(v_1) = degree(v_2) = 0$, is undefined, because:

$$s(v_1, v_2) = \frac{|N(v_1) \cap N(v_2)|}{|N(v_1) \cup N(v_2)|} = \frac{|N(v_1) \cap N(v_2)|}{|N(v_1)| - |N(v_1) \cap N(v_2)| + |N(v_2)|} = 0/0 \rightarrow undef.$$

Therefore, the hierarchical algorithm is forced to insert all disconnected nodes simultaneously at the last step. This problem is clearly observable in Figure 18.

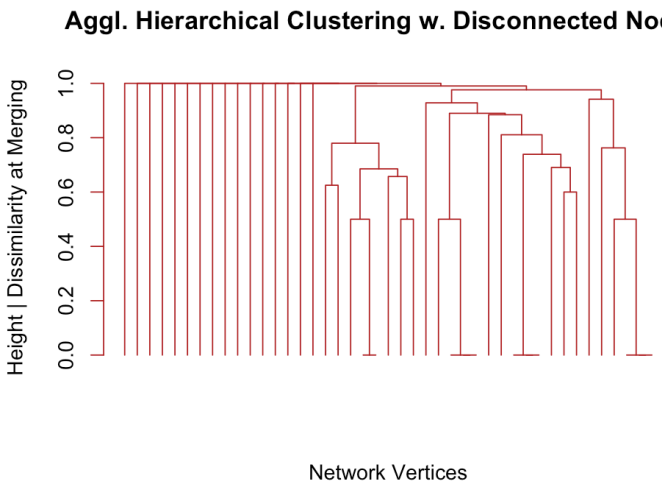


Figure 18: Hierarchical Clustering in the presence of disconnected nodes.

6.3.2 Stochastic Block Model

The Stochastic Block Model is a model based approach with the goal of revealing the hidden community structure and inferring the probability matrix. It estimates the probability of each vertex belonging in each of the Q communities and, simultaneously, the probability of a vertex from community q to form an edge with one from community r: $\pi_{q,r}$. In this case, of an undirected network, naturally: $\pi_{q,r} = \pi_{r,q}, \forall r, q = 1, \ldots, Q$.

The estimation of the Stochastic Block Model in R is performed through a alteration of the Expectation - Maximization algorithm, the **Variational E - M**. Of course, the number of the underlying communities, Q, must be specified prior to the estimation. This is no issue, as the SBM can be estimated for a range of values of $Q = 2, \ldots, 5$. Then, as suggested by Kolaczyk & Csárdi (2014), the selection of Q can be supported by the ``**Integrated Classification Likelihood**'' criterion, which conceptually similar to

the AIC and BIC criteria but adapted to community - detection problems. This is a measure of internal clustering evaluation.

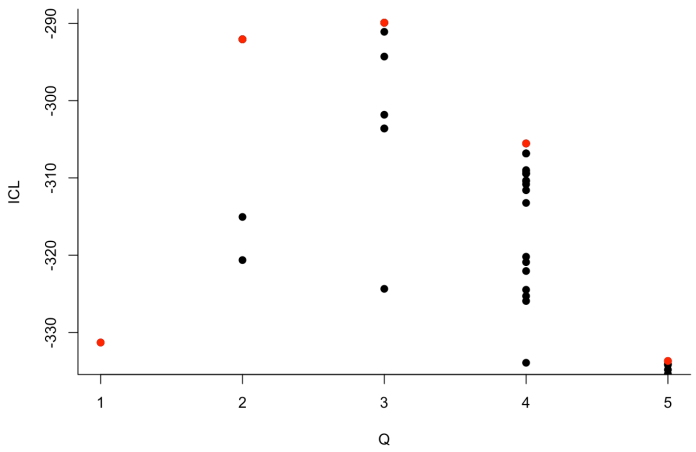


Figure 19: Comparing ICL for different Q specifications.

In the horizontal (x) axis are the different values of Q, specifically from 2 to 5 communities and on the vertical (y) axis are the ICL values (red points). The number of communities with the greatest (mean) ICL is $Q = 3$ communities. The resulting communities can be studied visually in Figure 20.

