

Modelling Dynamic Networks with Centrality-Based Logistic Regression

by

Nikolay Kulmatitskiy

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Master of Mathematics
in
Statistics

Waterloo, Ontario, Canada, 2011

© Nikolay Kulmatitskiy 2011

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Statistical analysis of network data is an active field of study, in which researchers investigate graph-theoretic concepts and various probability models that explain the behaviour of real networks. This thesis attempts to combine two of these concepts: an exponential random graph and a centrality index. Exponential random graphs comprise the most useful class of probability models for network data. These models often require the assumption of a complex dependence structure, which creates certain difficulties in the estimation of unknown model parameters. However, in the context of dynamic networks the exponential random graph model provides the opportunity to incorporate a complex network structure such as centrality without the usual drawbacks associated with parameter estimation. The thesis employs this idea by proposing probability models that are equivalent to the logistic regression models and that can be used to explain behaviour of both static and dynamic networks.

Acknowledgements

I would like to thank Professor Shojaeddin Chenouri for suggesting that I do research in dynamic random graphs, encouraging me to write this Thesis, and organizing a seminar group devoted to the study of network data analysis. I would also like to thank the members of that group: Shojaedding Chenouri, Greg D’Cunha, Spencer Wheatley, and Melissa Lu; for the numerous discussions that helped me to understand and formulate many important concepts that I discuss in this thesis.

I would like to say special thanks to the committee members Cristopher Small and Chris Groendyke, who thoroughly read and commented on the thesis, suggesting numerous improvements to the structure, the contents, and the writing style of the thesis. The present submission contains the improvements and clarifications that resulted from their contributions.

Table of Contents

1	Introduction	1
2	Random Graphs	7
2.1	Preliminaries	7
2.2	Exponential Models for Random Graphs	12
2.3	Some Examples	19
2.4	Estimation	22
2.5	Dynamic Random Graphs	34
3	Centrality Indices	41
3.1	Introduction to Centrality	41
3.2	Eigenvector Centrality	45
3.3	Geodesic Centralities	48
3.4	Centrality Indices and the Lawyers Network	56
4	Balanced Potential Models	58
4.1	Balanced Potential Model	59
4.2	Balanced Centrality Markov Chains	68
5	Directions of Future Research	78
	APPENDICES	80

A	BCMC Simulations	81
	References	89

Chapter 1

Introduction

Many real world systems take the form of a network, for example: the Internet networks in the field of computer science, the neural and metabolic networks in the field of biology, social and organizational networks, such as acquaintances, e-mail communications, and scientific collaborations. There are many more examples, but what unites them altogether is the presence of *binary relationships* between the elements of a network. The terminology concerning these networks can be modified in various ways. Thus, the elements of a network are often called *vertices*, *nodes*, *actors*, or *agents*. The relationships between these elements are often called *edges*, *ties*, *links*, *arcs*, or *interactions*. When the relationships in a given network are observed and recorded, the resulting dataset is commonly referred to as either *network data* or *relational data*.

Recent decades have witnessed a surge in network research, as scientists have realized that *graph theory* provides powerful mathematical tools for the construction of methodologies that describe, explain, and simulate the various aspects of network “behaviour”. More concretely, network research is occupied with the regular *patterns* in the observed relationships. The presence of these patterns in the data is commonly referred to as *structure*, and the quantities that measure the structure are called *structural* quantities.

For a more detailed introduction to the topic of general network research, the reader is advised to read Newman (2003), which also discusses several examples of large real networks. This thesis is primarily related to the *statistical* analysis of network data, an area of network research specifically devoted to the probabilistic models that help to understand network structure. To give the reader an idea about the kinds of statistical questions that a network analyst may ask about a relational dataset, we provide the following example, which will be referenced several times throughout the thesis.

In 2001, Emmanuel Lazega published an extensive study (Lazega (2001)) of the social interactions among 71 lawyers in a New England law firm. Figure 1.1 demonstrates the network of collaborations among these lawyers.¹ The lawyers are labelled as $0, 1, 2, \dots, 71$, and two lawyers i and j are connected with a line if they ever worked together on the same case. A useful way to code this information is to define a variable x_{ij} and set $x_{ij} = 1$ if lawyer i collaborated with lawyer j and set $x_{ij} = 0$ otherwise. In addition to the collaboration relationship, various other attributes were recorded for each lawyer, such as his or her status in the firm (partner or not), seniority in the firm, cumulative work experience in years, and other attributes.

Given this dataset, consider the following questions.

- Are senior lawyers more likely to collaborate with each other than with junior lawyers? If the seniority does not affect collaborations, is there another lawyer attribute that *does* affect the chances of collaboration?
- If we label the lawyers who have the most collaborations as “stars”, do other lawyers become more inclined to collaborate with the stars, or is the “star” status irrelevant to their choices? In other words, does the status of a “star” play a greater role in attracting collaborations compared to other factors such as seniority and experience?
- If lawyer i collaborated with lawyer j , and lawyer j collaborated with lawyer k , are i and k more likely to collaborate with each other? In other words, to what extent is the *transitive property* present in this network? In network terminology, the three lawyers i, j, k are said to form a *transitive triad* if and only if $x_{ij}x_{jk}x_{ki} = 1$, i.e., all three of them have collaborated with each other. A *3-cycle* is an alternative term for such a triad. If the 3-cycles are abundant then the lawyers in the network can be categorized into several groups, or *cliques*, within which the lawyers collaborate extensively while having few or even no relationships between the cliques. A statistician then might ask whether these groups have formed entirely by chance or, on the contrary, systematically as the result of some unapparent social classification within the firm.

The fundamental problem raised by the above questions is this: do lawyers initiate collaborations independently of other ties in the network (i.e., based on individual characteristics

¹In addition to work-related collaboration, Lazega observed other relationships such as friendships outside work and solicitation of advice. We will only use the collaboration data in this thesis.

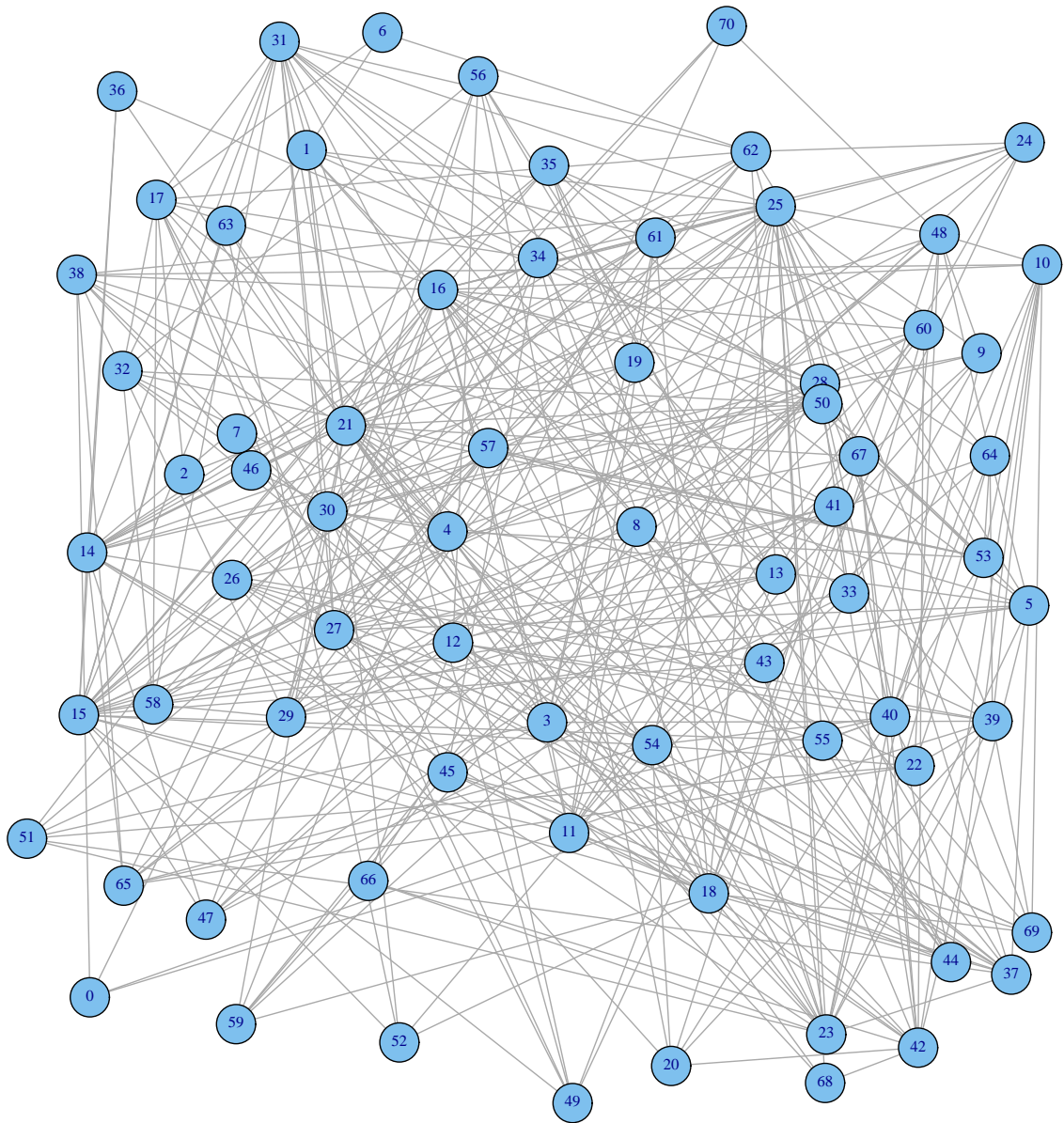


Figure 1.1: Lawyers Collaboration Network. In this visualization the circles represent lawyers, labelled $0, 1, 2, \dots, 71$, and the lines represent collaborations between the lawyers.

such as seniority) or do they take into account the observed structural patterns (e.g., which lawyers are “stars”; or who collaborates with whom)?

Let us further elaborate on the peculiar nature of the preceding example. On the one hand, we have a finite set of binary observations with several non-random explanatory covariates. Thus we could apply the familiar methods of binary data analysis, e.g., as laid out in Cox (1970). On the other hand, network analysts often report the presence of statistically significant structural patterns (such as the presence of “stars” or the abundance of 3-cycles) in the data, suggesting that these structural patterns should be used as explanatory covariates in the model. Note that these structural covariates are functions of the data, i.e., they are themselves random, and, moreover, they introduce complex statistical dependencies between the binary variables that represent edges in the network. Now, if we limit ourselves to the first approach (i.e., by ignoring the structural patterns), then we have to provide a probability model (e.g., logistic regression) for each individual pair of actors, and the model for the joint observations will follow with the assumption of statistical independence among the edges. However, if we accept the second approach (i.e., by taking into account the structural dependencies), then we have to provide a probability model for the network *as a whole*, since the structural covariates typically involve several relationships at once, and thus the “pair by pair” assignment of probabilities is no longer relevant.

If we decide to account for the structural patterns in the network, we get to choose from a large variety of structural attributes to be used as covariates in the model. Table 1.1 lists some of the commonly used attributes, expressed as functions of the relationships in the network (assuming the notation used in our previous example). To a person with no experience in network research, some of these attributes might make little sense on the first encounter. But for a social network analyst, for example, all of these attributes represent statistics of great interest and familiarity. The number of 3-cycles, for instance, represents the extent to which a given social network follows the principle “a friend of my friend is also my friend”.

Of course, with most relational datasets it is hard to tell a priori whether the use of non-structural covariates by themselves leads to adequate models, or whether the inclusion of structural covariates would be more appropriate. In other words, it is debatable whether the inclusion of structural covariates (such as those in Table 1.1) leads to more “realistic” models, compared to the alternative approach of only using non-structural network covariates (e.g., age, gender, income). The truth will differ from one case to another,

Attribute	Computation
Number of ties	$\sum_i \sum_j x_{ij}$
Number of reciprocal pairs (if the relationship is directed)	$\sum_{i < j} x_{ij} x_{ji}$
Number of 3-cycles (if the relationship is undirected)	$\sum_i \sum_j \sum_k x_{ij} x_{jk} x_{ik}$
Number of 3-cycles (if the relationship is directed)	$\sum_i \sum_j \sum_k x_{ij} x_{jk} x_{ki}$
Number of 2-stars*	$\sum_i \sum_j \sum_k x_{ij} x_{ik}$
Maximum degree	$\max_i \sum_j x_{ij}$
*Here the term “star” has a meaning different from the one mentioned in the main discussion.	

Table 1.1: Examples of Structural Attributes of a Relational Dataset

and some generic experience-based insights into the properties of the network can be quite helpful in deciding between the two approaches. However, what can be said with confidence is this. Once we depart from the non-structural analysis of binary data and start using the graph-theoretic structural covariates in our models, the techniques for parameter estimation become less intuitive, more computationally intensive, and endowed with fewer “good” theoretical properties than their “classical” counterparts. A detailed discussion of these matters constitutes an important segment in this thesis—specifically Section 2.4.

There is an opportunity to account for structural patterns in our models while still employing the familiar methods of binary data analysis. However, this opportunity only arises in the context of *dynamic* (or, *sequential*; or, *longitudinal*) network data, where we observe “snapshots” of the network at different points in time. For example, the lawyers network data could be assumed to represent the collaborations during a specific month, say January, while in reality the collaborations may change in the subsequent months. Essentially, we have a stochastic process $\{X_t : t = 1, 2, \dots\}$, in which the process state X_t at time t constitutes a separate network dataset, and we typically choose to model this process as a Markov chain, allowing us to predict the state of the network at time $t+1$ only on the basis of its present state (at time t). Now, the idea here is to have the structural attributes of the state at time t play the role of *non*-structural explanatory covariates of the state at time $t+1$. This approach leads to a satisfactory compromise, which means that we may propose more “realistic” models by using the structural patterns of the network, but we are not prevented from using the familiar methods of binary data analysis, in particular the logistic regression. This idea will be exploited in Chapter 4.

This thesis discusses the use of a special class of structural attributes called the *centrality indices*. Roughly speaking, a centrality index aims at quantifying the intuitive idea that

some network nodes are more important, or more “central”, than others. The “stars” in the lawyers network can be viewed as central because they are the nodes with the most connections. However there are popular ways to define centrality other than the number of connections that a tie holds. Combining centrality indices with our “compromise” approach to the modelling of dynamic networks, we propose a new class of models called the *Balanced Centrality Markov Chains* (BCMCs). In this thesis, we study their behaviour using simulations. Essentially BCMCs constitute a special case of the logistic regression, and this fact explains the title of the thesis.

We also propose a more general class of models called the *Balanced Potential Models* (BPMs) to be used with both static and dynamic networks in the presence of non-structural covariates. In fact, we present BCMCs as an elaboration on the BPMs.

The rest of the thesis is structured as follows.

- In Chapter 2 we briefly explain the fundamental concepts related to a *random graph* and then provide a detailed discussion of the *exponential random graph* models, which have proven to be quite useful in many areas of application. An important section is devoted to the estimation methods for exponential random graphs, where we reveal the common difficulties and problems associated with doing estimation for models that employ structural covariates. At the end, we discuss the concept of a dynamic random graph.
- Chapter 3 is entirely devoted to the concept of a *centrality index*.
- Chapter 4 begins with a thorough discussion of our Balanced Potential Models, and then combines them with the ideas from chapters 2 and 3, resulting in Balanced Centrality Markov Chain models for dynamic random graphs.

The thesis concludes in Chapter 5 with notes on the directions of future research.

Chapter 2

Random Graphs

2.1 Preliminaries

An important mathematical object in this thesis is a *graph*, which is used to represent a network. The term “graph” is synonymous with the term “network”, although some authors use the former to refer to the mathematical objects and the latter to refer to the physical situations of interest. This thesis is not strict about such a distinction; the reader will notice on occasion that both terms are used interchangeably.

Although it is not the purpose of the present section to cover yet again the basics of graph theory (with which, we assume, the reader is familiar), some repetition is inevitable because we require that all the terminology is established upfront and used with consistency throughout the thesis.

Formally, a graph G is an object that is comprised of two components:

1. an arbitrary finite set V , whose elements are called *vertices*, enumerated and represented by the integers $1, 2, \dots, N$; and
2. a set E called the *edge set* that is either entirely comprised of ordered pairs (i, j) or entirely comprised of unordered pairs $\{i, j\}$, where $1 \leq i, j \leq N$ and $i \neq j$.

A common notation is

$$G = (V, E).$$

The pairs in E are called *edges*. If the edges in E are ordered, we call them *directed edges*, and we say that the graph G is *directed*. Similarly, if the edges in E are unordered,

they are called *undirected edges* and we say that the graph G is *undirected*. It will be very convenient to apply the term *edge* to any hypothetical ordered or unordered pair of vertices (regardless of whether or not this pair actually belongs to any particular graph G).

In most of the upcoming discussions, we will not want to commit ourselves to either the class of directed graphs or the class of undirected graphs. For this reason, it is convenient to employ a universal notation e_{ij} for an edge, so that

$$e_{ij} = \begin{cases} (i, j) & \text{when ordered edges are implied by the context,} \\ \{i, j\} & \text{when unordered edges are implied.} \end{cases}$$

If the discussion is about undirected graphs, then clearly $e_{ij} = e_{ji}$ for all i and j in V .

Graphs are visualized by drawing vertices as points, and edges as lines, as illustrated by Figure 2.1b, which is a graphical representation of the graph formally defined in Figure 2.1a.

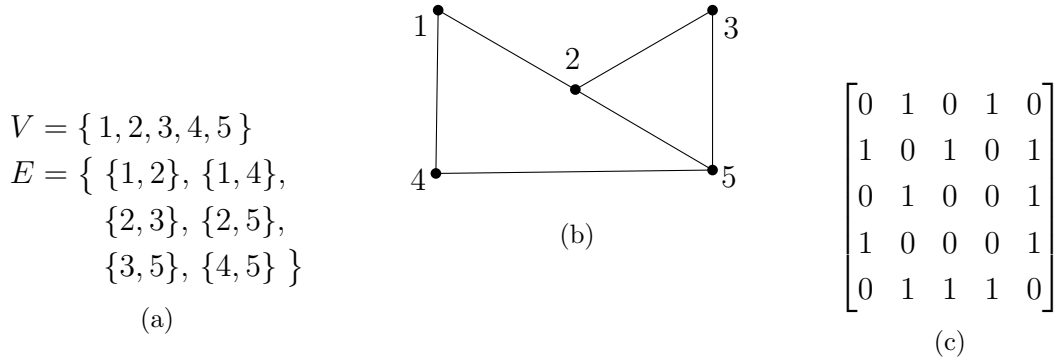


Figure 2.1: Three Representations of a Graph.

If V contains N vertices, then the maximum number of edges that a graph may have is easily shown to be

$$M_{und} := \binom{N}{2} \text{ for an undirected graph,}$$

$$M_{dir} := N(N - 1) \text{ for a directed graph.}$$

The *density* of a graph $G = (V, E)$ is defined as the ratio of the number of edges in E to the maximum possible number of edges. Thus, the density of an undirected graph is $|E|/M_{und}$, and the density of a directed graph is $|E|/M_{dir}$.

It is convenient to represent graphs as matrices, but first we have to define the notion of *adjacency*. We say that vertex i is *adjacent* to vertex j if the edge e_{ij} belongs to E . We then define binary variables x_{ij} , called *edge* (or *adjacency*) *indicators*, by

$$x_{ij} := \begin{cases} 1 & \text{if } e_{ij} \in E, \\ 0 & \text{otherwise.} \end{cases}$$

We always assume that no vertex i can be adjacent to itself, meaning that $x_{ii} = 0$ for all i in V . We can now define the *adjacency matrix* \mathbf{x} of a graph G as

$$\mathbf{x} := \begin{bmatrix} 0 & x_{12} & x_{13} & \cdots & x_{1N} \\ x_{21} & 0 & x_{23} & \cdots & x_{2N} \\ x_{31} & x_{32} & 0 & \cdots & x_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & x_{N3} & \cdots & 0 \end{bmatrix}.$$

We will use lowercase boldface letters (\mathbf{x}) to represent matrices because we will need to distinguish a given adjacency matrix \mathbf{x} from a *random* adjacency matrix, which we will denote with an upper-case boldface letter (\mathbf{X}). Note that undirected graphs have symmetric adjacency matrices, as shown in Figure 2.1c. We will most often refer to graphs by their adjacency matrices instead of using the formal definition. Thus, for example, we will say “suppose \mathbf{x} is a graph” instead of “suppose G is a graph”. We agree to use $E(\mathbf{x})$ as an alternative notation for edge set of the graph \mathbf{x} .

Let us emphasize one more time the distinction between the symbols e_{ij} and x_{ij} . We use e_{ij} to denote an edge (i, j) or $\{i, j\}$ in the abstract, without affiliation to any particular edge set E . On the other hand, the edge indicator x_{ij} is a binary variable that is tied to a specific graph \mathbf{x} with the specific edge set $E(\mathbf{x})$.

For any vertex i in an undirected graph G , the *degree of i* is the number of edges e_{ij} that belong to E . It can be expressed in terms of edge indicators as

$$\deg(i) := \sum_{j=1}^N x_{ij} = x_{i+}.$$

The *neighbourhood* $N(i)$ of a vertex i is the set of all vertices that are adjacent to i :

$$N(i) := \{ j : x_{ij} = 1 \}.$$

Note that a vertex’s degree is equal to the size of its neighbourhood.

For a directed graph, the notions of degree and neighbourhood are modified to account for edge directions as follows. The *out-degree* of i is the number of edges e_{ij} in E emanating from i , while the *in-degree* of j is the number of edges e_{ij} in E arriving at j . In terms of edge indicators:

$$\deg_{out}(i) := \sum_{j=1}^N x_{ij} = x_{i+}, \quad \deg_{in}(j) := \sum_{i=1}^N x_{ij} = x_{+j}.$$

By analogy, we define the *out-neighbourhood* (or *forward-neighbourhood*) $N_{out}(i)$ of i and the *in-neighbourhood* (or *backward-neighbourhood*) $N_{in}(j)$ of j as

$$N_{out}(i) := \{j : x_{ij} = 1\}, \quad N_{in}(j) := \{i : x_{ij} = 1\}.$$

If V' is a subset of the vertex set V and E' is defined as

$$E' = \{e_{ij} \in E : i, j \in V'\},$$

then the graph $G' = (V', E')$ is said to be the *subgraph of G induced by V'* .

2.1.1 Defining Random Graphs

To model network behaviour using probability, we first need to decide what our sample space is. We may choose one of the two approaches: either assume that the vertices are fixed while the edges are uncertain or assume that both vertices and edges are uncertain. A probability model can be constructed for either case. This thesis, however, is focused on the models that require a fixed set of vertices.

Let V be a fixed set of $N \geq 2$ vertices and suppose that Ω_{und} is the set of all possible undirected graphs on V . There are $2^{\binom{N}{2}}$ such graphs in total. Notice that Ω_{und} is equivalent to the set of all symmetric matrices with binary entries and zeros on the main diagonal. Assume that the set Ω_{und} is the sample space of a random experiment. For example, Ω_{und} is the collection of all possible friendship combinations in a group of N classmates. Again, we represent the graphs in Ω_{und} by their adjacency matrices. Thus, for example, we would write $\mathbf{x} \in \Omega_{und}$, where \mathbf{x} is the symmetric matrix in Figure 2.1c. (In this case $N = 5$.)

We denote by \mathbf{X} the outcome of the random experiment and we assign probabilities $P(\mathbf{X} = \mathbf{x})$ to each graph \mathbf{x} in Ω_{und} . The uncertain outcome \mathbf{X} is called an *undirected*

random graph. Note that \mathbf{X} can also be viewed as a random adjacency matrix. In fact, a useful way to think about \mathbf{X} is as of the collection

$$\{ X_{ij} : 1 \leq i < j \leq N \} \quad (1)$$

of $\binom{N}{2}$ binary random variables X_{ij} , where each X_{ij} is an edge indicator of the random graph.

By analogy, we define a *directed random graph* \mathbf{X} to range in the sample space Ω_{dir} , which is the set of all directed graphs on V . There are $2^{N(N-1)}$ such graphs in total. Again, the random adjacency matrix \mathbf{X} can be viewed as the collection

$$\{ X_{ij} : 1 \leq i, j \leq N, i \neq j \} \quad (2)$$

of $N(N-1)$ random edge indicators. Note the distinction between (1) and (2), reflecting the main difference between directed and undirected random graphs.

As mentioned earlier, we will often not want to commit to either directed or undirected graphs. Therefore, whenever we indicate the sample space of a random graph with the unadorned notation Ω , we will be implying that the random graph could be either directed ($\Omega = \Omega_{dir}$) or undirected ($\Omega = \Omega_{und}$), the actual case being non-essential to the discussion.

By the *inclusion rate* of an edge e_{ij} we mean the probability

$$p_{ij} := P(X_{ij} = 1).$$

By *complete edge independence* we mean the situation in which for any collection

$$\{e_{i_1 j_1}, \dots, e_{i_m j_m}\}$$

of distinct edges we have

$$\begin{aligned} P(X_{i_1 j_1} = x_{i_1 j_1}, \dots, X_{i_m j_m} = x_{i_m j_m}) &= \prod_{k=1}^m P(X_{i_k j_k} = x_{i_k j_k}) \\ &= \prod_{k=1}^m p_{i_k j_k}^{x_{i_k j_k}} (1 - p_{i_k j_k})^{1-x_{i_k j_k}}. \end{aligned}$$

For an extensive survey of random graph models, the reader may refer to Chapter 6 in Kolaczyk (2009). Our main focus will be on the exponential models for random graphs.

2.2 Exponential Models for Random Graphs

In general, an arbitrary discrete random variable X is said to belong to the *exponential family* if its probability mass function has the form

$$P(X = x; \theta) = \exp \left(\sum_{k=1}^r c_k(\theta) T_k(x) + c_0(\theta) + T_0(x) \right), \quad x \in \Omega, \quad \theta \in \Theta.$$

Alternatively, we could write

$$P(X = x; \theta) = \exp \left(\sum_{k=1}^r c_k(\theta) T_k(x) + T_0(x) \right) K(\theta), \quad x \in \Omega, \quad \theta \in \Theta, \quad (3)$$

where

$$K(\theta) := \exp(c_0(\theta)) = \left(\sum_{\mathbf{x} \in \Omega} \exp \left(\sum_{k=1}^r c_k(\theta) T_k(\mathbf{x}) + T_0(\mathbf{x}) \right) \right)^{-1}$$

is the *normalizing constant*, which ensures that the probabilities sum up to one.

Let \mathbf{X} be the adjacency matrix of a random graph (either directed or undirected) on N vertices. Suppose that the probability mass function of \mathbf{X} satisfies (3) for some parameter value θ that lies in an arbitrary parameter space Θ . Then \mathbf{X} is said to be an *exponential random graph* (ERG). The functions $T_k(\mathbf{X})$ are called *covariates* of the model. A covariate $T_k(\mathbf{X})$ can be one of the following types.

1. Covariate $T_k(\mathbf{X})$ is called *endogenous* (or *structural*) if it directly depends on the structural properties of the random graph \mathbf{X} , such as its density, degree distribution, number of 3-cycles, and many others.
2. Covariate $T_k(\mathbf{X})$ is called *exogenous* if it is not a function of the graph's structure, but a function of externally supplied data, such as edge attributes (e.g., weights) or vertex attributes (e.g., gender, income, geographic position).
3. Covariate $T_k(\mathbf{X})$ is *mixed* if it involves both structural and external data.

Note that, since the range of the function $\exp(\cdot)$ is positive, we have $P(\mathbf{X} = \mathbf{x}) > 0$ for all \mathbf{x} . In other words, every graph \mathbf{x} in Ω is probable. Thus the only way to have improbable graphs is to exclude these graphs from the sample space altogether, i.e., by assuming a smaller sample space $\Omega_1 \subsetneq \Omega$. To make this change, we are only required to recalculate the normalizing constant $K(\theta)$.

Using the vectors

$$\mathbf{c}(\theta) := (c_1(\theta), c_2(\theta), \dots, c_r(\theta)) \text{ and } \mathbf{T}(\mathbf{x}) := (T_1(\mathbf{x}), T_2(\mathbf{x}), \dots, T_r(\mathbf{x})),$$

we can rewrite the probability mass function of \mathbf{X} more compactly as

$$P(\mathbf{X} = \mathbf{x}; \theta) = \exp(\mathbf{c}(\theta)^T \mathbf{T}(\mathbf{x}) + T_0(\mathbf{x})) K(\theta), \quad x \in \Omega, \quad \theta \in \Theta. \quad (4)$$

As a warning, we note that, in general, the real-valued functions $c_k(\theta)$ ($k = 1, \dots, r$) may be functionally dependent. Therefore it would be wrong to assume that $\mathbf{c}(\theta)$ always ranges in the whole space \mathbb{R}^r . In fact, the range of \mathbf{c} might be constrained to a *subset* of \mathbb{R}^r . This subset

$$\{\mathbf{c}(\theta) \in \mathbb{R}^r : \theta \in \Theta\}$$

is called the *natural parameter space* of the exponential model.

2.2.1 The Log-linear Model

We now discuss a special case of (3). Letting $\Theta = \mathbb{R}^r$, we agree to denote $\boldsymbol{\theta}$ by $\boldsymbol{\beta}$ and let $\mathbf{c}(\boldsymbol{\beta}) = \boldsymbol{\beta}$. The probability mass function becomes

$$P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta}) = \exp\left(\sum_{k=1}^r \beta_k T_k(\mathbf{x}) + T_0(\mathbf{x})\right) K(\boldsymbol{\beta}), \quad \mathbf{x} \in \Omega. \quad (5)$$

We see that the probabilities depend on the linear combination of r covariates and r independently assigned parameters. Such a model defines a *log-linear* random graph.

The log-linear model (5) provides significant flexibility in the analysis of random graphs because any “candidate” collection $\{T_k(\mathbf{x})\}$ of covariates generates a model that explains the behaviour of a random graph based on the individual effects of $T_k(\mathbf{x})$. These effects are multiplicative in the following sense. Including a covariate $T_k(\mathbf{x})$ in the log-linear model is equivalent to adding a multiplicative factor $\gamma_k^{T_k(\mathbf{x})} > 0$ ($\gamma_0 = e$, $\gamma_k = e^{\beta_k}$, $k = 1, \dots, r$) to the probability $P(\mathbf{X} = \mathbf{x})$, which then can be written as the product

$$P(\mathbf{X} = \mathbf{x}, \boldsymbol{\gamma}) = e^{T_0(\mathbf{x})} \gamma_1^{T_1(\mathbf{x})} \gamma_2^{T_2(\mathbf{x})} \dots \gamma_r^{T_r(\mathbf{x})} K(\boldsymbol{\gamma}) \quad (6)$$

of all such factors (and the normalizing constant).

Expression (6) allows global as well as local interpretation of the parameters β_k . We will return to these types of interpretation in the forthcoming subsections. At the moment,

let us discuss some technical details that may help us distinguish log-linear graphs from ERGs that are not log-linear. In the definition (5), the natural parameter space is just $\Theta = \mathbb{R}^r$. Being orthogonal projections, the functions $c_k(\boldsymbol{\beta}) = \beta_k$ are linearly and functionally independent¹. We thus see that, with a log-linear model, it is possible to individually change any β_k while leaving other β_j 's fixed. In a general ERG model, however, changing one $c_k(\boldsymbol{\theta})$ may require changing some other $c_j(\boldsymbol{\theta})$'s at the same time.

Suppose that the covariates $T_k(\cdot)$ are linearly dependent, i.e., there exists m ($1 \leq m \leq r$) such that

$$T_m(\mathbf{x}) = \sum_{k \neq m} a_k T_k(\mathbf{x}) \text{ for all } \mathbf{x} \in \Omega.$$

Then the expression in the exponent in (5) can be rearranged as

$$\sum_{k \neq m} (\beta_k + \beta_m a_k) T_k(\mathbf{x}) + T_0(\mathbf{x}) = \mathbf{c}(\boldsymbol{\beta})^T \mathbf{T}^*(\mathbf{x}) + T_0(\mathbf{x}), \quad (7)$$

where the dimensions of $\mathbf{c}(\boldsymbol{\beta})$ and $\mathbf{T}^*(\mathbf{x})$ are $r - 1$. In this alternative form the functions $c_k(\boldsymbol{\beta})$ are functionally dependent, but the probabilities are the same as before. Thus we have a model that is equivalent to a log-linear model but does not involve a linear combination of covariates with independently assigned parameters. Notice, however, that $\mathbf{c}(\boldsymbol{\beta})$ is a linear transformation from the space $\Theta = \mathbb{R}^r$ to the space \mathbb{R}^{r-1} , so that $\mathbf{c}(\boldsymbol{\beta}) = L\boldsymbol{\beta}$ for some $(r - 1) \times r$ matrix L . We can then write

$$P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta}) = \exp(\boldsymbol{\beta}^T L^T \mathbf{T}^*(\mathbf{x}) + T_0(\mathbf{x})) K(\boldsymbol{\beta}),$$

confirming the relationship $\mathbf{T}(\mathbf{x}) = L^T \mathbf{T}^*(\mathbf{x})$ between the original covariate vector $\mathbf{T}(\mathbf{x})$ and the one obtained in (7). More importantly, the above equation brings us to the following conclusion. If $\mathbf{c}(\boldsymbol{\beta})$ is the result of an arbitrary linear transformation L acting on the space Θ , then the exponential random graph can be made log-linear by applying the transpose transformation L^T to the covariate vector. In the original definition of a log-linear random graph, the linear operator was taken to be the identity operator I .

Now suppose to the contrary of the previous case that the function \mathbf{c} taking $\boldsymbol{\beta} \in \mathbb{R}^r$ to $\mathbf{c}(\boldsymbol{\beta}) \in \mathbb{R}^p$ is *not* linear in its argument $\boldsymbol{\beta}$. Again, some of the functions $c_k(\boldsymbol{\beta})$ may be functionally dependent, but in this case we can not modify the model to make it log-linear.

¹A set $\{f_1, \dots, f_k\}$ of functions is called *functionally dependent* if they satisfy some functional equation

$$F(f_1, \dots, f_k) \equiv 0.$$

For example, it could be the case that the natural parameter space $\{ \mathbf{c}(\boldsymbol{\beta}) \in \mathbb{R}^p : \boldsymbol{\beta} \in \mathbb{R}^r \}$ is a submanifold of \mathbb{R}^p , $p > r$. The resulting exponential model is called a *curved exponential model*, since there are non-linear constraints on the values $c_k(\boldsymbol{\beta})$. For the analysis of curved exponential random graph models, refer to Hunter (2007).

To summarize, the natural parameter space of a log-linear random graph must be a linear space. In the literature on exponential random graph models, most of the papers deal with log-linear models exclusively, and we will assume the same restriction in what follows. It is worth noting that in the literature on social network analysis log-linear random graphs are very often called *p*-models*; this terminology originates in the paper by Anderson et al. (1999).

2.2.2 Global Interpretation

By the *global* interpretation we mean the description of how the differences in covariates T_k as well as the changes in the parameters β_k affect the differences in the probabilities of any two graphs \mathbf{x} and \mathbf{y} . In particular, suppose that \mathbf{x} and \mathbf{y} are two graphs in Ω and let

$$\delta_k(\mathbf{x}, \mathbf{y}) := T_k(\mathbf{x}) - T_k(\mathbf{y}), \quad k = 0, 1, \dots, r,$$

so that

$$\frac{P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta})}{P(\mathbf{X} = \mathbf{y}; \boldsymbol{\beta})} = e^{\delta_0(\mathbf{x}, \mathbf{y})} \gamma_1^{\delta_1(\mathbf{x}, \mathbf{y})} \gamma_2^{\delta_2(\mathbf{x}, \mathbf{y})} \dots \gamma_r^{\delta_r(\mathbf{x}, \mathbf{y})}. \quad (8)$$

Now, suppose for simplicity that graphs \mathbf{x} and \mathbf{y} differ in only one covariate, say T_k , so that

$$\frac{P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta})}{P(\mathbf{X} = \mathbf{y}; \boldsymbol{\beta})} = \gamma_k^{\delta_k(\mathbf{x}, \mathbf{y})} = (\exp(\beta_k))^{T_k(\mathbf{x}) - T_k(\mathbf{y})}.$$

Above, we recognize a function of the form $y = a^x$, whose behaviour can be simply analysed for two separate cases $a < 1$ and $a > 1$. Below is a summary of such an analysis.

$$\begin{array}{llll} \beta_k > 0, & T_k(\mathbf{x}) > T_k(\mathbf{y}) & \implies & P(\mathbf{x}; \boldsymbol{\beta}) > P(\mathbf{y}; \boldsymbol{\beta}) \\ \beta_k > 0, & T_k(\mathbf{x}) < T_k(\mathbf{y}) & \implies & P(\mathbf{x}; \boldsymbol{\beta}) < P(\mathbf{y}; \boldsymbol{\beta}) \\ \beta_k < 0, & T_k(\mathbf{x}) > T_k(\mathbf{y}) & \implies & P(\mathbf{x}; \boldsymbol{\beta}) < P(\mathbf{y}; \boldsymbol{\beta}) \\ \beta_k < 0, & T_k(\mathbf{x}) < T_k(\mathbf{y}) & \implies & P(\mathbf{x}; \boldsymbol{\beta}) > P(\mathbf{y}; \boldsymbol{\beta}) \\ \beta_k = 0 \text{ or } T_k(\mathbf{x}) = T_k(\mathbf{y}) & & \implies & P(\mathbf{x}; \boldsymbol{\beta}) = P(\mathbf{y}; \boldsymbol{\beta}) \end{array}$$

As an example, the first line of this summary can be interpreted as follows: for positive β_k , larger values of $T_k(\mathbf{x})$ result in higher probabilities of \mathbf{x} .

2.2.3 Local Interpretation

By the *local* interpretation we mean the description of how the covariates $T_k(\cdot)$ affect the conditional probability of any individual edge indicator X_{ij} in the random graph \mathbf{X} , *given* a set of fixed realizations $X_{kl} = x_{kl}$ of all *other* indicators. To make local interpretation simple, we need to introduce the following set of notation.

- Given a graph \mathbf{x} , denote by \mathbf{x}_{ij}^c the set $\{x_{kl} : (k, l) \neq (i, j)\}$, which is called the *complement set* of the edge indicator x_{ij} . Thus, the complement set of x_{ij} is comprised of all edge indicators other than x_{ij} itself.
- Given a graph \mathbf{x} , denote by \mathbf{x}_{ij}^+ the graph \mathbf{y} in which $y_{ij} = 1$ and $\mathbf{y}_{ij}^c = \mathbf{x}_{ij}^c$. In words, the graph \mathbf{x}_{ij}^+ contains all the edges of \mathbf{x} with the additional condition that the edge e_{ij} is included (regardless of whether it is included in the original graph \mathbf{x}).
- By analogy, denote by \mathbf{x}_{ij}^- the graph \mathbf{y} in which $y_{ij} = 0$ and $\mathbf{y}_{ij}^c = \mathbf{x}_{ij}^c$. In words, the graph \mathbf{x}_{ij}^- contains all the edges of \mathbf{x} except the edge e_{ij} (regardless of whether this edge exists in the original graph \mathbf{x}).

It should be clear that \mathbf{x} is actually equal to either \mathbf{x}_{ij}^+ (if $x_{ij} = 1$) or \mathbf{x}_{ij}^- (if $x_{ij} = 0$).

Choose an indicator variable X_{ij} in a random graph \mathbf{X} and compute its conditional odds given that $\mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c$:

$$\pi_{ij}(\mathbf{x}_{ij}^c; \beta) := \frac{P(X_{ij} = 1 \mid \mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \beta)}{P(X_{ij} = 0 \mid \mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \beta)} \quad (9)$$

$$\begin{aligned} &= \frac{P(X_{ij} = 1, \mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \beta)}{P(X_{ij} = 0, \mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \beta)} \frac{P(\mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \beta)}{P(\mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \beta)} \\ &= \frac{P(\mathbf{X} = \mathbf{x}_{ij}^+; \beta)}{P(\mathbf{X} = \mathbf{x}_{ij}^-; \beta)} = e^{\delta_0(\mathbf{x}_{ij}^+, \mathbf{x}_{ij}^-)} \gamma_1^{\delta_1(\mathbf{x}_{ij}^+, \mathbf{x}_{ij}^-)} \gamma_2^{\delta_2(\mathbf{x}_{ij}^+, \mathbf{x}_{ij}^-)} \dots \gamma_r^{\delta_r(\mathbf{x}_{ij}^+, \mathbf{x}_{ij}^-)}. \end{aligned} \quad (10)$$

Next, we introduce additional notation for the sake of compactness:

$$\delta_{kij}(\mathbf{x}_{ij}^c) := \delta_k(\mathbf{x}_{ij}^+, \mathbf{x}_{ij}^-) = T_k(\mathbf{x}_{ij}^+) - T_k(\mathbf{x}_{ij}^-).$$

The functions δ_{kij} are called the *change statistics*. In words, a change statistic $\delta_{kij}(\mathbf{x}_{ij}^c)$ represents the difference (in the covariate T_k) between having or not having the edge e_{ij} in graph \mathbf{x} , assuming that the rest of its edges remain fixed.