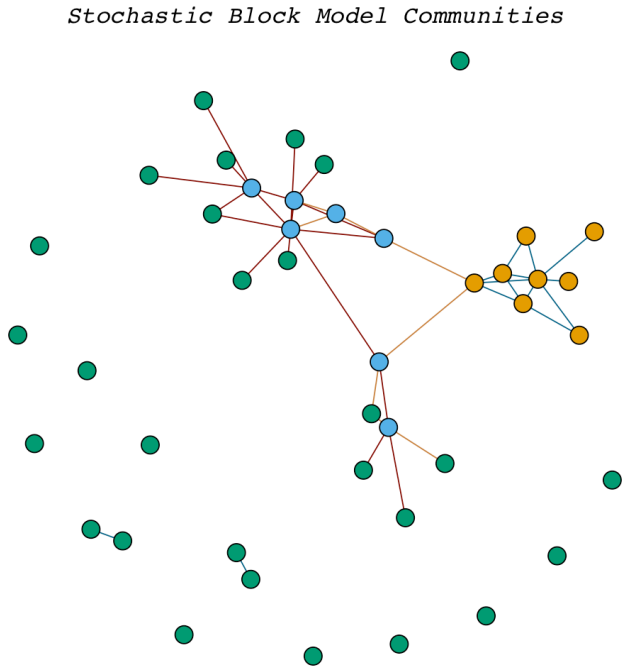


Figure 20: The resulting $Q = 3$ communities of the SBM.

Since the Stochastic Block Model is a soft clustering approach, meaning each faculty member belongs in each community with a probability ($p_{i,q} = \mathbb{P}(z_i = q)$), the community memberships of Figure 20 are those with the greatest $p_{i,q}$.

A crucial step after the estimation is to interpret the discovered communities. In the case at hand, the **green**

community consists of marginal faculty members, with none or at most 2 academic collaborations. The **blue community** are faculty members which are ``in the middle" and therefore cooperate both with each other and with faculty members of the green community. Lastly, the **yellow community** consists of only a subset of UNIPI faculty members which collaborate strongly with one another but not with many others.

It is also noteworthy that all members of the blue community are faculty members of AUEB. The only aspect remaining to study is the edge - forming probability matrix. When estimating a block model (either a stochastic or deterministic) the probability of a collaboration between two faculty members (v_i, v_j) is reduced to the probability of collaboration between any two faculty members of their communities. Mathematically, this is:

$$p_{i,j} = \mathbb{P}(Y_{i,j} = 1 \mid z_i = q, z_j = r) = \pi_{q,r}$$

Therefore, since $Q = 3$, the symmetric (undirected network) probability matrix of 3×3 dimensions is the following:

$$\begin{pmatrix} 0.419 & 0.038 & 0.000 \\ 0.038 & 0.461 & 0.074 \\ 0.000 & 0.074 & 0.003 \end{pmatrix}$$

The rows and columns correspond to: yellow, blue and green communities. The probability of two faculty members, both belonging to yellow community, to collaborate is 0.419, while the same for two faculty members of the blue community is 0.461. The probability of a ``yellow" faculty member to collaborate with a ``green" one is inferred to be 0, because there were no realized yellow $\leftrightarrow$ green edges in the network.

6.3.3 Latent Position Cluster Model

The Latent Position Cluster Model considers the latent positions of the vertices, $z_i, i = 1, \ldots, |V|$, to emerge from a finite mixture of $d-$ variate Normal distributions. Specifically:

$$z_i \sim \sum_{k=1}^{K} \pi_k \cdot N_d(\mu_k, \sigma_k^2 \cdot I_{d \cdot d})$$

It is a soft, model based clustering approach that infers (among other parameters) a posterior probability for each vertex to belong in each community, $\mathbb{P}(K_i = k \mid z_i, \mu_k, \ldots, \sigma_k^2), i = 1, \ldots, |V|$ and $k = 1, \ldots, K$. The estimation in R is the Bayesian performed through MCMC, as presented previously.

Before the estimation, the number of mixture components (communities), K, needs to be specified. Here, it will be specified as $K = 3$, so the resulting communities can be contrasted with those of the Stochastic Block Model.

In Figure 21, each point the latent position of a vertex in the $2-$ dimensional Euclidean space. The points depict a pie chart of the posterior probabilities of community membership for each faculty member. The three circles are the components of the finite mixture and the respective colored crosses are their centers $(\mu_k, k = 1, 2, 3)$.