Using the change statistics, we can restate the previous result (9) as

$$\pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) = e^{\delta_{0ij}(\mathbf{x}_{ij}^c)} \gamma_1^{\delta_{1ij}(\mathbf{x}_{ij}^c)} \gamma_2^{\delta_{2ij}(\mathbf{x}_{ij}^c)} \dots \gamma_r^{\delta_{rij}(\mathbf{x}_{ij}^c)}. \quad (11)$$

Fixing the complement set \mathbf{x}_{ij}^c of the edge indicator x_{ij} , suppose that the presence or absence of the edge e_{ij} in graph \mathbf{x} only affects the value of one covariate $T_k(\mathbf{x})$. By analogy with the analysis used for global interpretation, our assumption leads to the following relationships.

$$\begin{aligned} \beta > 0, \quad T_k(\mathbf{x}_{ij}^+) > T_k(\mathbf{x}_{ij}^-) &\implies \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) > 1 \\ \beta > 0, \quad T_k(\mathbf{x}_{ij}^+) < T_k(\mathbf{x}_{ij}^-) &\implies \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) < 1 \\ \beta < 0, \quad T_k(\mathbf{x}_{ij}^+) > T_k(\mathbf{x}_{ij}^-) &\implies \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) < 1 \\ \beta < 0, \quad T_k(\mathbf{x}_{ij}^+) < T_k(\mathbf{x}_{ij}^-) &\implies \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) > 1 \\ \beta = 0 \text{ or } T_k(\mathbf{x}_{ij}^+) = T_k(\mathbf{x}_{ij}^-) &\implies \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) = 1 \end{aligned}$$

The first two of these relationships can be interpreted as follows: for positive β_k , *given* the complement \mathbf{x}_{ij}^c of x_{ij} , the edge e_{ij} will be more often included in the random graph than not included in the random graph *only if* the inclusion of e_{ij} results in a positive gain in the covariate T_k .

Note that we can include the change statistics δ_{kij} ($k = 1, 2, \dots, r$) in the r -dimensional vector

$$\Delta_{ij}(\mathbf{x}_{ij}^c) = (\delta_{1ij}(\mathbf{x}_{ij}^c), \dots, \delta_{rij}(\mathbf{x}_{ij}^c))^T,$$

so that equation (11) can be expressed in terms of conditional log-odds as

$$\log \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) = \delta_{0ij}(\mathbf{x}_{ij}^c) + \boldsymbol{\beta}^T \Delta_{ij}(\mathbf{x}_{ij}^c).$$

2.2.4 The Case of Complete Edge Independence

Suppose that the conditional odds $\pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta})$ do not actually depend on the condition $\mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c$, that is:

$$\pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) = \pi_{ij}(\boldsymbol{\beta}) = \text{constant for each } \mathbf{x} \in \Omega \quad (12)$$

or

$$\delta_{0ij}(\mathbf{x}_{ij}^c) + \beta_1 \delta_{1ij}(\mathbf{x}_{ij}^c) + \dots + \beta_r \delta_{rij}(\mathbf{x}_{ij}^c) = \log \pi_{ij}(\boldsymbol{\beta}) = \text{constant for all } \mathbf{x} \in \Omega,$$

or

$$T_0(\mathbf{x}_{ij}^+) - T_0(\mathbf{x}_{ij}^-) + \boldsymbol{\beta}^T(\mathbf{T}(\mathbf{x}_{ij}^+) - \mathbf{T}(\mathbf{x}_{ij}^-)) = \log \pi_{ij}(\boldsymbol{\beta}) = \text{constant for all } \mathbf{x} \in \Omega. \quad (13)$$

For example, this condition is achieved when none of the change statistics $\delta_{kij}(\mathbf{x}_{ij}^c)$ actually involve \mathbf{x}_{ij}^c , i.e., $\delta_{kij}(\mathbf{x}_{ij}^c) = \delta_{kij} = \text{constant}$ for each \mathbf{x} in Ω .

This situation allows us to compute the inclusion rates as follows.

$$\begin{aligned} p_{ij}(\boldsymbol{\beta}) &= P(X_{ij} = 1) = \sum_{\mathbf{x}_{ij}^c} P(X_{ij} = 1 \mid \mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \boldsymbol{\beta}) P(\mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \boldsymbol{\beta}) \\ &= \frac{\pi_{ij}(\boldsymbol{\beta})}{1 + \pi_{ij}(\boldsymbol{\beta})} \sum_{\mathbf{x}_{ij}^c} P(\mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \boldsymbol{\beta}) = \frac{\pi_{ij}(\boldsymbol{\beta})}{1 + \pi_{ij}(\boldsymbol{\beta})}. \end{aligned} \quad (14)$$

Moreover, we can show complete edge independence in the random graph \mathbf{X} . To see this, consider first the abstract case in which X, Y, Z are any discrete random quantities, and use the elementary probability laws to obtain the relationship

$$\begin{aligned} P(X = x, Y = y) &= P(X = x \mid Y = y)P(Y = y) \\ &= \sum_z P(X = x \mid Z = z, Y = y)P(Z = z \mid Y = y)P(Y = y). \end{aligned}$$

Now substitute in the above formula $X = X_{ij}$, $Y = X_{kl}$ (where $e_{ij} \neq e_{kl}$), and

$$\mathbf{Z} = \{X_{pq} : e_{pq} \neq e_{ij}, e_{pq} \neq e_{kl}\},$$

i.e., \mathbf{Z} can be viewed as the random vector containing all edge indicators *except* X_{ij} and X_{kl} . We get

$$\begin{aligned} P(X_{ij} = x_{ij}, X_{kl} = x_{kl}) &= \sum_{\mathbf{z}} P(X_{ij} = x_{ij} \mid \cdots) P(\mathbf{Z} = \mathbf{z} \mid X_{kl} = x_{kl}) P(X_{kl} = x_{kl}) \\ &= P(X_{ij} = x_{ij}) P(X_{kl} = x_{kl}) \sum_{\mathbf{z}} P(\mathbf{Z} = \mathbf{z} \mid X_{kl} = x_{kl}) \\ &= P(X_{ij} = x_{ij}) P(X_{kl} = x_{kl}). \end{aligned}$$

Using the same technique we can show that for any collection $\{e_{i_1 j_1}, \dots, e_{i_m j_m}\}$ of distinct edges we have

$$P(X_{i_1 j_1} = x_{i_1 j_1}, \dots, X_{i_m j_m} = x_{i_m j_m}) = \prod_{k=1}^m P(X_{i_k j_k} = x_{i_k j_k}).$$

Note that the converse also holds—that is, if we begin with complete edge independence, then (12) must hold.

Looking at (13), we realize that we can choose *any* graph \mathbf{x} and calculate all the odds π_{ij} based on this particular choice. For example, we could use the empty graph $\mathbf{0}$ (zero incidence matrix), in which case

$$\delta_{kij} = T_k(\mathbf{0}_{ij}^+) - T_k(\mathbf{0}), \quad k = 0, 1, 2, \dots, r,$$

which represents the difference (in the covariate T_k) between having only one edge e_{ij} in the graph and not having any edges at all.

We conclude that the edge indicators X_{ij} follow the logistic regression model

$$\log \pi_{ij}(\boldsymbol{\beta}) = \log \left(\frac{p_{ij}(\boldsymbol{\beta})}{1 - p_{ij}(\boldsymbol{\beta})} \right) = \delta_{0ij} + \beta_1 \delta_{1ij} + \beta_2 \delta_{2ij} + \dots + \beta_r \delta_{rij}, \quad (15)$$

so the whole random graph \mathbf{X} can be viewed as the set of M independent binary variables following a familiar logistic regression model. (Here M is the number of possible edges.) This special case will be the basis for the Balanced Potential Models in Chapter 4. Using the result in (14), we can express the joint probability (the probability of the whole graph) as the product²

$$\begin{aligned} P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta}) &= \prod_i \prod_j \left(\frac{\pi_{ij}(\boldsymbol{\beta})}{1 + \pi_{ij}(\boldsymbol{\beta})} \right)^{x_{ij}} \left(\frac{1}{1 + \pi_{ij}(\boldsymbol{\beta})} \right)^{1-x_{ij}} \\ &= \prod_i \prod_j \frac{[\pi_{ij}(\boldsymbol{\beta})]^{x_{ij}}}{1 + \pi_{ij}(\boldsymbol{\beta})}. \end{aligned} \quad (16)$$

We note for future reference that the normalizing constant $K(\boldsymbol{\beta})$, as seen in the above formula, has the form

$$K(\boldsymbol{\beta}) = \prod_i \prod_j (1 + \pi_{ij}(\boldsymbol{\beta}))^{-1},$$

where $\pi_{ij}(\boldsymbol{\beta})$ comes from equation (15).

2.3 Some Examples

In this section we briefly review several well-known models for random graphs, all of which are examples of the log-linear model. The sole purpose of this section is to illustrate the

²In this product the ranges of vertex indices i and j depend on the underlying sample space Ω of the random graph \mathbf{X} . If \mathbf{X} is undirected, then $1 \leq i, j \leq N$ ($i \neq j$); if \mathbf{X} is directed, then $1 \leq i < j \leq N$. Throughout this chapter, the same clarification applies to all other sums and products of this form.

various possibilities offered by log-linear random graphs, and not to provide a critique of the models or compare them with each other. For in-depth analysis, the reader is advised to consult with the original papers that we refer to in each of these examples.

2.3.1 Bernoulli graphs

Bernoulli random graphs assume complete edge independence and assign inclusion probability p_{ij} to each edge e_{ij} . The distribution of the whole graph is then

$$P(\mathbf{X} = \mathbf{x}; p_{ij}) = \prod_i \prod_j p_{ij}^{x_{ij}} (1 - p_{ij})^{1-x_{ij}},$$

or

$$P(\mathbf{X} = \mathbf{x}; \beta_{ij}) = \exp \left(\sum_i \sum_j \beta_{ij} x_{ij} \right) \prod_i \prod_j (1 + e^{\beta_{ij}})^{-1},$$

which indicates that \mathbf{X} is a log-linear random graph, with $\beta_{ij} := \log \left(\frac{p_{ij}}{1-p_{ij}} \right)$.

If we set $p_{ij} = p$ for all i and j , then the resulting graph is often called the $G_{N,p}$ -graph, where N is the number of vertices. Its distribution then reduces to

$$P(\mathbf{X} = \mathbf{x}; \beta) = \exp(\beta |E(\mathbf{x})|) (1 + e^\beta)^{-M},$$

where $\beta := \log(\frac{p}{1-p})$ and M is the number of possible edges (depends on the underlying sample space Ω). $G_{N,p}$ graphs are considered too restrictive to model real networks. However, $G_{N,p}$ graphs and similar models studied by Bollobas (1985) are useful in combinatorics, where their asymptotic behaviour (as $N \rightarrow \infty$) helps prove the existence of certain graph-theoretic properties.

Note that letting $p = 1/2$ yields $\beta = 0$, hence

$$P(\mathbf{X} = \mathbf{x}) = (1/2)^M,$$

which simply means that *all* graphs in Ω are equally likely.

2.3.2 The p_1 -model

The p_1 -model was proposed by Holland & Leinhardt (1981) for directed random graphs with *dyad independence*. This means that if we define the *dyads* as the random vectors

$$\mathbf{D}_{ij} := (X_{ij}, X_{ji}), \quad 1 \leq i < j \leq N,$$

then the \mathbf{D}_{ij} 's are assumed to be statistically independent. Now, each dyad \mathbf{D}_{ij} can take one of the four values:

$$(0, 0), \quad (1, 0), \quad (0, 1), \quad (1, 1),$$

where $\mathbf{D}_{ij} = (0, 0)$ represents a *null* pair, $\mathbf{D}_{ij} = (1, 0)$ or $\mathbf{D}_{ij} = (0, 1)$ represent an *asymmetric* pair, and lastly $\mathbf{D}_{ij} = (1, 1)$ represents a *mutual* pair. To completely describe the distribution of the whole random graph \mathbf{X} , it suffices to specify the distribution for each dyad \mathbf{D}_{ij} ($i < j$), which is done by allocating the total probability among the four possible values:

$$\begin{aligned} m_{ij} &:= P(\mathbf{D}_{ij} = (1, 1)), & n_{ij} &:= P(\mathbf{D}_{ij} = (0, 0)), \\ a_{ij} &:= P(\mathbf{D}_{ij} = (1, 0)), & a_{ji} &:= P(\mathbf{D}_{ij} = (0, 1)), \\ m_{ij} + a_{ij} + a_{ji} + n_{ij} &= 1. \end{aligned}$$

Then

$$P(\mathbf{X} = \mathbf{x}; m_{ij}, n_{ij}, a_{ij}, a_{ji}) = \prod_{i < j} m_{ij}^{x_{ij}x_{ji}} n_{ij}^{(1-x_{ij})(1-x_{ji})} a_{ij}^{x_{ij}(1-x_{ji})} a_{ji}^{(1-x_{ij})x_{ji}}.$$

This distribution can also be expressed as

$$P(\mathbf{X} = \mathbf{x}; \rho_{ij}, \theta_{ij}) = \exp \left(\sum_{i < j} \rho_{ij} x_{ij} x_{ji} + \sum_{i \neq j} \theta_{ij} x_{ij} \right) \prod_{i < j} n_{ij},$$

where ρ_{ij} ($i < j$) and θ_{ij} ($i \neq j$) are chosen appropriately as functions of a_{ij}, m_{ij}, n_{ij} (see Holland & Leinhardt (1981)). Thus the p_1 -model is an example of a log-linear random graph, which actually gives rise to several convenient interpretations of the parameters ρ_{ij} ($i < j$) and θ_{ij} ($i \neq j$). A special case is when $m_{ij} = m, n_{ij} = n, a_{ij} = a_{ji} = a$ for all $i < j$, and Holland & Leinhardt give an interesting interpretation of this assumption. The p_1 -model is often called the *reciprocity* model because it allows to control for the tendencies of dyads to become reciprocal (mutual, symmetric).

2.3.3 Markov Graphs

Markov Graphs were proposed by Frank & Strauss (1986) as a step away from the presumably unrealistic assumption of dyad independence (p_1 -models). Instead, Frank & Strauss assume Markov dependency between dyads, according to which any dyad \mathbf{D}_{ij} , when conditioned on the values of all other dyads, is independent of those dyads \mathbf{D}_{kl} which do not

share a vertex with \mathbf{D}_{ij} . It can be shown that such an assumption implies a very specific functional form of the random graph's distribution. For an undirected random graph \mathbf{X} on N vertices, this distribution is

$$P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta}, \tau) = \exp \left(\beta_1 |E(\mathbf{x})| + \sum_{k=2}^{N-1} \beta_k S_k(\mathbf{x}) + \tau T(\mathbf{x}) \right) K(\boldsymbol{\beta}, \tau), \quad (17)$$

where $S_k(\mathbf{x})$ ($2 \leq k \leq N-1$) counts the number of k -stars in the graph, and $T(\mathbf{x})$ counts the number of triangles (3-cycles):

$$S_k(\mathbf{x}) = \sum_{i=1}^N \binom{\deg_{\mathbf{x}}(i)}{k}, \quad T(\mathbf{x}) = \sum_{1 \leq i < j < h \leq N} x_{ij} x_{jh} x_{hi}.$$

In fact, the form of the distribution in (17) is both sufficient and necessary for the random graph \mathbf{X} to have the Markov dependency structure. Note that in undirected graphs the dyads \mathbf{D}_{ij} are actually identified with the edge indicators X_{ij} . The authors discuss mainly the special case in which $\beta_k = 0$ for $k \geq 3$, i.e., there are only three parameters in the model. For directed random graphs, the corresponding necessary distribution is somewhat involved. See Frank & Strauss (1986) for the details.

Based on the relationship between $S_k(\mathbf{x})$ ($k \geq 2$) and the observed vertex degrees $\deg_{\mathbf{x}}(i)$ ($i = 1, 2, \dots, N$), Snijders et al. (2006) demonstrate that it is possible to use the degree counts

$$d_k(\mathbf{x}) := \text{number of vertices with degree } k \text{ in } \mathbf{x}$$

instead of the S_k 's in the exponent of (17) and still remain in the class of Markov graphs. Snijders et al. then propose to define $\beta_k = e^{-\alpha k}$ for some $\alpha > 0$, leading to the *geometrically decreasing degree distribution* assumption. However, this approach implies that the β_k 's cannot vary independently of each other. Thus, according to our explanations in Subsection 2.2.1, this approach should be classified among the family of curved exponential random graphs rather than the log-linear ones.

2.4 Estimation

We now turn to the problem of estimating the unknown parameters β_1, \dots, β_r in the exponential random graph \mathbf{X} whose probability mass function is

$$P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta}) = \exp \left(\sum_{k=1}^r \beta_k T_k(\mathbf{x}) + T_0(\mathbf{x}) \right) K(\boldsymbol{\beta}), \quad \mathbf{x} \in \Omega.$$

The purpose of this section is twofold: to introduce the reader to the popular estimation methods and to provide a line of arguments in favour of our proposition that complex edge dependencies (brought in by structural covariates) require difficult and often unreliable estimation methods. The reader may recall us mentioning this problem in Chapter 1, where we said that the more realistic modelling of edge dependence comes with the cost of troublesome model fitting.

Having observed the graph realization \mathbf{x} , we can view it as the joint outcome of all the edge indicators X_{ij} , i.e.:

$$P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta}) = P(X_{ij} = x_{ij} \text{ for all } i, j; \boldsymbol{\beta}). \quad (18)$$

In theory, then, we could take the Maximum Likelihood approach and find the value $\hat{\boldsymbol{\beta}}$ of the parameter vector $\boldsymbol{\beta}$ that maximizes the above probability for a fixed set of observations $X_{ij} = x_{ij}$. In practice, this approach is unavailable due to the high computational cost required to work out the normalizing function

$$K(\boldsymbol{\beta}) = \left(\sum_{\mathbf{x} \in \Omega} \exp(\boldsymbol{\beta}^T \mathbf{T}(\mathbf{x}) + T_0(\mathbf{x})) \right)^{-1}. \quad (19)$$

For example, if Ω is the set of all undirected graphs on 20 vertices, then the above summation involves as many as 2^{190} terms.

Since the Maximum Likelihood function is intractable both computationally and analytically, the authors in the field (e.g., Wasserman & Robins (2005)) suggest two possible solutions. First, one could use a Markov Chain Monte Carlo algorithm to approximate the ML estimates (this approach is abbreviated as MCMC MLE). Second, one could use the Pseudolikelihood estimation instead of MLE. But before we proceed with these methods, let us comment on the relevance of asymptotic properties of ML estimators in the context of random graphs.

2.4.1 A Note on Asymptotics

In the trivial case of complete edge independence, the joint likelihood function (18) is expressed as the product (16) of M individual likelihoods (for M independent observations $X_{ij} = x_{ij}$), where M is the number of possible edges in the graph on N vertices. Since M grows whenever N grows, the standard asymptotic results for the ML estimator are quite relevant. For example, we can assert that the ML estimator $\hat{\boldsymbol{\beta}}$ is approximately

normal when the number of vertices is large. However, if we assume dependence between edges, then we are confined to a single observation (of the whole graph) that can not be interpreted as M independent observations. Therefore, unless the joint likelihood can be factored into several likelihoods whose number grows as the number of vertices grows, we can not make practical use of the standard Cramér assumptions (see Cramér (1946)) to prove the asymptotic properties of the ML estimator. For example, to assert that the ML estimator $\hat{\beta}$ is approximately normal, we would have to observe independently a large number of graphs instead of just one. In practice, however, we rarely observe more than one graph realization from a particular model. van Dujin et al. (2009) note that, in the absence of edge independence, the ML-based methods are not privileged to other methods based on the asymptotic arguments alone. However, this statement is too strong, since for many statistical models it is still possible, in theory, to prove the asymptotic normality and efficiency of the ML estimator based on more general assumptions than those of Cramer. For example, we could aim at proving local asymptotic normality of Le Cam (1960), for which statistical independence of observations (edges) is not required. (Although it seems that such a task may turn out very difficult for many ERG models.)

2.4.2 Introduction to MCMC MLE

There are many different ways to solve an intractable estimation problem by means of Monte Carlo simulation (see, e.g., Gilks et al. (1996)). The MCMC MLE algorithm is an iterative algorithm whose t 'th iteration consists of two steps:

1. *Simulation Step.* Simulate a large number, say n , of random draws from the probability distribution with the parameter set to current estimate $\hat{\beta}^{(t)}$.

This step allows us to approximate the intractable characteristics of the distribution. For example, suppose that $g(\mathbf{X})$ is some graph statistic, and we wish to approximate its expectation

$$\mu(g(\mathbf{X})) := \mathbb{E}_{\hat{\beta}}(g(\mathbf{X})) = \sum_{\mathbf{x} \in \Omega} g(\mathbf{x}) P(\mathbf{X} = \mathbf{x}; \hat{\beta}),$$

which can not be computed by ordinary means. The corresponding Monte Carlo approximation is given by

$$\hat{\mu}_{MC}(g(\mathbf{X})) = \sum_{k=1}^n g(\mathbf{x}^{(k)})/n,$$

where $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)}$ are the simulated draws.

2. *Update Step.* Use the simulated draws $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)}$ together with the actually observed outcome \mathbf{x} to update the parameter estimate from $\hat{\boldsymbol{\beta}}^{(t)}$ to $\hat{\boldsymbol{\beta}}^{(t+1)}$, where $\hat{\boldsymbol{\beta}}^{(t+1)}$ is computed in accordance with any one of the several applicable techniques that we discuss below.

These two steps are repeated until the parameter estimates stabilize. The initial estimate $\hat{\boldsymbol{\beta}}^{(1)}$ can be chosen arbitrarily, although this choice may largely affect the convergence of the algorithm.

Several methods are available for each of the two steps, thus there exist many variations of the MCMC MLE procedure. We will now briefly introduce two popular methods to be performed at the update step, and then we will discuss the simulation step. The simulation step turns out to be the most problematic step in the context of random graphs.

2.4.3 Geyer-Thompson Update Step

This method originates from Geyer & Thompson (1992). Let \mathbf{x} be the observed graph and let $\hat{\boldsymbol{\beta}}^{(t)}$ be the current estimate of $\boldsymbol{\beta}$. Note that maximizing the likelihood function is equivalent to maximizing the relative log-likelihood function

$$r(\boldsymbol{\beta}, \hat{\boldsymbol{\beta}}^{(t)}) = \left(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{(t)} \right)^T \mathbf{T}(\mathbf{x}) + \log \left(\frac{K(\boldsymbol{\beta})}{K(\hat{\boldsymbol{\beta}}^{(t)})} \right),$$

where $K(\cdot)$ is the intractable normalizing function in (19). Observe that

$$\begin{aligned} \frac{K(\hat{\boldsymbol{\beta}}^{(t)})}{K(\boldsymbol{\beta})} &= \sum_{\mathbf{y} \in \Omega} \exp \left(\boldsymbol{\beta}^T \mathbf{T}(\mathbf{y}) + T_0(\mathbf{y}) \right) K(\hat{\boldsymbol{\beta}}^{(t)}) \\ &= \sum_{\mathbf{y} \in \Omega} \exp \left(\boldsymbol{\beta}^T \mathbf{T}(\mathbf{y}) - \hat{\boldsymbol{\beta}}^{(t)T} \mathbf{T}(\mathbf{y}) \right) \exp \left(\hat{\boldsymbol{\beta}}^{(t)T} \mathbf{T}(\mathbf{y}) + T_0(\mathbf{y}) \right) K(\hat{\boldsymbol{\beta}}^{(t)}) \\ &= \mathbb{E}_{\hat{\boldsymbol{\beta}}^{(t)}} \exp \left((\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{(t)})^T \mathbf{T}(\mathbf{X}) \right) \\ &\approx \frac{1}{n} \sum_{k=1}^n \exp \left((\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{(t)})^T \mathbf{T}(\mathbf{x}^{(k)}) \right), \end{aligned}$$

where $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)}$ are the draws obtained in the simulation step. We thus get a Monte Carlo approximation of the relative log-likelihood function:

$$\hat{r}_{MC}(\boldsymbol{\beta}, \hat{\boldsymbol{\beta}}^{(t)}) = \left(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{(t)} \right)^T \mathbf{T}(\mathbf{x}) - \log \left(\frac{1}{n} \sum_{k=1}^n \exp \left((\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^{(t)})^T \mathbf{T}(\mathbf{x}^{(k)}) \right) \right).$$

We then set $\hat{\boldsymbol{\beta}}^{(t+1)}$ as the maximizer of the above function of $\boldsymbol{\beta}$, i.e., we solve an additional optimization problem. It is noted by Hunter & Handcock (2006) that $\hat{r}_{MC}(\boldsymbol{\beta}, \hat{\boldsymbol{\beta}}^{(t)})$ strongly converges to $r(\boldsymbol{\beta}, \hat{\boldsymbol{\beta}}^{(t)})$ as the number n of Monte Carlo draws tends to infinity.

2.4.4 Robbins-Monro Update Step

This update step was suggested by Snijders (2002) to be used with exponential random graphs. In fact, this step is known as the *adaptive Robbins-Monro* procedure, is due to Robbins & Monro (1951), Venter (1967), and Nevel'son & Hasminski (1973), and can be viewed as an elaboration on the Newton-Raphson algorithm.

For convenience, assume that the random graph \mathbf{X} has the probability distribution

$$P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta}) = \exp(\boldsymbol{\beta}^T \mathbf{T}(\mathbf{x})) K(\boldsymbol{\beta}),$$

i.e., the non-parametrized covariate $T_0(\mathbf{x})$ is not used in the distribution. A well known fact (see, e.g., Lehmann (1983)) is that the ML estimate $\hat{\boldsymbol{\beta}}$ is also the solution to the equation

$$\mathbb{E}_{\boldsymbol{\beta}}(\mathbf{T}(\mathbf{X})) = \mathbf{T}(\mathbf{x}),$$

where \mathbf{x} is the observed graph. Thus we can say that the goal of the MCMC procedure is to converge to a solution to this equation. Define $\boldsymbol{\mu}(\boldsymbol{\beta}) := \mathbb{E}_{\boldsymbol{\beta}}(\mathbf{T}(\mathbf{X}))$, then the above equation becomes

$$\boldsymbol{\mu}(\boldsymbol{\beta}) = \mathbf{T}(\mathbf{x}).$$

If we were able to compute $\boldsymbol{\mu}(\boldsymbol{\beta})$, then the appropriate Robbins-Monro update step would be

$$\hat{\boldsymbol{\beta}}^{(t+1)} = \hat{\boldsymbol{\beta}}^{(t)} - (1/t) \mathbf{D}^{(t)}(\mathbf{T}(\mathbf{y}^{(t)}) - \mathbf{T}(\mathbf{x})),$$

where $\mathbf{y}^{(t)}$ is a random draw of a random graph with the distribution $P(\mathbf{X} = \mathbf{x}; \hat{\boldsymbol{\beta}}^{(t)})$ and the matrix $\mathbf{D}^{(t)}$ is the inverse Hessian (derivative) matrix of $\boldsymbol{\mu}(\boldsymbol{\beta})$ evaluated at $\hat{\boldsymbol{\beta}}^{(t)}$. But the function $\boldsymbol{\mu}(\boldsymbol{\beta})$ is intractable, and we have to employ one of the *adaptive* techniques to estimate the matrix $\mathbf{D}^{(t)}$ during the approximation process.

Note that this approach implies that only one random draw $\mathbf{y}^{(t)}$ was obtained in the simulation step (i.e., $n = 1$). For the adaptive techniques to estimate $\mathbf{D}^{(t)}$, as well as other numerous details and references, the reader is advised to look in Snijders (2002). Snijders notes that the difference between the Geyer-Thompson and the Robbins-Monro approaches is a matter of convenience. We notice, however, that the former has the disadvantages of having to solve an optimization sub-problem and having to generate a large number of draws in the simulation step. Snijders (1996) suggests that between 100 and 500 iterations of the Robbins-Monro procedure are required, provided that the algorithm is able to converge (a matter that we discuss next).

2.4.5 Simulation Step and Degeneracy

Due to the intractability of the normalizing constant, direct sampling is not available for the simulation step. To simulate random draws from the random graph distribution $P(\mathbf{X} = \mathbf{x}; \hat{\boldsymbol{\beta}}^{(t)})$, one could use either the Gibbs sampling technique or the more general Metropolis-Hastings algorithm. To illustrate the main idea, we describe the Gibbs sampling technique.

The goal is to generate a Markov chain $\{\mathbf{X}^{(k)} : k = 0, 1, 2, \dots\}$ whose equilibrium distribution is equal to the “target” distribution $P(\mathbf{X} = \mathbf{x}; \hat{\boldsymbol{\beta}}^{(t)})$ of the exponential random graph. Since the approach is based on the limiting behaviour of the transitional probabilities $P(\mathbf{X}^{(k)} = \mathbf{x}^{(k)} \mid \mathbf{X}^{(k-1)} = \mathbf{x}^{(k-1)}; \hat{\boldsymbol{\beta}}^{(t)})$ ($k \geq 1$), it is wise and is the common practice in MCMC to disregard the beginning segment of some length (say, $k = 1000$) in this chain. Starting with an arbitrary graph $\mathbf{x}^{(0)}$, the rest of the chain is generated as follows.

At step $k \geq 1$, we have at our disposal the graph $\mathbf{x}^{(k-1)}$ obtained in the previous step. Beginning with some edge indicator, say $X_{12}^{(k-1)}$, all the $X_{ij}^{(k-1)}$ ’s are updated in turn to a new randomly chosen value $X_{ij}^{(k)}$ until the whole collection of edge indicators is traversed. The updating mechanism specifies that, if $X_{ij}^{(k-1)}$ is the edge indicator currently updated, then the new value $X_{ij}^{(k)}$ is generated according to the conditional distribution of $X_{ij}^{(k)}$ given all the values $X_{ml}^{(k)} = x_{ml}^{(k)}$ that have been updated up to this point as well as all the values $X_{pq}^{(k-1)} = x_{pq}^{(k-1)}$ ($e_{pq} \neq e_{ij}$) that still remain unchanged from the previous step. Letting $\mathbf{y}_{ij}^{(k)}$ denote the collection $\{x_{ml}^{(k)}, x_{pq}^{(k-1)}\}$ that we just described, we can now formally require

the conditional distribution of $X_{ij}^{(k)}$ given $\mathbf{Y}_{ij}^{(k)} = \mathbf{y}_{ij}^{(k)}$ to match its “target” version, i.e:

$$\begin{aligned} P(X_{ij}^{(k)} = x_{ij} \mid X_{ml}^{(k)} = x_{ml}^{(k)}, X_{pq}^{(k-1)} = x_{pq}^{(k-1)}; \hat{\boldsymbol{\beta}}^{(t)}) &= P(X_{ij} = x_{ij} \mid \mathbf{X}_{ij}^c = \mathbf{y}_{ij}^{(k)}; \hat{\boldsymbol{\beta}}^{(t)}) \\ &= \frac{\left[\pi_{ij}(\mathbf{y}_{ij}^{(k)}; \hat{\boldsymbol{\beta}}^{(t)}) \right]^{x_{ij}}}{1 + \pi_{ij}(\mathbf{y}_{ij}^{(k)}; \hat{\boldsymbol{\beta}}^{(t)})}. \end{aligned}$$

The right-hand side of the above equation is, of course, quite familiar to us from our acquaintance with log-linear random graphs (subsection 2.2.3 in particular). After we have updated all the edge indicators, we are left with the graph $\mathbf{x}^{(k)}$ that represents the k ’th realization of the Markov chain $\{ \mathbf{X}^{(k)} : k = 0, 1, 2, \dots \}$.

We may contend at this point that the MCMC MLE approach is rather computationally intensive. Indeed, the procedure often requires a huge number of draws and iterations. However, this is not the largest disadvantage of the algorithm, nor is it a weakness that could be ascribed specifically to random graphs. (MCMC methods are generally well-known for their high computational demands—see, e.g., Lai (2003)). The biggest drawback, as discussed at length by numerous sources (Wasserman & Robins (2005), Snijders (2002), and others), is the tendency of the simulation part of the algorithm to exhibit *degeneracy* (or, *instability*) when applied to random graphs.

Degeneracy occurs when the model places a disproportionately large probability mass on only a few of the possible graphs. Moreover, these graphs are often uninteresting, e.g., empty graphs or complete graphs. Snijders (2002) generalizes this phenomenon in the discussion of *bimodality* (or *multimodality*) of certain specifications of the log-linear random graph model. This is a situation in which the sample space Ω is divided into two or more regions (or *regimes*) such that the typical MCMC algorithm is prone to spending extremely long time within any one of these regions, as the probability of moving from one region to another is negligible. These problems can invalidate the estimation algorithm. It is important to realize that degeneracy is not the drawback of the estimation algorithm itself, but of the specified model for the exponential random graph as well as the true parameter values. For instance, Snijders et al. (2006) mention that if in a Markov graph model (Subsection 2.3.3) the values β_k are positive for large k , then such a model assigns high probabilities to graphs with large degrees—a circumstance that contributes heavily to the problem of degeneracy. Therefore, unless we perform a check for degeneracy beforehand, we can not be sure that the MCMC MLE procedure will converge or that it will converge to reliable parameter estimates. The study of degeneracy is an active topic

in network research. See Handcock (2003), Handcock et al. (2003), and Schweinberger (2011).

It is not surprising that the method is highly sensitive to the initial parameter estimate $\hat{\boldsymbol{\beta}}^{(1)}$. For example, the algorithm shows poor convergence when the initial estimate is far from the MLE. The study of ways to improve convergence speed as well as ways to choose initial values is another active area in network research. See Bartz et al. (2010) and Bartz (2011).

2.4.6 Introduction to Pseudolikelihood

As an estimation method to be applied to the generic log-linear models with untractable normalizing constants, Maximum Pseudolikelihood Estimation (MPL) originates in Besag (1975). In his paper Besag is concerned with the statistical analysis of spatial data that comes as a rectangular array of binary variables. We can see the connection to random graph modeling, where the random adjacency matrix \mathbf{X} is exactly a rectangular array of binary variables, but with the diagonal elements excluded. As an approach to be used specifically with log-linear random graphs, MPL estimation was first suggested in Frank & Strauss (1986) apropos of the Markov Graph models, followed by Strauss & Ikeda (1990), which is entirely devoted to MPL for log-linear random graphs, and finally popularized by Anderson et al. (1999). From the purely theoretical perspective, Arnold & Strauss (1991) is one of the most notable sources, since it provides the definition and asymptotic properties of MPL estimates in the most general case, not tied to any specific area of application. Part of the upcoming discussion will be based on their particular presentation of the topic.

The MPL estimation method is based on the fact that the conditional odds of X_{ij} given its complement $\mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c$ do not involve the normalizing constant $K(\boldsymbol{\beta})$. Thus we expect the required computation to be significantly simplified. Recall the form of these conditional odds:

$$\pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) = \exp \left(\delta_{0ij}(\mathbf{x}_{ij}^c) + \boldsymbol{\beta}^T \Delta_{ij}(\mathbf{x}_{ij}^c) \right).$$

Also recall the corresponding conditional probabilities:

$$P(X_{ij} = x_{ij} \mid \mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c) = \frac{[\pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta})]^{x_{ij}}}{1 + \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta})}.$$

Given the observed graph $\mathbf{X} = \mathbf{x}$, define the *pseudolikelihood* function as

$$\text{PL}(\boldsymbol{\beta} \mid \mathbf{x}) := \prod_i \prod_j P(X_{ij} = x_{ij} \mid \mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \boldsymbol{\beta}). \quad (20)$$

The pseudolikelihood *estimate* of β is then the value $\hat{\beta}$ that maximizes the pseudolikelihood function in (20). Alternatively, we could aim at the maximization of the log-pseudolikelihood function

$$\begin{aligned}\log \text{PL}(\beta \mid \mathbf{x}) &= \sum_i \sum_j \log P(X_{ij} = x_{ij} \mid \mathbf{X}_{ij}^c = \mathbf{x}_{ij}; \beta) \\ &= \sum_i \sum_j \{x_{ij} \log \pi_{ij}(\mathbf{x}_{ij}^c; \beta) - \log(1 + \pi_{ij}(\mathbf{x}_{ij}^c; \beta))\}.\end{aligned}\quad (21)$$

In the trivial case of complete edge independence, we can refer to the discussion in Subsection 2.2.4 and conclude that maximization of pseudolikelihood is exactly the maximization of regular likelihood for a logistic regression model in which the log-odds of edge indicators X_{ij} are regressed against the fixed change functions δ_{kij} .

If a model involves more complex dependencies between edges, this dependence is ignored or misrepresented by the PL function. Many recent sources such as Wasserman & Robins (2005), Snijders (2002), and van Duijn et al. (2009) openly admit that the properties of the PL estimator are not well understood. On the other hand, in the literature on social network analysis it used to be quite common to regard the PL function as an approximate likelihood function, but little explanation was given of what this “approximation” entailed in more or less exact terms. Some authors (e.g. Frank & Strauss (1986)), note in passing that, in cases where the ML estimates are available, the MPL approach gives estimates similar to those obtained by maximum likelihood. The authors could be referring, for example, to the p_1 reciprocity model (Subsection 2.3.2), for which it *is* possible to give an explicit formula for the ML estimates. Quite possibly, such simple cases once constituted the initial basis for the wide-spread assertion that MPL estimates somehow approximate their ML counterparts. It was suggested by Cessie & van Houwelingen (1994) that, in cases where the correlations are close to zero (in our case, the correlations ρ_{ijkl} between edge indicators X_{ij} and X_{kl}), then the MPL estimates are expected to have smaller losses in efficiency in comparison to the case with highly correlated data. More recently, however, advances in computational power gave rise to a number of additional studies devoted to the comparison of MPL estimates with MCMC ML estimates (e.g., Robins et al. (2007), Lubbers & Snijders (2007), van Duijn et al. (2009)) with the general consensus that the traditional ML estimators clearly show superiority over the MPL estimators. The commonly cited problems were the large bias of the MPL estimators and their occasional tendency to output infinite values for the estimates.

Above, we described those problems with MPL estimation that become apparent from

the practical application of the method and subsequent comparison with MCMC MLE approach. Now we would like to give a critique of the MPL approach from the theoretical point of view. In particular, we address the mathematical and the rational justifications for the MPL approach. By mathematical justification we mean the answer to the question: what are the “good” mathematical properties (e.g., asymptotic properties) of MPL estimators? By rational justification we mean the answer to the question: what is the intuitive motivation behind the maximization of the MPL function?

2.4.7 On Mathematical Justification

It is often quoted that MPL estimators are *consistent* and *asymptotically normal*. However, careful review of the corresponding theorems reveals that, in order to give practical meaning to these results in the context of random graphs, we require to observe several independent observations of the whole graph from the same model. This requirement, as noted in the preceding discussion of the asymptotic properties of ML estimators, is rarely met in practice. To understand what we mean, it is important to reproduce the assumptions of a generalized MPL approach, as provided by Arnold & Strauss (1991).

We suppose that we have n independent observations $(\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(n)})$, where each $\mathbf{Y}^{(j)}$ is a d -dimensional random vector distributed with the joint density $f(\mathbf{y}, \theta)$, same for each $j = 1, \dots, m$. The density $f(\mathbf{y}, \theta)$ can be either continuous (i.e., a density with respect to the Lebesgue measure) or discrete (i.e., a density with respect to the counting measure), the actual case being immaterial to the discussion. Now suppose that, in addition to the $\mathbf{Y}^{(j)}$'s, we are given a set of m transformations of the $\mathbf{Y}^{(j)}$'s:

$$\mathbf{Z}_i^{(j)} = h_i(\mathbf{Y}^{(j)}), \quad i = 1, 2, \dots, m; \quad j = 1, 2, \dots, n.$$

There are no restrictions on the dimensions and the functional forms of the $\mathbf{Z}_i^{(j)}$'s. The generalized MPL estimation strategy involves maximization of the following objective function:

$$\text{PL}(\theta \mid \mathbf{z}_i) = \prod_{j=1}^n \left[\prod_k \prod_{k'} \left(f(\mathbf{z}_k^{(j)} \mid \mathbf{z}_{k'}^{(j)}, \theta) \right)^{a_{kk'}} \right], \quad (22)$$

where it is assumed that k and k' are such that the conditional densities are well defined, and the powers $a_{kk'}$ are positive. Implied also is the free choice of the conditional densities—that is, we are not required to include a specific set of the conditional densities all at once, but only those that we deem relevant (k and k' range accordingly; in fact, the range of k'

may depend on the value of k). Thus, there are many different versions of the function in (22), giving us certain flexibility, at least in theory. The main result here is that, under the standard regularity conditions given in Lehmann (1983), the maximizer $\hat{\theta}$ is a consistent and asymptotically normal estimator of θ .³ Note that both of these asymptotic properties are stated with respect to the number n (of i.i.d. observations) tending to infinity.