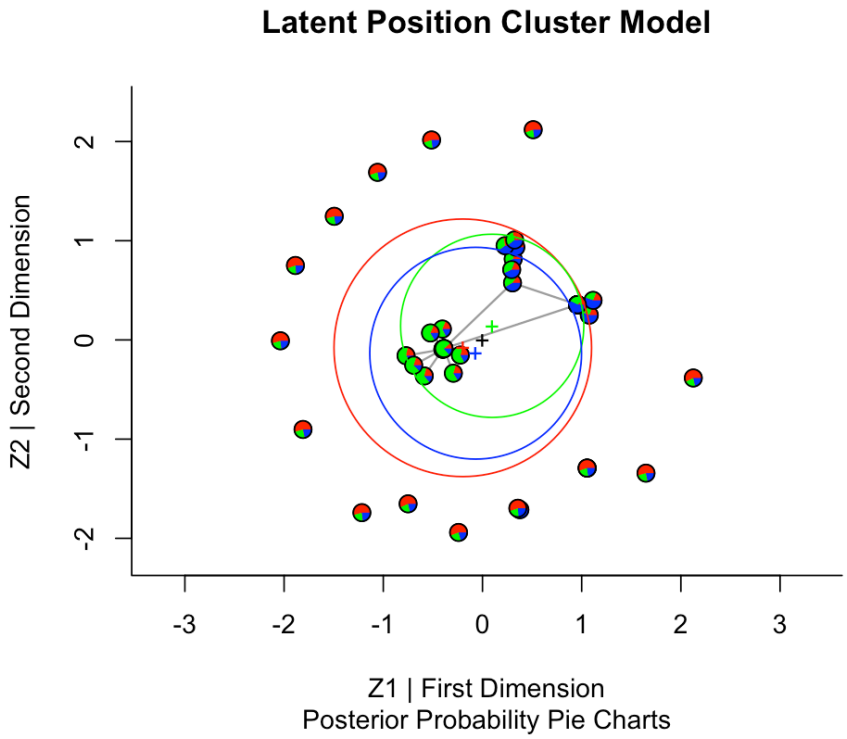


Figure 21: The plot with community membership probabilities of the LPCM.

In comparison, a similar plot with community membership pie charts for the Stochastic Block Model:

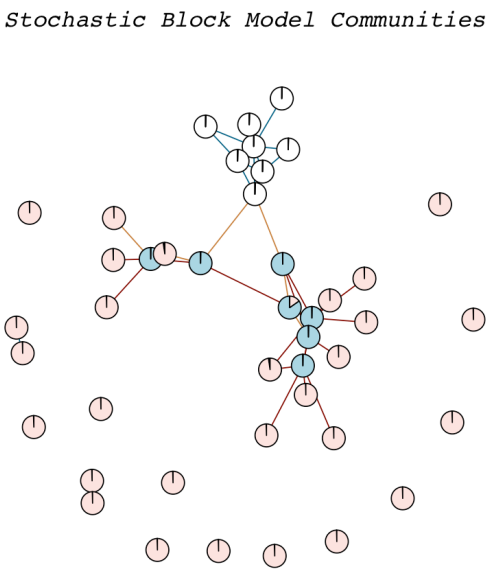


Figure 22: The plot with community membership probabilities for the SBM.

Interestingly, the community membership probabilities in the SBM are more defined than those of the

LPCM, meaning there is less uncertainty on the community a vertex belongs in. This is due to the fact that, in the LPCM the specification of the model suggests that the mixture components of the latent positions, z_i, are circular (in the $d = 2$ case), which is not supported in this particular application.

In order to obtain the final component memberships of the vertices' latent positions, a **maximum a - posteriori** rule needs be applied. The resulting communities can be studied in Figure 23.

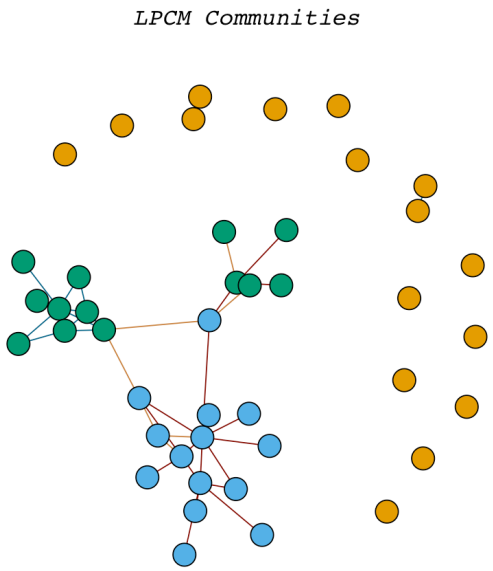


Figure 23: The resulting communities of the LPCM after MAP rule.

Not surprisingly, the latent community structure is different than that of the SBM. It should be noted that, the same color / label (e.g. blue) corresponds to different communities in the SBM and LPCM. In Figure 23, the blue community comprises almost entirely of AUEB members which collaborate with each other. The blue community are almost all UNIPI members which ``connect'' to the blue community academically only through two blue nodes. Lastly, the yellow community corresponds the disconnected nodes of both AUEB and UNIPI.

Community - detection, or clustering, methods can be evaluated internally, (e.g. to define the optimal number of clusters), and externally (e.g. to assess agreement between different approaches). The **Adjusted Rand Index** is a metric which quantifies the agreement in vertex community placement between two methods, while adjusting for the fact that between different methods indices referring to the ``same'' cluster vary.

The Adjusted Rand Index takes values close to 1 for methods that strongly agree and even negative values for methods whose agreement is very low. In this case, it is equal to 0.2726.

Therefore, the two approaches agree approximately 28% of the times in the placement of faculty members in communities, which is a **moderate** agreement.

A final comparison can be made, where the size of vertices is determined by the betweenness centrality.

Since the disconnected nodes have a betweenness centrality of 0, a correction of $+4$ to all vertices is applied.

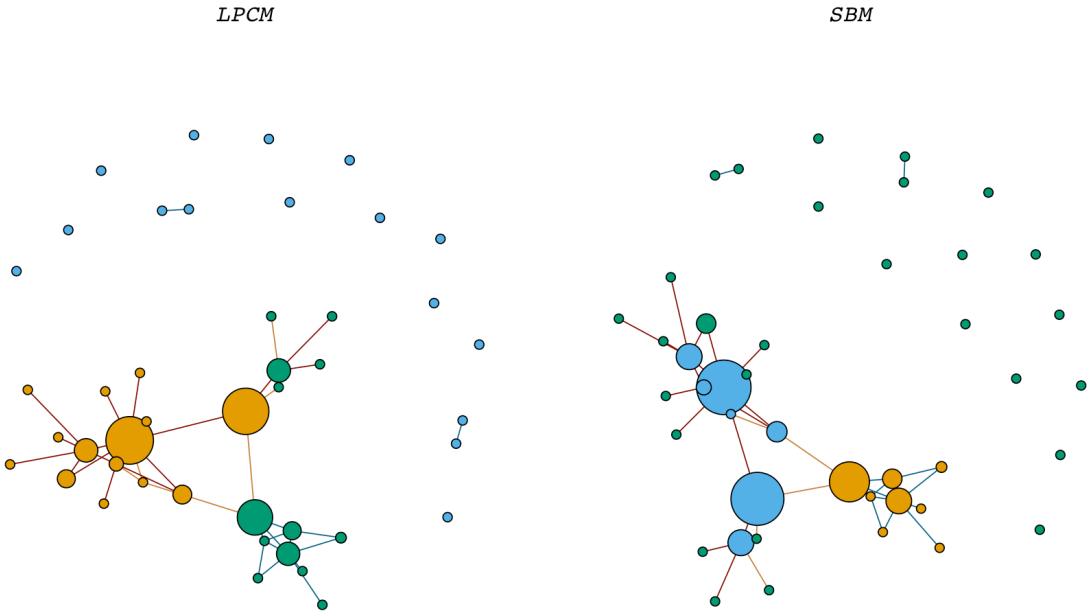


Figure 24: Side - by - side comparison of the resulting communities of SBM & LPCM.

What has been earlier implied regarding the importance of nodes with high betweenness centrality, can now be visually confirmed. Ignoring the community of disconnected faculty members, if the three nodes with highest betweenness centrality (largest points) were to vanish, the connection between the two collaborative communities (yellow and green - left & blue and yellow - right) would fragment.

7 Conclusions

This thesis serves as a survey for the prominent statistical methods for social network analysis. Various descriptive statistics, statistical models and community - detection methods were presented, each aiming to capture different network characteristics and dependencies. Some of the presented methods were then applied in the real, academic - collaboration dataset of Athens' two statistics departments. Through the application, some interesting insights about the real academic collaboration network were gained.