Comparing the generalized pseudolikelihood in (22) with the random graph-specific pseudolikelihood in (20), we have $n = 1$ in the latter case. We are thus justified in our claim that the asymptotic properties of either MPL or ML estimators are somewhat irrelevant in the context of log-linear random graphs, even as the number of vertices grows.

Some authors also note that the MPL estimator is not a function of the sufficient statistics $T_k(\cdot)$ and therefore is not an efficient estimator. Lastly, we note that it is also widely agreed that the usual tests of goodness of fit do not strictly apply when there are dependencies within the data. Thus, for example, Wasserman & Robins (2005) assert that the pseudolikelihood *deviance*, computed by the standard logistic regression packages, is not necessarily an asymptotic chi-square random variable. Accordingly, the authors suggest to take such measures of fit as heuristic guides.

2.4.8 On Rational Justification

If we rewrite the log-pseudolikelihood function in (21) as

$$\log \text{PL}(\boldsymbol{\beta} \mid \mathbf{z}) = \sum_i \sum_j \{z_{ij} \log \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) - \log(1 + \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}))\},$$

where

$$\log \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta}) = \delta_{0ij}(\mathbf{x}_{ij}^c) + \boldsymbol{\beta}^T \Delta_{ij}(\mathbf{x}_{ij}^c),$$

then we may recognize a logistic regression model for some set $\mathbf{Z} = \{Z_{ij}\}$ of independent random variables, where the log-odds of each Z_{ij} are given by $\log \pi_{ij}(\mathbf{x}_{ij}^c; \boldsymbol{\beta})$. Therefore the MPL estimation based on the realization $\mathbf{X} = \mathbf{x}$ is equivalent to the ML estimation for the logistic regression based on the realization $\mathbf{Z} = \mathbf{x}$. Thus the MPL estimation can be done in practice using any of the available software tools for the familiar logistic regression analysis. Although it makes it very easy to use the MPL method in practice, this observation does not justify the treatment of \mathbf{x} as a realization of \mathbf{Z} instead of \mathbf{X} .

³For the precise list of regularity conditions stated explicitly in the context of the pseudolikelihood problem, the reader may refer to Appendix A in Geys et al. (1997).

In fact, it is not obvious what the sample space of \mathbf{Z} actually is. We now attempt to construct such a sample space and show that it is not an appropriate representation of the real experiment originally modelled by \mathbf{X} .

Fix an observation \mathbf{x} of the original random graph \mathbf{X} . For each edge e_{ij} define the sample space

$$\Omega_{ij}(\mathbf{x}) := \{ \mathbf{x}_{ij}^+, \mathbf{x}_{ij}^- \}.$$

which essentially codes the two possibilities $X_{ij} = 1$ and $X_{ij} = 0$, given that $\mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c$. Define the random variables Z_{ij} on $\Omega_{ij}(\mathbf{x})$ by

$$Z_{ij} := \begin{cases} 1 & \text{if } \mathbf{x}_{ij}^+ \text{ is observed,} \\ 0 & \text{if } \mathbf{x}_{ij}^- \text{ is observed.} \end{cases}$$

and assume that

$$P(Z_{ij} = 1; \boldsymbol{\beta}) = P(X_{ij} = 1 \mid \mathbf{X}_{ij}^c = \mathbf{x}_{ij}^c; \boldsymbol{\beta}).$$

Let M be the number of possible edges in the original random graph \mathbf{X} and define the new sample space $\Omega^M(\mathbf{x})$ as the Cartesian product of all $\Omega_{ij}(\mathbf{x})$'s. Assume that all the Z_{ij} 's are statistically independent. We have thus constructed the sample space $\Omega^M(\mathbf{x})$ with the property that the maximum likelihood function

$$L(\boldsymbol{\beta} \mid \mathbf{z}_{ij}) = \prod_i \prod_j P(Z_{ij} = z_{ij}; \boldsymbol{\beta})$$

is exactly the same as the pseudolikelihood function $\text{PL}(\boldsymbol{\beta} \mid \mathbf{x})$ in (20) under the assumption that $z_{ij} = x_{ij}$ for all i, j . At this point we encourage the reader to admit that such a sample space is quite unusual, definitely not equivalent to the original sample space Ω , and raising concerns about $\Omega^M(\mathbf{x})$ being a plausible representation of reality.

2.4.9 Summary of Estimation Methods

To summarize, the estimation methods for random graphs with edge dependence often result in unreliable estimates or convergence problems. This is a drawback if we compare these methods with the ML estimation for the familiar logistic regression (for random graphs with edge independence). Nevertheless, these methods seem to be our only choices if we need to use structural covariates $T_k(\cdot)$ to make a log-linear random graph model more realistic. As the computational capabilities grow, statisticians tend to choose the MCMC MLE approach over MPL estimation.

2.5 Dynamic Random Graphs

In Chapter 1 we acknowledged the need for the probabilistic description of dynamic networks—relational datasets that change over time. It seems natural to expand the concept of a random graph to the definition of a *dynamic* random graph, which essentially is a stochastic process comprised of random graphs. However, different researchers may turn out to hold different views about what a dynamic network really is. Indeed, to give a formal definition of a dynamic random graph means to make the following decisions concerning the desired level of abstraction.

1. Do we allow the vertex set V to change over time, or are we only interested in modelling the changes in the edge indicators X_{ij} while keeping V fixed?
2. Do we model the changes in continuous or discrete time?

In theory, of course, it is possible to give a definition that covers all possible answers to the above questions. For example, we may let T be an arbitrary index set (discrete or continuous), and suppose that to each value t in T corresponds a sample space $\Omega(t)$, which is comprised of all graphs (either all directed or all undirected) on some vertex set $V(t)$ (also dependent on t). Thus, a dynamic random graph is defined in the most general sense as the collection

$$\{ \mathbf{X}(t) : t \in T \},$$

where $\mathbf{X}(t)$ is a random graph with the sample space $\Omega(t)$ for each t . In applied research, however, it is often more convenient to adopt a narrower definition—for example, to let $\{ \mathbf{X}(t) : t \in T \}$ be the collection of random graphs on a fixed set V of vertices, with the continuous time index $t \in [0, \infty)$.

Two additional decisions should be made about the probabilistic assumptions of the dynamic random graph.

3. Should we assume that the stochastic process has the Markov property?
4. Should we assume that not more than a single edge X_{ij} can change its state at each moment, or should we allow an arbitrary number of simultaneous changes?

In fact, we can exclude question 3 from the list since the unanimous answer to this question is “yes”. Indeed, Markov Chains are among the most convenient stochastic structures, and all dynamic random graphs are modelled as such.

Now, each combination of answers to questions 1, 2, and 4 generates a “concept image” of a dynamic random graph that needs its own theoretical treatment. Continuous time-models are treated differently than discrete-time models, fixed vertices are treated differently than variable vertices, and so on. In order to get organized, we propose to categorize the different approaches to dynamic random graphs according to Table 2.1, which divides all dynamic random graph models into eight types, depending on whether the time scale is discrete or continuous, whether multiple edge indicators may switch simultaneously, and whether the vertex set remains fixed or not. Note that Table 2.1 shows an additional division of dynamic random graphs into evolutionary and transactional models. We will return to this distinction later on.

	Fixed V	Variable V	
Continuous time, single edge change	<i>Type 1</i>	<i>Type 5</i>	} Evolutionary models
Discrete time, single edge change	<i>Type 2</i>	<i>Type 6</i>	
Continuous time, multiple edge changes	<i>Type 3</i>	<i>Type 7</i>	} Transactional models
Discrete time, multiple edge changes	<i>Type 4</i>	<i>Type 8</i>	

Table 2.1: Classification of Dynamic Random Graphs

The significance of Table 2.1 is that it can be used to identify untapped areas of research. Presently, not much research is devoted to dynamic random graphs, with the prominent exception of a sequence of papers by Snijders and his colleagues devoted to continuous-time models with single edge changes (Snijders (1996), Huisman & Snijders (2003), and Snijders (2005)). By focusing largely on the models of Type 1 and Type 5, these authors tend to neglect other types of dynamic random graphs (for example, discrete-time models with multiple edge changes). However, in the real world we can find examples for which a continuous-time model is not appropriate, but a discrete one is. Similarly, the assumption of single edge change may be more suited to some situations than others. Thus we can find the types of models in Table 2.1 that have been poorly represented in the academic literature, and we can attempt to work with these types of models in parallel with the more popular types of models. This thesis will propose a model of Type 4 in Chapter 3.

We will come back once again to the discussion of the different types of dynamic random graphs after we briefly introduce the models that are the subject of research by Snijders and his colleagues, as well as the preferential attachment model, which is another well-known example of a dynamic random graph.

2.5.1 Holland-Leinhardt-Snijders Approach

Wasserman (1978) gives an exposition of models for (directed) dynamic random graphs that were most relevant at the time of his publication. Among those models is the “dynamic model” of Holland and Leinhardt (1977a, 1977b) which is the basis on which the work of Snijders elaborates. In fact, this “dynamic model” is generic enough to be called a “framework” for modelling continuous-time dynamic random graphs with single edge changes. Naturally, it assumes the Markov property for continuous stochastic processes, with the transition rate satisfying

$$P(\mathbf{X}(t+h) = \mathbf{y} \mid \mathbf{X}(t) = \mathbf{x}) \rightarrow \delta_{\mathbf{x}\mathbf{y}} \text{ as } h \rightarrow 0,$$

where \mathbf{x} and \mathbf{y} are two possible realizations from the sample space Ω , and $\delta_{\mathbf{x}\mathbf{y}}$ is the Kronecker delta function that takes the value of 1 if $\mathbf{x} = \mathbf{y}$ and is 0 otherwise. In words, the current observation \mathbf{x} of the random graph is all that is needed to describe the future behaviour of the process.

The second important assumption is called *conditional choice independence*, which essentially formalizes the idea that the chance of any two edge indicators changing simultaneously must be zero. In formal terms, we have

$$P(\mathbf{X}(t+h) = \mathbf{x} \mid \mathbf{X}(t) = \mathbf{x}(t)) = \prod_{i,j} P(X_{ij}(t+h) = x_{ij} \mid \mathbf{X}(t) = \mathbf{x}(t)) + o(h) \text{ as } h \rightarrow 0.$$

The interpretation is that for very small time intervals the changes in the random graph are statistically independent. Now, the above assumption allows us to introduce the concept of the *rate of change* λ_{ij} , so that the probability of a change in the edge indicator X_{ij} is modelled by

$$P(X_{ij}(t+h) = 1 - x_{ij}(t) \mid \mathbf{X}(t) = \mathbf{x}(t)) = h\lambda_{ij}(\mathbf{x}(t), t) + o(h) \text{ as } h \rightarrow 0,$$

where λ_{ij} can be seen as the infinitesimal transition rate of the edge indicator X_{ij} , and this rate itself may depend on both the current time t and the current state $\mathbf{x}(t)$.

The task of modelling the dynamic random graph $\{\mathbf{X}(t) : t \in \mathbb{R}\}$ is therefore reduced to modelling the change rate $\lambda_{ij}(\mathbf{x}(t), t)$ as a function of $\mathbf{x}(t)$ and t . Any structural or non-structural covariate $T(\mathbf{x}(t))$ can be incorporated in the functional form of $\lambda_{ij}(\mathbf{x}(t), t)$. The aforementioned works by Snijders and his colleagues explore many such possibilities. See, in particular, Huisman & Snijders (2003).

2.5.2 Preferential Attachment Model

There are several dynamic graph models in which the graph grows by the gradual addition of vertices and edges. These models are called *growth models* (or, *models for network growth*), and a thorough overview of their different cases can be found in Section VII in Newman (2003). Here we only describe their general theme—the *preferential attachment* scheme popularized by Barabasi & Albert (1999). The growth procedure is designed as follows.

We begin with a single vertex v_0 and we assume (just as an exception for this first vertex) that it is adjacent to itself, so that $\deg(v_0) = 1$. Then at each step $k \geq 1$ a new vertex v_k appears and is joined with one of the existing vertices v_0, v_1, \dots, v_{k-1} . With probability $0 < p < 1$, this choice is made uniformly at random. With probability $1 - p$, this choice is made not uniformly, but proportionally to the degrees of the existing vertices.

The preferential attachment model is motivated by the frequent observation that in physical networks some already highly connected vertices are likely to become even more connected in comparison with those vertices that have smaller degrees. This is particularly relevant in the World Wide Web, where vertices represent pages and edges represent links. Another reason for the popularity of these models is that they generate vertex degree frequencies that are approximated by the power law—a feature which is widely sought out in the general field of network research (see Mitzenmacher (2004)).

2.5.3 Evolutionary vs. Transactional Models

Returning to the topic of categorizing dynamic random graphs, it is important to recognize the distinction between the *evolutionary* models and the *transactional* (or, *recurrent*) models. This distinction is most easily recognized if we think about modelling dynamic random graphs in discrete time, i.e. $t = 0, 1, 2, \dots$.

The overwhelming majority of dynamic random graphs that come up in the literature are of the evolutionary type. They include, among others, the Holland-Leinhardt-Snijders models as well as the preferential attachment model. In an evolutionary model, an edge indicator $X_{ij}(t)$ represents *accumulated* changes in some relationship up to and including time t . This means that, if $X_{ij}(0) = X_{ij}(1)$, then the state of the relationship between i and j has not undergone any physical change from $t = 0$ to $t = 1$. For example, if the relationship indicates friendship, then the states $X_{ij}(0) = 1$ and $X_{ij}(1) = 1$ represent the

same friendship between i and j that has been present for two consecutive time periods $t = 0$ and $t = 1$.

A transactional model is entirely different because an edge indicator $X_{ij}(t)$ now represents the unique and non-cumulative realization of some relationship occurring at time t . For example, suppose that the relationship represents a financial transaction between market agents. That is, $X_{ij}(t) = 1$ if and only if there was a transaction between i and j in the time period t . Then the states $X_{ij}(0) = 1$ and $X_{ij}(1) = 1$ *do not* represent the same transaction any more (contrary to the “friendship” example above), but they represent two *distinct* transactions: one at time $t = 0$, another one at time $t = 1$. In fact, the terminology of the “change of state” for an individual edge indicator $X_{ij}(t)$ does not strictly make sense under the transactional approach. If agent i made a trade A with agent j yesterday and also made a trade B with him today, an evolutionary approach would suggest that there is no change of state in their relationship—which, of course, does not make sense in this particular situation.

The two approaches require completely different ways of modelling and simulating random graphs. The evolutionary approach, as we have seen in Subsection 2.5.1, leads to the modelling of *individual changes* of edge indicators (one by one). If we were to simulate these dynamic random graphs, we would only need to choose (at random) which edge indicator X_{ij} to change from one time period t to the next time period $t + 1$. In other words, by removing an edge or adding a single edge we end up in a new state of the process. The transactional approach, on the other hand, implies that the *whole graph* has to be regenerated “from scratch” at every time period. Accordingly, the transactional approach leads to the modelling the whole graph $\mathbf{X}(t)$ at time t and not the individual changes that lead from, say, $\mathbf{X}(t - k)$ to $\mathbf{X}(t)$.

So why, then, are some real world networks evolutionary and others transactional? The answer is contained in the single property of the relationship that is being modelled—namely, whether or not it admits a *duration*. We see that evolutionary graphs arise when the modelled relationship has a duration, for example: friendships, most other social networks, the World Wide Web. Transactional graphs arise when the modelled relationship does not permit duration, but is instead interpreted as a “one-time thing”, for example: work collaborations, financial transactions, electronic communications. It is not clear why the evolutionary models predominate in the academic literature on dynamic random graphs. It is especially surprising given the availability of a large data set of a transactional-type network, namely the Enron email communication data, which can be found in Cohen

(2009).

Sometimes a transactional dynamic graph can be turned into an evolutionary one by imposing *accumulation*. For example, consider the case of cumulative collaboration, where the question becomes: have the two lawyers *ever* worked on the same case up to and including time t ? On the other hand, evolutionary graphs can be expressed in terms of transactional graphs:

$$\mathbf{X}(t) = \mathbf{X}(t-1) + \mathbf{A}(t) - \mathbf{R}(t).$$

Above, an evolutionary model $\mathbf{X}(t)$ is decomposed into two transactional processes: $\mathbf{A}(t)$ for the added edges at time t , $\mathbf{R}(t)$ for the removed edges at time t .

2.5.4 Exponential Dynamic Random Graphs

In Chapter 4 we propose a class of dynamic random graphs based on the following simple ideas.

First, we assume that $\{\mathbf{X}(t) : t = 0, 1, 2, \dots\}$ is a dynamic random graph on a fixed set of N vertices, in discrete time, and with multiple edge changes (i.e., we take the transactional approach). We then impose the Markov Chain property:

$$P(\mathbf{X}(t+1) = \mathbf{x} \mid \mathbf{X}(j) = \mathbf{x}(j), j = 0, 1, \dots, t) = P(\mathbf{X}(t+1) = \mathbf{x} \mid \mathbf{X}(t) = \mathbf{x}(t)).$$

Furthermore, we assume that the random graph $\mathbf{X}(t+1)$, conditional on the current state $\mathbf{X}(t) = \mathbf{x}(t)$, is distributed as a log-linear random graph:

$$P(\mathbf{X}(t+1) = \mathbf{x} \mid \mathbf{X}(t) = \mathbf{x}(t); \boldsymbol{\beta}) = \exp \left(\sum_{k=1}^r \beta_k T_k(\mathbf{x}, \mathbf{x}(t)) + T_0(\mathbf{x}, \mathbf{x}(t)) \right) K(\boldsymbol{\beta}, \mathbf{x}(t)),$$

where the covariates $T_k(\cdot)$ are now dependent on both the current realization $\mathbf{x}(t)$ and the future one \mathbf{x} . The immediate advantage of this approach is that we are only required to understand the properties of exponential random graphs (Sections 2.2 and 2.4) to start working with the particular cases of the above model. This approach appears in the paper by Robins & Pattison (2001) and is further elaborated by Hanneke et al. (2010).

We may choose covariates $T_k(\mathbf{x}, \mathbf{x}(t))$ in such a way that they depend structurally on the current observation $\mathbf{x}(t)$ but do *not* depend structurally on the future graph \mathbf{x} . The term “*interim* attributes” might be appropriate for these T_k ’s. The reader may recall us speaking in Chapter 1 of a structural attribute at time t playing the role of a non-structural

explanatory covariate at time $t+1$. Then, assuming complete edge independence in $\mathbf{X}(t+1)$ conditioned on $\mathbf{X}(t) = \mathbf{x}(t)$, we end up with a model for a dynamic random graph that incorporates structural attributes into probabilities, but at the same time admits edge independence, which allows us to use the classical methods of estimation instead of the ones described in Section 2.4. Balanced Centrality Markov Chains (BCMCs), to which Chapter 4 is devoted, illustrate a particular rendition of this approach. In a BCMC, the interim structural attributes are constructed from an arbitrary *centrality* index, the meaning of which we explain in the following chapter.

Chapter 3

Centrality Indices

3.1 Introduction to Centrality

Consider an arbitrary (fixed) graph $G = (V, E)$. The concept of “centrality” is an attempt to quantify the intuitive feeling that some vertices (or some edges) in G are more “important” than other vertices (or other edges). Although this insight is equally relevant both to the edges $e_{ij} \in E$ and the vertices $i \in V$, we will apply it to the vertices of a graph.

We briefly illustrate the idea of “importance” using the lawyers collaboration network. Since the full network of 72 lawyers is too large to be analysed visually, we chose to simplify our illustration by independently picking two random samples of 10 lawyers each and plotting the subgraphs induced by these samples (Figure 3.1). Looking at each of the samples, we may “pretend” that there are no lawyers other than the 10 lawyers included in the sample, so that each subgraph is viewed a complete representation of the work relationships among the sampled lawyers. We can make the following informal remarks about these samples.