The two academic departments have very different collaboration dynamics. Specifically, the faculty members of AUEB collaborate much more closely with one another, compared to those of UNIPI. This is shown both through the density descriptive statistics, but also from the block models where the probability of academic collaboration is higher between two AUEB faculty members, than two of UNIPI. Generally, however, the probability of collaboration between two members of different departments was found to be significantly lower than of two of the same department. This is an indication of homophily in the network. Moreover, the vertex degree histograms of both networks had similar form. Both heavily leptokurtic and positively skewed. The latter means that most faculty members have 0, 1 or 2 inter - departmental collaborations, while only one of AUEB has 8 and one of UNIPI has 7, indicating preferential attachment. Lastly, three communities were inferred within the academic collaboration network, which proved to have interesting interpretation.

The application of methods in real data indicated certain practical challenges. To begin with, the ERGM models, while very powerful in utilizing the network's structure, are not scalable in terms of their estimation. It would be greatly beneficial to include as many network descriptive statistics and node - specific covariates as possible. However, these would need to be calculated each time an edge is altered in the MCMC - MLE estimation. For more complex network statistics, e.g. the number of $k-$ cores or $k-$ cliques, this is computationally expensive and oftentimes unfeasible. Additionally, the strong presence of disconnected nodes in the academic collaboration network rendered methods such as the hierarchical clustering, the closeness centrality, average path length and network diameter useless.

What was covered in this thesis should by no means considered exhaustive. It would be very interesting, but also more complicated, to collect and analyze data for a citations network. By definition, this is directed & weighted and thus directed - network specific methods can be used, conversely the PageRank centrality. For instance, the PageRank centrality could be compared to the prominent h - index and $i-10$ index for assessing a scientist's quality of work, by giving more weight to citations from important and citation - parsimonious scientists. Also, it would be interesting to study dynamic academic collaboration networks.

A Model Summary Tables

The model summary tables were conveniently created using `texreg` by Leifeld (2013) package in `R`.

	Erdős – Rényi
edges	$-3.07^{\cdots}$
	(0.16)
AIC	329.55
BIC	334.35
Log Likelihood	-163.77

$^{\cdots}p < 0.001$; $^{\cdot\cdot}p < 0.01$; $^{\cdot}p < 0.05$

Table 7: Erdős – Rényi through ERGM model summary.

	Block Model
edges	$-4.03^{\cdots}$
	(0.36)
nodematch.department.AUEB	$1.89^{\cdots}$
	(0.44)
nodematch.department.UNIPI	$1.10^{\cdot}$
	(0.45)
AIC	312.38
BIC	326.80
Log Likelihood	-153.19

$^{\cdots}p < 0.001$; $^{\cdot\cdot}p < 0.01$; $^{\cdot}p < 0.05$

Table 8: Block Model through ERGM model summary.

	LDM
(Intercept)	0.46
	$[-0.57; 1.58]$
BIC (Overall)	497.57
BIC (Likelihood)	235.42
BIC (Latent Positions)	262.15

Table 9: Latent Distance Model without covariates model summary.

	LDM
(Intercept)	-0.90
	$[-2.01; 0.29]$
nodematch.dpt.AUEB	$2.16^{\cdot}$
	$[1.13; 3.28]$
nodematch.dpt.UNIPI	$1.17^{\cdot}$
	$[0.12; 2.25]$
BIC (Overall)	474.45
BIC (Likelihood)	238.62
BIC (Latent Positions)	235.83

Table 10: Latent Distance Model with categorical covariate model summary.

	ERGM
mutual	$-0.88^{\cdots}$
	(0.25)
transitive	$-0.57^{\cdots}$
	(0.05)
AIC	1546.32
BIC	1557.32
Log Likelihood	-771.16

$^{\cdots}p < 0.001$; $^{\cdot\cdot}p < 0.01$; $^{\cdot}p < 0.05$

Table 11: ERGM for the simulated directed citations network model summary.

	LPCM
(Intercept)	0.18
	$[-0.79; 1.32]$
BIC (Overall)	463.18
BIC (Likelihood)	237.59
BIC (Latent Positions)	225.59

Table 12: Latent Position Cluster Model summary.

B Packages Used

The packages used in the application of methods to the real data.

R Package	Creator(s)	Use
`readxl`	Wickham et al. (2019)	Inserting `.xlsx` files into R.
`igraph`	Csardi (2013)	Network data wrangling and descriptive analytics.
`network`	Butts (2008)	Network data wrangling and descriptive analytics.
`ergm`	Hunter et al. (2008)	Estimation and goodness-of-fit for ERGMs.
`texreg`	Leifeld (2013)	Producing model summary tables.
`mclust`	Scrucca et al. (2023)	Computing the Adjusted Rand Index.
`blockmodels`	Leger (2016)	Estimating Stochastic Block Models.
`latentnet`	Krivitsky et al. (2015)	Estimating Latent Distance and Latent Position Cluster Models.
`intergraph`	Bojanowski (2023)	Supporting the switch between `igraph` and `network` data types.

References

Bojanowski, M. (2023). intergraph: Coercion routines for network data objects [Computer software manual]. (R package version 2.0-3)

Bron, C., & Kerbosch, J. (1973). Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, *16*(9), 575--577.

Butts, C. T. (2008). network: a package for managing relational data in r. *Journal of Statistical Software*, *24*(2). doi: 10.18637/jss.v024.i02

Casella, G., & George, E. I. (1992). Explaining the gibbs sampler. *The American Statistician*, *46*(3), 167--174.

Csardi, M. G. (2013). Package 'igraph'. -, *3*(09), 2013.

Fruchterman, T. M., & Reingold, E. M. (1991). Graph drawing by force-directed placement. *Software: Practice and experience*, *21*(11), 1129--1164.

Garlaschelli, D. (2009). The weighted random graph model. *New Journal of Physics*, *11*(7), 073005.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (1995). *Bayesian data analysis*. Chapman and Hall/CRC.

Goldenberg, A., Zheng, A. X., Fienberg, S. E., & Airoldi, E. M. (2010). A survey of statistical network models. *Foundations and Trends® in Machine Learning*, *2*(2), 129--233.

Handcock, M. S., Raftery, A. E., & Tantrum, J. M. (2007). Model-based clustering for social networks. *Journal of the Royal Statistical Society Series A: Statistics in Society*, *170*(2), 301--354.

Holland, P. W., & Leinhardt, S. (1981). An exponential family of probability distributions for directed graphs. *Journal of the american Statistical association*, *76*(373), 33--50.

Hunter, D. R., & Handcock, M. S. (2006). Inference in curved exponential family models for networks. *Journal of computational and graphical statistics*, *15*(3), 565--583.

Hunter, D. R., Handcock, M. S., Butts, C. T., Goodreau, S. M., & Morris, M. (2008). ergm: A package to fit, simulate and diagnose exponential-family models for networks. *Journal of Statistical Software*, *24*(3), 1--29. doi: 10.18637/jss.v024.i03

Kamada, T., & Kawai, S. (1989). An algorithm for drawing general undirected graphs. *Information processing letters*, *31*(1), 7--15.

Kaur, H., Rastelli, R., Friel, N., & Raftery, A. E. (2023). Latent position network models. *arXiv preprint arXiv:2304.02979*.

Kolaczyk, E. D., & Csárdi, G. (2014). *Statistical analysis of network data with r* (Vol. 65). Springer.

Krivitsky, P. N., Handcock, M. S., Shortreed, S. M., Tantrum, J., Hoff, P. D., Wang, L., … Krivitsky, M. P. N. (2015). Package 'latentnet'. -.

Leger, J.-B. (2016). Blockmodels: A r-package for estimating in latent block model and stochastic block model, with various probability functions, with or without covariates. *arXiv preprint arXiv:1602.07587*.

Leifeld, P. (2013). texreg: Conversion of statistical model output in R to LaTeX and HTML tables. *Journal of Statistical Software*, *55*(8), 1--24.

Liao, L., Jia, Z., & Deng, Y. (2019). Coarse-graining method based on hierarchical clustering on complex networks. *Communications and Network*, *11*(1), 21--34.

Robert, C. P., & Casella, G. (2004). The metropolis—hastings algorithm. *Monte Carlo statistical methods*, 267--320.

Scrucca, L., Fraley, C., Murphy, T. B., & Raftery, A. E. (2023). *Model-based clustering, classification, and density estimation using mclust in R*. Chapman and Hall/CRC. doi: 10.1201/9781003277965

van Der Pol, J. (2019). Introduction to network modeling using exponential random graph models (ergm): theory and an application using r-project. *Computational Economics*, *54*(3), 845--875.

Wickham, H., Bryan, J., Kalicinski, M., Valery, K., Leitienne, C., Colbert, B., … Bryan, M. J. (2019). Package 'readxl'. *Version*, *13*, 1.