- In both samples there are lawyers that are “central” and there are lawyers that are “peripheral”. For example, in Figure 3.1a vertices 2 and 5 are “peripheral” while vertices 4 and 6 are “central”, based either on an informal visual inspection or on the degree of a vertex. In Figure 3.1b the most “central” is 2 while the “peripheral” vertices are 7,8, and 4.
- We can also suggest that the “importance” of a lawyer does not merely depend on his degree (number of collaborations) but also on the “importance” of those lawyers who

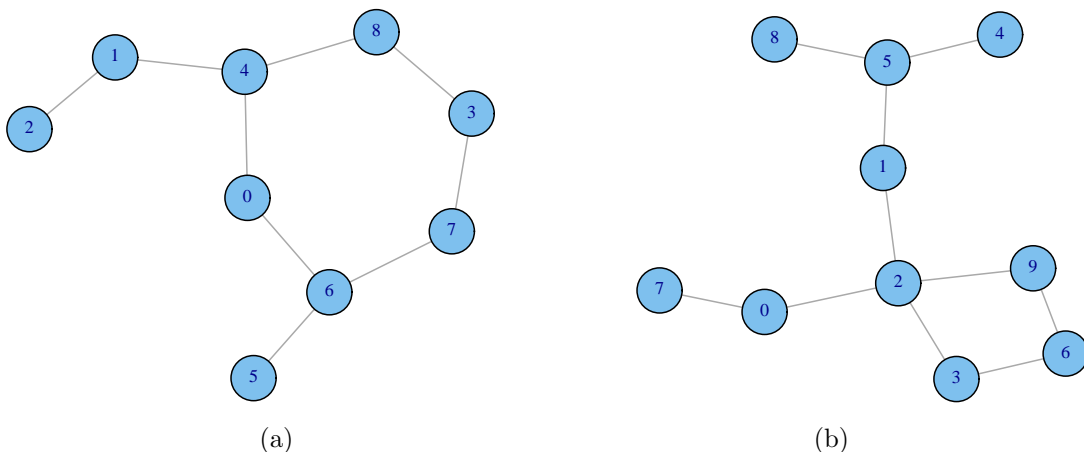


Figure 3.1: Sampled Subgraphs from the Lawyers Data.

he collaborates with. For example, in Figure 3.1a vertices 0 and 3 both have degree 2, yet vertex 3 can be judged more “important” because lawyer 0 collaborates with more “central” lawyers than those with which lawyer 3 is connected. Similarly, in Figure 3.1b vertices 0 and 1 both have degree 2, yet vertex 1 connects two “central” vertices (2 and 5) with each other and thus may be deemed more important in the network.

Intuitively, a lawyer who is “central” in the network might have more opportunities to become successful in the firm, although the actual benefits surely depend on many other variables. In other networks (for example, transportation networks) the immediate benefits of being “central” are more concrete. See, for example, the discussion of facility location problems in Section 3.3.2 of Brandes & Erlebach (2009).

3.1.1 General Centrality

Accepting the basic idea that some vertices are more central than others, suppose we are confronted with the task of building a mathematical model that describes this idea. Before proposing anything specific, we suggest taking note of the following three points:

1. In general, “importance” as well as “centrality” are relative notions, depending entirely on the context in which they are used. Therefore, we need to select a frame of reference before we propose any particular model for centrality. The concept of

centrality emerges in the analysis of networks—an applied subject, in which, unlike in the abstract graph theory, the vertex set V most commonly represents specific real objects, while the edge set E represents a relationship of a specific kind. Therefore the assertion

$$\text{“vertex } i \text{ is more important than vertex } j\text{”} \tag{1}$$

should always be interpreted on account of the real world phenomenon that G represents.

2. The model should assign a numerical value C_i to each vertex i in V , with C_i being the *centrality of i in G* , so that the mathematical statement

$$C_i > C_j$$

is interpreted as equivalent to the assertion (1). That is, centrality should reveal the relative importance of i compared to other vertices j in G .

3. The exogenous attributes (i.e., those independent of G) of a vertex i should have nothing to do with the centrality C_i of i in G because C_i should be determined entirely by the adjacencies that i holds with its neighbours, and by the adjacencies that its neighbours hold with *their* neighbours, and so on. Thus C_i is more of a function of the edge set E , than of the vertex i itself.

For example, if V is taken to be $\{0, 1, 2, \dots, n\}$ and a vertex $i = 0$ is chosen, then $C_i = C_0$ should have no association with the fact that $i = 0$. If V were to be *relabelled* as $\{-1, 0, 1, \dots, n-1\}$, then C_{-1} after relabelling would be equal to C_0 before relabelling.

In mathematical terms, centrality C_i has to be invariant under graph isomorphisms¹ Koschutzki et al pose a requirement in the edited volume of Brandes & Erlebach (2009) that any centrality measure should be a *structural index*, which is defined as any function $s : V \rightarrow \mathbb{R}$ that is invariant under graph isomorphisms.

It follows from the first point above that there is no formal definition of a generic centrality measure. There are many ways to assign centralities to vertices, each of them being suitable to different situations. A *centrality index* can be defined as a way to assign centralities C_i to vertices i in G , conforming to the three principles above, i.e., it has to be interpretable and it has to be invariant under isomorphisms. Whenever we want to

¹We assume that the reader is familiar with the concept of graph isomorphisms and isomorphic graphs.

propose a centrality index, we need to “match” it with a real-world (application-based) interpretation. The analysis of real-world networks (in particular, social networks) provides a range of possible interpretations of centrality, which can be roughly grouped into three categories:

- rank/prestige/status;
- capacity to exert influence/power on others;
- capacity to transfer information/goods/substance.

In principle, one could recommend a centrality index with a purely combinatorial and/or purely geometric interpretation. However, the primary goal of this thesis is to suggest probability models that use centrality indices to explain the dynamic behaviour of real-world networks. Thus our emphasis is on the applied, network-based interpretations of centrality.

3.1.2 Degree Centrality

To see how the three principles apply to a specific index, consider a very simple centrality index that coincides with the degree of a vertex in an undirected graph:

$$C_i := \deg(i), \quad i = 1, \dots, N.$$

It is clear that C_i is invariant under isomorphisms (independent of vertex labels). Thus, it suffices to check that C_i is interpretable. Due to its simplicity, the interpretation of the degree centrality index is straightforward: the greater the proportion of edges incident to i , the more prominent i is in the graph. In more applied terms, the more relationships directly involve the node i , the more important it is in the network of interest.

Note that we refer to $\deg(i)$ as a very simple centrality index because it accounts only for the immediate neighbours of a vertex i . More elaborate centrality indices incorporate the additional information about edges that are *not* incident to i .

3.1.3 Normalization

Regardless of the interpretation involved, it is reasonable to expect from any centrality index that a vertex i attains the *maximum* possible centrality if i is adjacent to *all* other

vertices. See, for example, the star graph shown in Figure 3.2. In this example, vertex i should attain the maximum centrality that could occur in an arbitrary graph on 8 vertices. If this is *not* the case for a particular index C_i , then this index is, most likely, not an appropriate measure of centrality.

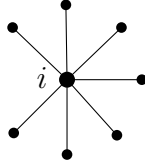


Figure 3.2: A star graph.

It is convenient to normalize centrality indices so that C_i belongs to $[0, 1]$ for each i . With normalization, $C_i = 1$ usually represents the maximum centrality that can be obtained in an arbitrary graph on a given vertex set. This approach allows one to compare the centralities of vertices from different graphs, as well as to compare different centrality indices applied to the same vertex. For example, the degree centrality index can be normalized by letting $C_i := \frac{\deg(i)}{N-1}$, where N is the total number of vertices in the graph. In this case, the maximum centrality $C_i = 1$ is obtained if and only if i is adjacent to all vertices.

In this thesis, we focus on two most common classes of centrality indices (geodesic and eigenvalue-based centralities) with the goal of using them to model dynamic random graphs. Thus, we omit some classes of centrality indices, such as those based on information, network flows, and random walks. For an introductory overview of common topics, the reader may first refer to Chapter 5 in Wasserman & Faust (1994). After that, the reader may continue with Chapters 3–5 in Brandes & Erlebach (2009) for a more thorough examination of the subject.

3.2 Eigenvector Centrality

Eigenvalue-based centrality (proposed by Bonacich (1972)) emerged in social network analysis as an attempt to capture the notion of *prestige* (rooted in Katz (1953)), which asserts that a vertex is as central as its neighbours. Suppose that G is an undirected graph and that C_i is a centrality index that follows the idea that a vertex is as central as its neighbours. The mathematical implication is that C_i is proportional to the aggregate centrality

of i 's neighbours:

$$C_i = \psi \sum_{j \in N(i)} C_j, \quad (2)$$

where ψ is an unidentified (for the moment) constant. Suppose that vertices j_1, j_2, \dots, j_n comprise the neighbourhood of vertex i , i.e.,

$$N(i) = \{j_1, j_2, \dots, j_n\},$$

then

$$\begin{aligned} C_i &= \psi C_{j_1} + \dots + \psi C_{j_n} = \psi \left\{ \psi C_i + \psi \sum_{\substack{k \in N(j_1) \\ k \neq i}} C_k \right\} + \psi C_{j_2} + \dots + \psi C_{j_n} \\ &= \psi^2 C_i + \text{other terms.} \end{aligned} \quad (3)$$

The above expression shows that C_i appears recursively in its own computation. Thus if we assume non-negative centralities with at least one positive index $C_j > 0$, then the recursive appearance of C_i in (3) is possible only if $0 < \psi \leq 1$.

Definition (2) can be expressed as

$$C_i = \sum_{j=1}^N \psi x_{ij} C_j.$$

Joining all the centralities C_i in vector form,

$$\mathbf{c} := \begin{pmatrix} C_1 \\ C_2 \\ \vdots \\ C_N \end{pmatrix},$$

we can write

$$\mathbf{c} = \psi \begin{pmatrix} \sum_{j=1}^N x_{1j} C_j \\ \sum_{j=1}^N x_{2j} C_j \\ \vdots \\ \sum_{j=1}^N x_{Nj} C_j \end{pmatrix} = \psi \begin{bmatrix} 0 & x_{12} & \cdots & x_{1N} \\ x_{21} & 0 & \cdots & x_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & 0 \end{bmatrix} \begin{pmatrix} C_1 \\ C_2 \\ \vdots \\ C_N \end{pmatrix} = \psi \mathbf{x} \mathbf{c}. \quad (4)$$

Let $\lambda := \psi^{-1}$, then $\lambda \geq 1$ and

$$\mathbf{x} \mathbf{c} = \lambda \mathbf{c}, \quad (5)$$

which means that \mathbf{c} is a non-negative (by assumption) eigenvector of the adjacency matrix \mathbf{X} , corresponding to an eigenvalue $\lambda \geq 1$.

Although the eigenvector centrality index follows the basic motivation to assign centralities that are proportional to the aggregate “neighbouring” centralities, it is very difficult to directly interpret the exact numeric values C_i . Indeed, for every appropriate eigenvalue λ there are infinitely many eigenvectors (centrality vectors) \mathbf{c} that satisfy (5). (Although often the one with unit Euclidean norm is chosen in practice.) Moreover, it is hard to interpret the difference between using one eigenvalue λ_1 versus another one λ_2 . (Although often the largest eigenvalue λ_{\max} is chosen in practice.) We conclude that the eigenvector centrality may be criticized for being an “arbitrary” index whose exact numeric values do not have a physical meaning. On the other hand, this feature of eigenvector centrality actually corresponds to the perceived arbitrariness of the socioeconomic notion of “prestige” (the latter represented by C_i). Indeed, any attempt at a formal definition of “prestige” would result in a mere approximation of its real meaning, even though the real meaning is understood by almost everyone on the intuitive level. Ultimately, our interpretation of an eigenvector centrality index C_i can be as vague as the socioeconomic concept of “prestige” that C_i represents. It is also difficult to normalize eigenvector centrality index, therefore direct numeric comparison with other graphs and other centrality indices is difficult.

For directed connected graphs, we may use either *forward* or *backward* eigenvector centralities, both of which are constructed by analogy with (2), in particular:

$$\begin{aligned} C_i^{out} &= \psi_{out} \sum_{j \in N_{out}(i)} C_j^{out} = \psi_{out} \sum_{j \neq i} x_{ij} C_j^{out}, \\ C_i^{in} &= \psi_{in} \sum_{j \in N_{in}(i)} C_j^{in} = \psi_{in} \sum_{j \neq i} x_{ji} C_j^{in}, \end{aligned}$$

where C_i^{out} represents the forward eigenvector centrality of vertex i and C_i^{in} represents the backward eigenvector centrality of vertex i . The corresponding vector versions \mathbf{c}_{out} and \mathbf{c}_{in} are the solutions to the eigenvector equations

$$\mathbf{x} \mathbf{c}_{out} = \psi_{out}^{-1} \mathbf{c}_{out} \quad \text{and} \quad \mathbf{x}^T \mathbf{c}_{in} = \psi_{in}^{-1} \mathbf{c}_{in}.$$

Lastly, we discuss the extension of eigenvector centrality to disconnected graphs. The adjacency matrix \mathbf{x} of a disconnected (undirected) graph can be rearranged (by relabelling

the vertices) as a *block* matrix

$$\mathbf{x} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_b \end{bmatrix},$$

where each block \mathbf{A}_k corresponds to a component in the graph. Hence there are b components in total. Suppose that for each of the blocks (subgraphs) \mathbf{A}_k we can find an eigenvalue $\lambda_k > 1$ and its corresponding eigenvector \mathbf{c}_k . We see that vector \mathbf{c}_k contains the eigenvalue centralities C_i for vertices i in k 'th component, since expression (2) definitely holds for vertex i and $\psi = \lambda_k^{-1}$. Thus the composite vector

$$\mathbf{c} = \begin{pmatrix} \mathbf{c}_1 \\ \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_b \end{pmatrix}$$

contains the eigenvalue centralities for the whole graph \mathbf{x} . This centrality index has the same interpretation problems as the version for connected graphs. There is an additional difficulty in justifying the differences between the constants λ_k^{-1} corresponding to different components. These constants maybe unequal, and the following question can be posed: why should we set C_i to λ_1^{-1} times the sum of neighbouring centralities for vertex i in component 1 but set C_j to λ_2^{-1} times the sum of neighbouring centralities for vertex j in component 2?

3.3 Geodesic Centralities

Two of the most widely used centrality indices, *centrality* and *betweenness*, are based on *geodesic paths*. We begin with the introduction of this important graph-theoretic concept.

3.3.1 Connectivity in Graphs

Formally, a *path* is defined as a chain

$$v_1 \ e_{v_1 v_2} \ v_2 \ e_{v_2 v_3} \ \cdots \ v_{k-1} \ e_{v_{k-1} v_k} \ v_k, \quad (6)$$

in which vertices v_i alternate with edges $e_{v_i v_j}$ according to the following rules:

1. all vertices v_i are distinct and all edges $e_{v_i v_j}$ are distinct;
2. the path starts with a vertex and ends with a vertex;
3. each vertex v_i , except the last one v_k , is followed by an edge;
4. each edge $e_{v_i v_{i+1}}$ is preceded by the vertex v_i and followed by the vertex v_{i+1} (both of which are incident to the edge $e_{v_i v_{i+1}}$).

The *length* of a path is given by the number $(k - 1)$ of edges it contains. The path in (6) is said to *join* vertex v_1 and v_k . In case of a directed graph, we can also say that the path follows *from* v_1 *to* v_k .

Recall that there are no multiple edges in our graphs. In case of an undirected graph, there may be at most one edge $\{u, v\}$ incident to any pair u, v of vertices. In case of a directed graph, there may be at most one edge (u, v) emanating from a vertex u and arriving at a vertex v . Therefore the explicit inclusion of edges $e_{v_i v_{i+1}}$ in the presentation (6) is redundant. Instead, we will use a shorter presentation of a path:

$$v_1 \ v_2 \ \cdots \ v_{k-1} \ v_k,$$

which unambiguously *implies* the formal presentation in (6). As a caution, we have to note that an ordered tuple $(v_1, v_2, \dots, v_{k-1}, v_k)$ of distinct vertices from V represents a path in a given graph $G = (V, E)$ if and only if every implied edge actually belongs to the edge set E .

We illustrate these concepts with an example. Refer to the graph shown in Figure 3.3a. Bold lines represent the path

$$AFDEBC$$

joining A and C . The length of this path is 5. The path AC is a shorter path (of length 1) joining the same vertices, shown in Figure 3.3b. In fact, of all paths that join A and C , the path AC is the shortest one. We say that the path AC is a *geodesic* path. In general, not all vertices can be joined by a path. Consider the graph shown in Figure 3.4. There are no paths joining B to F , or C to D . Such graph is said to be *disconnected*, and the formal definition to this term will be given later in this section.

Let $G = (V, E)$ be an undirected graph. The [*geodesic*] *distance* $d(u, v)$ between two vertices u and v is defined as either (a) the length of the shortest path joining u and v , if they can be joined by at least one path in G , or (b) infinity (∞) if no such path exist. A

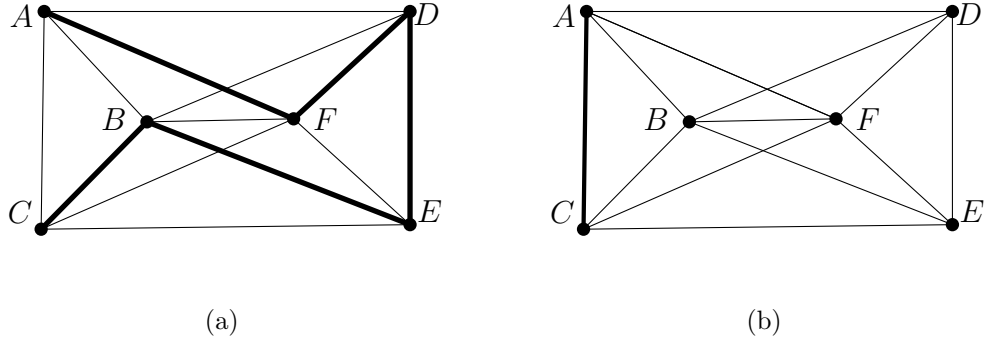


Figure 3.3: Paths in an undirected graph.

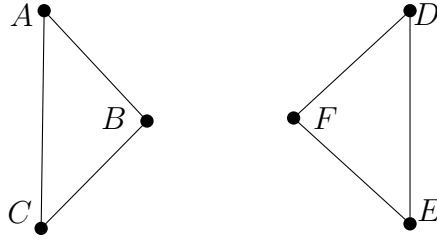


Figure 3.4: A disconnected graph.

path $(u, x_1, x_2, \dots, x_{d-1}, v)$ is said to be *geodesic* if its length is equal to the distance d between u and v . There may be several geodesic paths joining a given pair of vertices, as illustrated in Figures 3.5a to 3.5c.

An undirected graph G is said to be *connected* if every pair u, v of vertices can be joined by a path. In this case, all the distances $d(u, v)$ are finite. Moreover, it is easily checked that $d(u, v)$ is a metric on the vertex set V , with the convention that $d(v, v) = 0$ for any v . If G is not connected, then it is *disconnected*. A disconnected graph (Figure 3.4) is always comprised of components. A [*connected*] *component* is defined as a subgraph $H = (V_H, E_H)$ of G with the following properties:

1. H is connected;
2. No edge e_{ij} in the original edge set E joins a vertex e in V_H with a vertex j outside V_H .

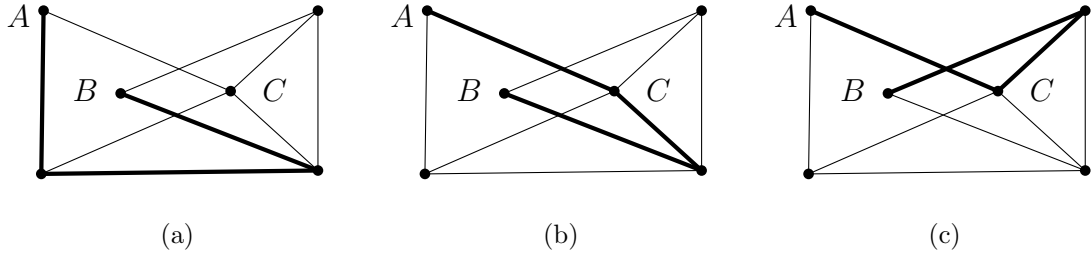


Figure 3.5: Geodesic Paths.

The computational aspects of finding geodesic paths, calculating geodesic distances, and finding components are outside the scope of this thesis. The concepts of this section suffice to introduce centrality indices that are based on graph geodesics.

3.3.2 Closeness

Geodesic centralities emerged in the early works by Hakimi (1965) and Sabidussi (1966), in which the concepts of closeness centrality were first introduced. Let G be an undirected connected graph on $N \geq 2$ vertices. We saw that the geodesic distance $d(i, j)$ is a metric on the vertex set V . *Closeness* centrality index (first proposed by Sabidussi (1966), and also popularized by Freeman (1979)) asserts an inverse relationship between centrality C_i and the *total* distance between i and all other vertices. It is thus defined as

$$C_i := \frac{1}{\sum_{j \in V} d(i, j)}, \quad i \in V.$$

We can modify closeness centrality to vary inversely with the *average* distance between v and other vertices, the result being

$$C_i := \frac{N - 1}{\sum_{j \in V} d(i, j)}, \quad i \in V. \quad (7)$$

The latter index is preferred to the former one because it is normalized. Indeed, $C_i = 1$ if and only if i is adjacent to every other vertex in G .

If G is disconnected, then the sum $\sum_{j \in V} d(i, j)$ is undefined for every i in V , because $d(i, j) = \infty$ for at least one pair i, j of vertices. There are several approaches to extend

closeness centrality index to disconnected graphs. Suppose that G consists of k components, denoted G_1, G_2, \dots, G_k . Let V_1, V_2, \dots, V_k be their corresponding vertex sets, let N_1, N_2, \dots, N_k be their corresponding vertex counts, and let M_1, M_2, \dots, M_k be their corresponding edge counts.

The first idea is to compute closeness centralities using (7) separately for each component G_h , and then scale these centralities proportionally to the component's size. The caveat here is that there are two ways to define the relative size of G_h : we could use either the vertex count ratio N_h/N or the edge count ratio M_h/M , where N and M are, respectively, vertex and edge counts in the whole graph G . We now discuss both of these approaches and then make side-by-side comparison of the resulting centralities for the graph G shown in Figure 3.7.

For each vertex i belonging to the component G_h , compute the *v-weighted closeness* centrality of i by the formula

$$C_i := \begin{cases} \left(\frac{N_h}{N} \right) \frac{N_h - 1}{\sum_{j \in V_h} d(i, j)} & \text{if } N_h \geq 2, \\ 0 & \text{if } N_h = 1. \end{cases}$$

The special case $N_h = 1$ has to be set apart from the case $N_h \geq 2$ to avoid division by zero. By analogy, the *e-weighted closeness* centrality of i is

$$C_i := \begin{cases} \left(\frac{M_h}{M} \right) \frac{N_h - 1}{\sum_{j \in V_h} d(i, j)} & \text{if } N_h \geq 2, \\ 0 & \text{if } N_h = 1. \end{cases}$$

Observe that both of these indices reduce to the non-weighted closeness centrality (7) when G is connected. Moreover, both of them lie between 0 and 1. However, the extent

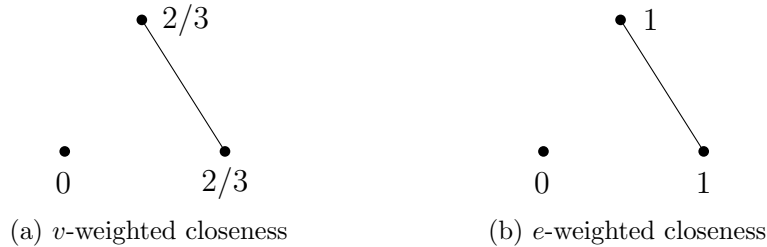


Figure 3.6: Weighted Closeness.

of normalization is different for each of the indices. To see this, note that a v -weighted closeness $C_i = 1$ implies that i is connected to every other vertex in the whole graph G , i.e., G is connected. However, an e -weighted closeness $C_i = 1$ does *not* imply that G is connected (as illustrated by Figure 3.6). In other words, a v -weighted closeness C_i equals 1 if and only if i is connected to all vertices, while for an e -weighted closeness the latter condition is sufficient but not necessary for C_i to equal 1.

To compare these two ways of assigning closeness weights, refer to Table 3.1 that lists centralities in graph G shown in Figure 3.7. The main point to take from this comparison is not the fact that the two methods produce different numeric values, but the fact that the induced *ordering* on V is different between the two approaches. For example, vertex 6 has the highest v -weighted closeness index, while vertices 2, 3, 4, 5 have the highest e -weighted closeness index. Based on intuition only, it seems impossible to justify a preference to one case or the other.

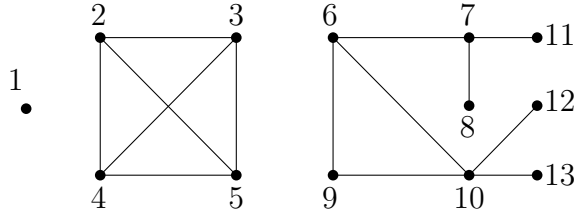


Figure 3.7: Graph G .

Another method of extending closeness centrality entails changing the definition of distance between disconnected vertices i and j from $d(i, j) = \infty$ to $d(i, j) = N$. Since no two vertices in any graph can have distance larger than $N - 1$, it follows that $d(i, j) = N$ if and only if i and j belong to different components. This tweak allows using formula (7) for a disconnected graph. We refer to the resulting centrality measure as *capped closeness*. Table 3.1 includes the capped closeness centralities for graph G shown in Figure 3.7. Again, we find that capped closeness centralities produce an ordering that is different from both of the two weighted closeness centralities.

	1	2, 3, 4, 5	6	7	8	9	10	11	12	13
within component	0.00	1.00	0.70	0.54	0.37	0.50	0.58	0.37	0.39	0.39
<i>v</i> -weighted	0.00	0.31	0.43	0.33	0.23	0.31	0.36	0.23	0.24	0.24
<i>e</i> -weighted	0.00	0.43	0.40	0.31	0.21	0.29	0.33	0.21	0.22	0.22
capped	0.08	0.11	0.17	0.17	0.15	0.16	0.17	0.15	0.16	0.16
<i>v</i> -weighted	<p>Vertices ordered by increasing closeness:</p> <p>1, (8, 11), (12, 13), (2, 3, 4, 5, 9), 7, 10, 6.</p> <p>1, (8, 11), (12, 13), 9, 7, 10, 6, (2, 3, 4, 5).</p> <p>1, (2, 3, 4, 5), (8, 11), (12, 13), 9, 7, 10, 6.</p>									
<i>e</i> -weighted										
capped										
	<i>Note:</i> vertices with equal centralities are grouped in parentheses.									

Table 3.1: Closeness indices applied to Graph G in Figure 3.7.

3.3.3 Betweenness

Betweenness is a centrality index that is also based on the geodesic distances in a graph but is very different from closeness. We now describe the procedure of computing betweenness centrality C_i for some fixed vertex i (in an undirected graph). For every pair k, l of vertices distinct from i ($k \neq i, l \neq i$) we compute their contribution $p_{kl}(i)$ to the betweenness of i as follows.

- If there are no paths connecting k to l , we set $p_{kl}(i) = 0$.
- If there is at least one geodesic path connecting k and l , we set $p_{kl}(i)$ as the proportion of the geodesic paths between k and l that contain vertex i .

For example, looking at the graph in Figure 3.5 we see three geodesic paths between vertices A and B , two of which contain vertex C . Therefore the contribution of A and B to the betweenness of C is $p_{AB}(C) = 2/3$.

We then define *betweenness* C_i of vertex i as the sum

$$C_i := \sum_{\substack{k, l \\ k, l \neq i}} p_{kl}(i).$$

The value C_i can thus be interpreted as the total “level of involvement” of vertex i in the geodesic paths of the graph. In other words, the more geodesic paths contain i , the more

central this vertex is. In fact, betweenness centrality is most useful when the edges e_{ij} in the graph represent the passage of information or goods between two vertices i and j . The contribution $p_{kl}(i)$ is then interpreted as the probability that information or goods will pass through vertex i when sent from k to l (or from l to k), assuming that all of the geodesic paths between k and l are given an equal chance of being chosen for the transfer. Note that it is possible to modify betweenness centrality to account for *all* paths in the graph (not just the geodesic paths)—see White & Smyth (2003).

Betweenness can be normalized in the usual way after we derive the largest betweenness that may occur in an arbitrary graph on a fixed set of N vertices. Suppose that vertex i that is adjacent to every other vertex in the graph. If such a vertex exists, then the distance between every pair $\{k, l\}$ of vertices distinct from i is exactly 2. If we further suppose that none of the remaining vertices are adjacent (see the star graph in Figure 3.2) then $p_{kl}(i) = 1$ for every pair k, l . Thus $C_i = \binom{N-1}{2}$ is the largest attainable betweenness with a fixed set of N vertices. The normalized version of betweenness can then be defined as

$$C_i = \frac{1}{\binom{N-1}{2}} \sum_{\substack{k, l \\ k, l \neq i}} p_{kl}(i).$$

3.3.4 Medians and Other Geodesic Indices

Here we briefly introduce the concept of a *median* vertex in a graph and its relation to centrality. In fact, there is no unique way to define medians in graphs. There are several ways to accommodate the notion of the median from one-dimensional data analysis to the notion of a graph median using the geodesic paths. One approach is discussed by Hakimi (1965) as well as Minieka (1977), who use the definition of a graph median originating from the optimal facility location problem. The median is then a vertex i_m with the property

$$\sum_{i=1}^N d(i, i_m) \leq \sum_{i=1}^N d(i, j), \quad j = 1, 2, \dots, N. \quad (8)$$

It is not difficult to see that a median i_m has the largest closeness centrality in the graph.

A different approach to defining the median is taken by Small (1997), who uses the concept of depth function to rank the vertices and define their median. The depth function itself is constructed using the concept of *geodesic convexity* and may be used as a centrality index in a graph. The rankings and the medians obtained this way are related to betweenness, although certainly not in the way as obvious as the previous notion in (8)

is related to closeness. The algorithm that computes betweenness and the algorithm that computes the depth function both require a subroutine that traverses through the geodesic paths that contain a specific vertex. In other words, the numerical values of the depth and the betweenness of a vertex both closely depend on some analysis of the specific geodesic paths that contain that vertex. Given that betweenness and depth function require similar algorithms to obtain similar goals, it might be interesting to see how often these approaches induce the same ranking on the vertices.

3.4 Centrality Indices and the Lawyers Network

Here we compare three centrality indices (eigenvector, closeness, betweenness) computed for the lawyer's collaboration network. After removing a vertex that has no connections (a lawyer who has not collaborated with anyone), the graph becomes connected, thus our analysis of centrality is somewhat simplified. Figure 3.8 presents the histograms for the eigenvector, closeness, and betweenness centralities. For closeness and betweenness we computed their normalized versions. For the eigenvector centralities we use the largest eigenvalue and its corresponding eigenvector with unit Euclidean norm.

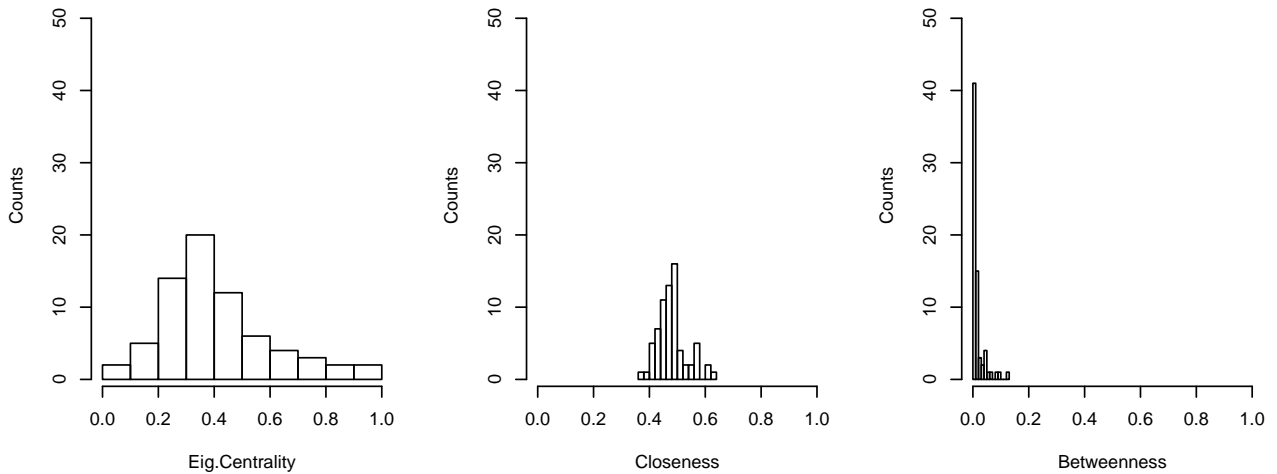


Figure 3.8: Centrality indices applied to Lawyers Data.

Looking at the histograms we gain additional insight into the concept of centrality. The three histograms are plotted on the same scale, and we see immediately that the eigen-

vector centrality has the largest standard deviation, which gives us a basis to differentiate lawyers by their “prestige” in the collaboration network. Neither closeness nor betweenness give us a better basis to differentiate between lawyers since these centrality indices are spread within a much smaller range of values. In fact, we propose the following informal explanation to this phenomenon. Both closeness and betweenness are based on the premise that shortest (geodesic) paths in the graph add to the centralities of those vertices that are involved in these paths. However, in the case of collaboration network, it is not easy to assign physical meaning to shortest paths (contrary to, say, a transportation network, in which shortest paths correspond to “accessibility”). Therefore interpretation of closeness and betweenness indices is not a straight-forward task in this particular application. Eigenvector centrality, on the other hand, is more appropriate in this application despite the fact that our choice to use the largest eigenvalue with unit length eigenvector can be criticized for arbitrariness. Indeed, a lawyer’s “prestige” in the firm involves not just the number of co-workers who he collaborates with, but also the corresponding “prestige” values of these co-workers.

To conclude this chapter we emphasize again that different centrality indices may be more appropriate than others in different situations—depending on the particular physical network that the graph represents.

Chapter 4

Balanced Potential Models

In this chapter we propose the class of Balanced Potential Models (BPMs) and discuss their properties extensively. Roughly speaking, these models are based on exogenous (non-structural) edge attributes, whose total value is optimized with respect to a constraint on the graph’s density. Some variations of these models appear in the literature, but complex structural covariates are typically included in addition to the non-structural edge attributes (see, e.g., Section 6.5.4 in Kolaczyk (2009) and Section 5 in Snijders et al. (2006)). The models that we propose here have never been identified as a special class, hence their properties have not been carefully explored. This inattention is not surprising considering the preoccupation of network researchers with the goal of using structural rather than non-structural attributes to explain network behaviour. The BPMs do not attain this goal when applied exclusively to static random graphs. However, the BPMs do attain this goal to some extent when applied to dynamic random graphs with suitably defined *interim* covariates (recall the brief exposition of Subsection 2.5.4). This idea is distinctly clarified by the specialized example of Balanced Centrality Markov Chains (BCMCs). Within this thesis, our recommendation of BCMCs may be seen as the ultimate purpose, at which all previous discussions were aimed. Informally, the BCMCs are based on the presumption that vertices with high centrality (at time t) are attracted to each other and therefore should be given an increased chance of becoming adjacent in the following time period $(t + 1)$, subject to a certain constraint on the graph’s expected density.

Since the BCMCs grow out of the Balanced Potential Models, we need to follow the complete path from the concept of a “potential” to the motivation and the prototype model for a BPM, to the formal definition of a BPM and the review of its properties, and finally to the formal definition of a BCMC.

4.1 Balanced Potential Model

4.1.1 Introducing Potentials and the Prototype Model

Suppose that \mathbf{X} is a random graph (either directed or undirected) on N vertices. Suppose that f is a function that assigns a real value $f(i, j)$ to each ordered pair (i, j) of vertices. If \mathbf{X} is an unordered random graph, then we assume that $f(i, j)$ is symmetric in its arguments i and j . From the perspective of statistical analysis, $\{f(i, j) : i, j = 1, \dots, N\}$ can be viewed as the set of exogenous edge covariates. These covariates are exogenous to \mathbf{X} because they do not depend on the structure of \mathbf{X} but are supplied externally and assumed to be known. Alternatively, for edge e_{ij} the value $f(i, j)$ can also be viewed as the *weight* of this edge. Then the random variable

$$S(\mathbf{X}) := \sum_i \sum_j X_{ij} f(i, j)$$

represents the total weights observed in \mathbf{X} .¹

Consider the exponential random graph model for \mathbf{X} given by the distribution

$$P(\mathbf{X} = \mathbf{x}; \beta) = \exp(\beta S(\mathbf{x})) K(\beta), \quad \mathbf{x} \in \Omega. \quad (1)$$

This model is the “prototype” of the BPMs, and we will keep referring to it as such.

Following the global interpretation of exponential random graphs (Subsection 2.2.2), we see that this model assigns high probabilities to outcomes \mathbf{x} with large total weights whenever $\beta > 0$. If, on the other hand, $\beta < 0$, then high probabilities are assigned to the graphs \mathbf{x} with small total weights.

For a local interpretation (Subsection 2.2.3), we compute the change function

$$\delta_{ij}(\mathbf{x}_{ij}^c) = S(\mathbf{x}_{ij}^+) - S(\mathbf{x}_{ij}^-) = f(i, j).$$

It follows that the conditional odds

$$\pi_{ij}(\mathbf{x}_{ij}^c; \beta) = e^{\beta f(i, j)} := \pi_{ij}(\beta)$$

¹In the sum $\sum_i \sum_j \mathbf{X}_{ij} f(i, j)$ the ranges of vertex indices i and j depend on the underlying sample space Ω of the random graph \mathbf{X} . If \mathbf{X} is undirected, then $1 \leq i, j \leq N$ ($i \neq j$); if \mathbf{X} is directed, then $1 \leq i < j \leq N$. Throughout this chapter, the same clarification applies to all other sums and products of this form.

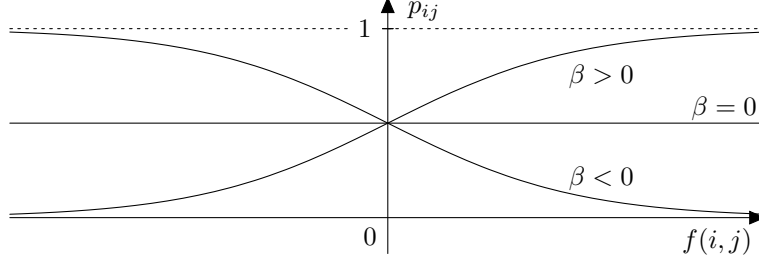


Figure 4.1: Inclusion rate p_{ij} as a function of one exogenous edge weight $f(i, j)$ for different values of β .

do not depend on the complement \mathbf{x}_{ij}^c of x_{ij} , but only on the weight $f(i, j)$ assigned by f to the edge e_{ij} . As a consequence, we assert complete edge independence (as discussed in Subsection 2.2.4) and we compute the inclusion rates $p_{ij}(\beta)$ as

$$p_{ij}(\beta) = \frac{\pi_{ij}(\beta)}{1 + \pi_{ij}(\beta)} = \frac{e^{\beta f(i, j)}}{1 + e^{\beta f(i, j)}}. \quad (2)$$

Figure 4.1 illustrates the behaviour of p_{ij} as a function of the edge weight $f(i, j)$. We see that, given a positive value of β , an edge e_{ij} with a large weight $f(i, j)$ is more likely to occur than an edge e_{kl} with a smaller weight $f(k, l)$. In fact, the function $f(i, j)$ can be thought to represent *mutual attractiveness* between a vertex i and a vertex j . Thus, if the attractiveness between i and j is positive, the edge e_{ij} is more likely to occur than not. On the other hand, if the attractiveness between i and j is negative then e_{ij} is more likely to *not* occur. Finally, an attractiveness value of 0 represents indifference: the edge e_{ij} is equally likely to occur and not to occur.

When β is negative, the function $f(i, j)$ admits an opposite meaning. In this case, it represents *mutual repulsion*: positive repulsion $f(i, j) > 0$ implies an unlikely occurrence of e_{ij} , negative repulsion $f(i, j) < 0$ implies a likely occurrence of e_{ij} , and neutral repulsion $f(i, j) = 0$ implies a 50% of occurrence.

As will be clarified by several upcoming examples, the idea of “pre-determined” attractiveness and repulsion weights can explain, on the intuitive level, the behaviour of real world networks, in particular: social networks, economic transaction networks, and some physical networks.

Having thus explained the basic motivation behind the prototype model in (1), we would like to impose a somewhat personal modification to our notation and terminology. First, instead of the overly suggestive terms “mutual attractiveness” and “mutual

repulsion”, we are inclined to use, correspondingly, the less colourful terms *potential* and *antipotential*. Second, we will use the symbol s_{ij} instead of $f(i, j)$ to denote the potential (or antipotential) attributed to the edge e_{ij} . The graph statistic

$$S(\mathbf{X}) := \sum_i \sum_j X_{ij} s_{ij} = \sum_{e_{ij} \in E(\mathbf{X})} s_{ij}$$

will be called the *total realized potential* (or total realized antipotential, depending on the context) of \mathbf{X} . If $s_{ij} = 0$, we say that s_{ij} is a *neutral* potential (or antipotential).

Looking at the identity (2), we realize that the inclusion rate $p_{ij} = P(X_{ij} = 1)$ does not depend on i and j , but only on s_{ij} . Accordingly, we will denote by $p(s_{ij})$ the inclusion rate as a function of the potential (or antipotential) s_{ij} :

$$P(X_{ij} = 1) = p(s_{ij}) = \frac{e^{\beta s_{ij}}}{1 + e^{\beta s_{ij}}}. \quad (3)$$

We will often refer to $p(\cdot)$ as the *rate function*.

Now that the basic concepts and terminology have been introduced, we move on to the discussion of our model’s drawbacks.

4.1.2 Shortcomings of the Prototype Model

Assume that $\beta > 0$ and let s_{ij} be the edge potentials. Informally speaking, the prototype model in (1) aims at maximizing the total realized potential (i.e., it assigns high probabilities to the outcomes \mathbf{x} that have large total realized potentials $S(\mathbf{x})$). Similarly, if $\beta < 0$ and s_{ij} are antipotentials, then (1) can be said to be minimizing the realized antipotential.

Our prototype model has two notable drawbacks, both of which can be eliminated with one simple modification. Let us first explain what these drawbacks are.

1. Model (1) assigns the highest probabilities to “extreme” outcomes. For example, when s_{ij} are edge potentials and $\beta > 0$, the N -complete graph² \mathbf{x}_{max} receives the highest probability because such a graph attains the highest possible value of $S(\mathbf{x})$. We also see that, unless the majority of potentials are neutral, model (1) will tend to generate graphs with high density. In practice, networks tend to have low edge density.

²A graph on N vertices in which *all* vertices are adjacent.

On the other hand, when s_{ij} are antipotentials and $\beta < 0$, then the highest probability is assigned to the empty graph $\mathbf{0}$ (with no edges) because such a graph has the smallest value of $S(\mathbf{x})$. Consequently, model (1) will tend to generate graphs with low edge density.

We conclude that an improvement to the prototype model needs to incorporate control for the edge density.

2. As seen in Figure 4.1 as well as in the identity (3), a neutral potential $s_{ij} = 0$ will always result in a 50% inclusion rate. If we define the *point of indifference* as the value s^* such that $p(s^*) = 0$, then clearly 0 is the point of indifference in our prototype model. But the point of indifference rarely occurs exactly at 0 in real situations. In practice (as the example in the forthcoming subsection will suggest), the point of indifference is often shifted to the right from the origin.

A related observation is the following one. Assuming that all the potentials are neutral (i.e., $s_{ij} = 0$ for every edge e_{ij} , the resulting random graph has an $G_{N,0.5}$ distribution (recall Subsection 2.3.1). The expected density of this graph is just the inclusion probability $p = 0.5$, which is already too high compared to many real networks.

Of course, we can make analogous observations about the case in which $\beta < 0$ and the s_{ij} 's represent antipotentials. We conclude that an improvement to the prototype model needs to incorporate control for the point of indifference s^* .

We feel that our solution to the above problems will be better understood if we derive it through a hypothetical example.

4.1.3 Modifying the Prototype Through an Example

Consider a set V of N market participants (agents) who make weekly economic transactions with each other. Let $X_{ij} = 1$ if agent i decides to initiate a transaction with agent j next week. (In other words, suppose that $X_{ij} = 1$ if agent i makes a phone call to agent j with the intention to buy goods from j .) Otherwise, we let $X_{ij} = 0$. Presumably, there is some uncertainty in the individual decisions of the agents, hence we may represent this situation with a (directed) random graph \mathbf{X} , the X_{ij} 's ($i \neq j$) being its edge indicators.

Suppose that the agents keep track of their past transactions, and let s_{ij} represent the total profits accumulated by agent i as the result of all past transactions with agent

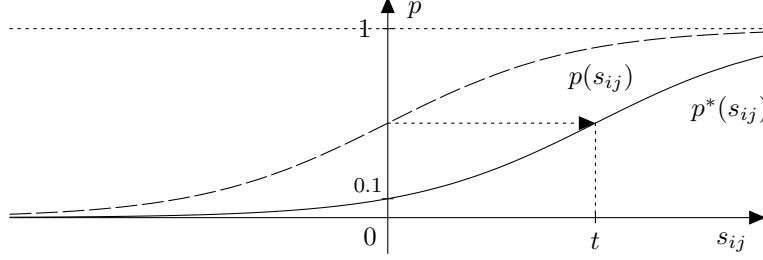


Figure 4.2: The shift of the rate function by t units.

j . Clearly, the s_{ij} 's represent edge potentials in our terminology. Indeed, a large value $s_{ij} > 0$ of the cumulative profits should motivate agent i to trade with agent j again. On the other hand, a net loss $s_{ij} < 0$ should discourage agent i to trade with j again. One could suggest to employ the prototype model (1) (with $\beta > 0$) as a representation of this simplified marketplace. Now, the prototype model implies that, given a neutral economic history $s_{ij} = 0$ for i and j , agent i is as willing to trade with j as to not to trade with him. It is reasonable to contend, however, that this should not be the case in a conservative economic environment. We are thus willing to modify the inclusion rate function $p(\cdot)$ in such a way that accounts for a more conservative economic environment. We will denote the modified rate function as $p^*(\cdot)$ to distinguish it from the initial one.

Let i and j be two agents and suppose that, given a neutral net profit $s_{ij} = 0$, agent i will choose to initiate a trade with j one time out of ten (on average). Thus, we would like to have $p^*(0) = 0.1$. Looking at Figure 4.1 again (with $\beta > 0$), we find an obvious way of moving the y -intercept from 0.5 to 0.1 while keeping intact the *curvature* of the initial rate function $p(s_{ij})$. All we need to do is to implement a horizontal *parallel shift* to the right by $t > 0$ units:

$$p^*(s_{ij}) = p(s_{ij} - t) = \frac{e^{\beta s_{ij} - \beta t}}{1 + e^{\beta s_{ij} - \beta t}} = \frac{ke^{\beta s_{ij}}}{1 + ke^{\beta s_{ij}}}, \quad s_{ij} \in \mathbb{R}, \quad (4)$$

where $k = e^{-\beta t}$ ($0 < k < 1$) was defined just for convenience. We then solve the equation $0.1 = p^*(0)$ for k :

$$0.1 = \frac{k}{1 + k} \quad \Longleftrightarrow \quad k = 1/9 \quad \Longleftrightarrow \quad t = \frac{\ln 9}{\beta}.$$

Thus we need to shift the initial rate function by $\beta^{-1} \ln 9$ units. Figure 4.2 illustrates this shift.

Our next observation is that the probability mass function

$$P(\mathbf{X} = \mathbf{x}; \beta) = \exp(\beta S(\mathbf{x}) - (\ln 9)|E(\mathbf{x})|)K(\beta), \quad \mathbf{x} \in \Omega_{dir}, \quad (5)$$

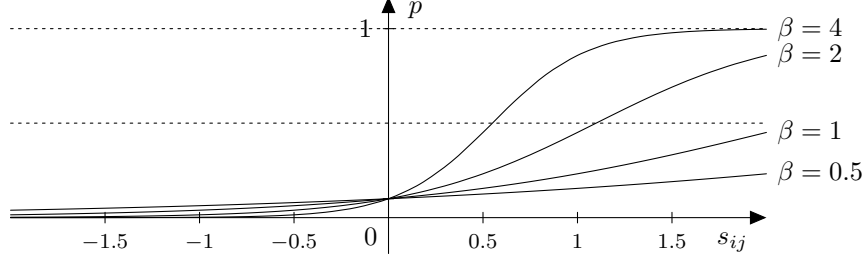


Figure 4.3: Rate functions with β set to different values.

yields exactly the inclusion rate function $p^*(\cdot)$ that we want. Indeed, applying the familiar algebraic techniques (Subsections 2.2.3 and 2.2.4), we get

$$\pi_{ij}(\mathbf{x}_{ij}^c) = \exp(\beta[s_{ij} - 0] + \ln(1/9)[1 - 0]) = (1/9)e^{\beta s_{ij}} =: \pi_{ij} \text{ (independent from } \mathbf{x}_{ij}^c),$$

from which

$$p^*(s_{ij}) = \frac{\pi_{ij}}{1 + \pi_{ij}} = \frac{(1/9)e^{\beta s_{ij}}}{1 + (1/9)e^{\beta s_{ij}}},$$

which is precisely the rate function in (4) with $1/9 = k = e^{-\beta t}$.

This example would not be complete if we didn't show how to choose the numeric value for the parameter β . Looking again at Figure 4.2, we notice that the point of indifference s^* (where $p(s^*) = 0.5$) has shifted from 0 to t . This means that, under model (5), the net profit of t dollars implies equal chances of agent i contacting or not contacting agent j . Now, by graphing the rate functions generated by various values of β (Figure 4.3), we see that the intercept $0.1 = p^*(0)$ remains fixed while the curvature (shape) of the rate function keeps changing, and so does the point of indifference s^* . If we set $\beta = 2$, we see that the point of indifference occurs a bit above \$1000. (We have tacitly assumed that the profits s_{ij} are measured in thousands of dollars). Thus, assuming that we have a good reason to believe that such an indifference point accurately reflects the attitudes in our hypothetical marketplace, we may choose to set $\beta = 2$. Ultimately, our probability model is given by the distribution

$$P(\mathbf{X} = \mathbf{x}) = \exp(2S(\mathbf{x}) - (\ln 9)|E(\mathbf{x})|)K, \quad \mathbf{x} \in \Omega_{dir},$$

where K , as usual, is the normalizing constant.

To reiterate, we began with the goal of reducing the neutral inclusion rate $p(0)$ to a reasonably small value, thus explicitly removing one of the two drawbacks (the second one) of the prototype model. We achieved this by adding the term $-(\ln 9)|E(\mathbf{x})|$ into the

exponent of the probability distribution in (1). Note that this new term acts *against* a large number of edges. Indeed, using the interpretation techniques of Subsection 2.2.2, we conclude that if graphs \mathbf{x} and \mathbf{y} have the same total realized potential ($S(\mathbf{x}) = S(\mathbf{y})$), then one of them with the highest density is assigned a smaller probability. Therefore, we have incidentally removed *both* drawbacks of the prototype model. The resulting model (5) is a special example of a Balanced Potential Model, formally defined below.

4.1.4 Formal Definition

Let \mathbf{X} be a random graph (either directed or undirected) and let s_{ij} be edge potentials (or antipotentials). Then \mathbf{X} is said to follow a *Balanced Potential Model* if its probability mass function is given by

$$P(\mathbf{X} = \mathbf{x}; \beta, \phi) = \exp(\beta S(\mathbf{x}) - \phi |E(\mathbf{x})|) K(\beta, \phi), \quad \mathbf{x} \in \Omega. \quad (6)$$

If s_{ij} represent potentials, we require that $\beta > 0$; if s_{ij} represent antipotentials, we require that $\beta < 0$. The value of ϕ is unconstrained, although it is assumed to be positive in most cases.

As a reminder, we note that the values s_{ij} are supplied externally as the edge attributes whose interpretation was discussed in the beginning of this section. Alternatively, the potentials s_{ij} may be constructed from the vertex attributes instead of edge attributes. As an example, think of the lawyer’s collaboration dataset that we described in the Introduction. One of the vertex attributes in that set was age (in years). Denote the age of lawyer i by s_i . It is conceivable that every lawyer is more inclined to collaborate with an older lawyers rather than with a younger one. Assuming that this insight is correct, we may define the edge potentials s_{ij} as the *joint* age of lawyer i and lawyer j :

$$s_{ij} = s_i + s_j.$$

In the lawyers’ dataset, the age is not the only attribute that may be used to define the edge potentials s_{ij} . For example, the number of years that a lawyer has been with the firm is another such attribute (assuming that every lawyer is indeed inclined to collaborate with lawyers who have been with the firm for many years rather than few). Thus the same network may contains several “candidates” for the definition of the edge potentials s_{ij} . We are thus motivated to introduce the “multipotential” generalization of a BPM.

Suppose that we are given r different families of potentials (or antipotentials). That is, for each $k = 1, 2, \dots, r$, we have a set $\{s_{kij}\}$ of potentials (or antipotentials) that represent the same edge attribute. In principle, all of them can be included in a BPM. The distribution of the corresponding model is then

$$P(\mathbf{X} = \mathbf{x}; \boldsymbol{\beta}, \phi) = \exp \left(\sum_{k=1}^r \beta_k S_k(\mathbf{x}) - \phi |E(\mathbf{x})| \right) K(\boldsymbol{\beta}, \phi), \quad \mathbf{x} \in \Omega, \quad (7)$$

where $S_k(\mathbf{x})$ ($k = 1, 2, \dots, r$) is the k 'th total realized potential of \mathbf{x} given by

$$S_k(\mathbf{x}) := \sum_i \sum_j x_{ij} s_{kij} = \sum_{e_{ij} \in E(\mathbf{X})} s_{kij}.$$

The log-odds of an edge indicator X_{ij} are easily shown to be

$$\ln \pi_{ij} = -\phi + \beta_1 s_{1ij} + \beta_2 s_{2ij} + \dots + \beta_r s_{rij},$$

and the normalizing constant is just

$$K(\boldsymbol{\beta}, \phi) = \prod_i \prod_j (1 + \pi_{ij})^{-1},$$

which we have already encountered in Subsection 2.2.4. (Recall that we have complete edge independence here.)

Basically the random graph \mathbf{X} is described as the familiar logistic regression model. Thus, in principle, we may use all the available statistical tools offered by the logistic regression. For example, we can formally test the hypotheses like $H_0 : \beta_k = 0$ and perform other procedures for model selection and goodness-of-fit evaluation.

Thus far, we have made abundant remarks concerning the interpretation of potentials and the parameters β_k . We now turn to the interpretation of ϕ . Informally, the model in (7) can be viewed as maximizing the realized potential while at the same time minimizing the density of the outcome. This “balancing act” is where the name “balanced potential” originates. It can be further clarified by the following observation. Suppose that graphs \mathbf{x} and \mathbf{y} are comprised of exactly the same edges with the exception of e_{ij} , which is included in \mathbf{x} but not in \mathbf{y} . (Symbolically, we have $x_{ij} = 1$, $y_{ij} = 0$, $\mathbf{x}_{ij}^c = \mathbf{y}_{ij}^c$.) Assuming that

$$\sum_{k=1}^r \beta_k (S(\mathbf{x}) - S(\mathbf{y})) = \sum_{k=1}^r \beta_k s_{kij} > 0,$$

the above difference represents the “gain” in total realized potentials (weighted by the β_k 's) resulting from the choice of outcome \mathbf{x} instead of outcome \mathbf{y} . Assuming that $\phi > 0$,

our previous “gain” has an associated “cost” equal to ϕ , resulting from the addition of the extra edge e_{ij} to \mathbf{y} . The parameter ϕ is thus called the *penalty factor*, while each β_k is called the *potential force* corresponding to the potentials (or antipotentials) s_{kij} .

4.1.5 Dual Specification

Here we return to the case of a single set of potentials (model (6)). The *dual specification* property reflects the simple fact that a BPM can be specified using two equivalent approaches. The first approach is to directly specify the potential force β together with the penalty factor ϕ . The second, equally interpretive approach, is to specify the *neutral rate* p_0 together with the *point of indifference* s_0 .

- The *neutral rate* p_0 ($0 < p_0 < 1$) equals the inclusion rate $p(0)$ obtained at a neutral potential $s_{ij} = 0$.
- The *point of indifference* s_0 is the value that the potential s_{ij} has to be in order to obtain a 50% inclusion rate $p(s_0)$.

After performing some simple derivations mirroring those in Subsection 4.1.3, we obtain the following relationships between β , ϕ , p_0 , and s_0 :

$$\begin{aligned}\phi &= \log(1 - p_0) - \log p_0, & \beta &= \frac{\log(1 - p_0) - \log p_0}{s_0}, \\ p_0 &= \frac{e^{-\phi}}{1 + e^{-\phi}}, & s_0 &= \frac{\phi}{\beta}.\end{aligned}$$

We see that specifying the pair (β, ϕ) is equivalent to specifying the pair (p_0, s_0) . Recall that in our example of the economic transaction network we set $\beta = 2$ and $\phi = \ln 9 \approx 2.197$. For that particular example, these parameter values do not tell us much, compared to the equivalent specification of the neutral inclusion rate $p_0 = 0.1$ and the point of indifference $s_0 \approx 1.099$ (or, \$1,099 if we impose \$000 units of measure).

4.2 Balanced Centrality Markov Chains

4.2.1 Definition

Let $\{\mathbf{X}(t) : t = 1, 2, \dots\}$ be a dynamic random graph on a fixed set of N vertices and let $C_i(t)$ denote the centrality of vertex i at time t , where $C_i(t)$ is an *arbitrary* centrality index. (For example, it could be any one of the centrality indices discussed in Chapter 2.) The dynamic graph is said to be a *Balanced Centrality Markov Chain* (BCMC) if it is a Markov chain with the transition probabilities given by

$$P(\mathbf{X}(t+1) = \mathbf{x} \mid \mathbf{X}(t) = \mathbf{x}(t); \beta, \phi) = \exp \left(\beta \sum_i \sum_j x_{ij} (C_i(t) + C_j(t)) - \phi |E(\mathbf{x})| \right) K(\beta, \phi). \quad (8)$$

Above we recognize simultaneously the exponential dynamic random graph model discussed in Subsection 2.5.4 and the Balanced Potential Model with the potentials given at time $t + 1$ by

$$s_{ij}(t + 1) := C_i(t) + C_j(t).$$

Thus the centralities obtained at step t become the exogenous covariates for the (conditional) exponential random graph at time $t + 1$. The index ranges for the summation $\sum_i \sum_j x_{ij} (C_i(t) + C_j(t))$ depend, as usual, on the state space Ω of the dynamic random graph (i.e., whether it is directed or undirected). The term $K(\beta, \phi)$ is the normalizing constant for which the expression was given in Subsection 4.1.4. Note also that the model does not specify a probability distribution for the initial random graph $\mathbf{X}(1)$. Any model can be suggested for $\mathbf{X}(1)$. For example, it might be convenient to assume that the initial state $\mathbf{X}(1) = \mathbf{x}(1)$ is known and fixed, i.e., non-random. However, we propose another approach. At time $t = 1$ we propose a Balanced Potential Model with the same parameters β and ϕ as above, but we assume zero potentials $s_{ij}(1)$ given for all edges, so that

$$P(\mathbf{X}(1) = \mathbf{x}; \phi) = \exp(-\phi |E(\mathbf{x})|) K(\phi), \quad \mathbf{x} \in \Omega, \quad (9)$$

from which we compute the inclusion rates

$$p_{ij} = \frac{e^{-\phi}}{1 + e^{-\phi}} = p_0 \text{ for all } i, j.$$

In other words, the initial random graph $\mathbf{X}(1)$ is simply the G_{N, p_0} Bernoulli graph that we saw in Subsection 2.3.1, where p_0 is the neutral inclusion rate corresponding to the choice of β and ϕ used in the BCMC transition probabilities (8).

Assuming that $\beta > 0$, we see that vertices whose *joint* centrality $C_i(t) + C_j(t)$ is high at time t are more likely to be adjacent at time $t + 1$ with those vertices whose joint centrality is low. Thus the model fits well to the situation in which we have a reason to believe that central vertices of a dynamic network tend to form ties with other central vertices. Note that eigenvector centrality is a particularly applicable index for such a model. Indeed, if we imagine that each of the vertices aims at maximizing its individual eigenvalue centrality as time passes, then such a goal can be achieved by making connections with vertices whose eigenvalue centrality is also high. (In the case of closeness or betweenness, it is not necessarily true that a vertex i can maximize its centrality by admitting an edge e_{ij} with the most central vertex j .) The idea that in some networks vertices represent agents whose goal is to maximize their centrality is not unusual. In their paper “Dynamics of networking agents competing for high centrality and low degree”, Holme & Ghoshal (2006) provide an interesting account of various optimal strategies that agents (vertices) use to maximize their centrality (closeness) while keeping their costs (degree) low. We see that this area of research is very similar in its interpretation to our investigation of Balanced Centrality Markov Chains. In comparison to the approach taken by Holme & Ghoshal, our model “penalizes” for the overall density of the graph rather than for individual vertex degrees.

4.2.2 Simulations

Here we discuss several Monte Carlo simulations of a BCMC. In all simulations presented in this section the length of the generated Markov chain will be $T = 40$ time periods. Thus by a single *simulation* we mean the sequence of $T = 40$ observations

$$\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(40),$$

where the transition probabilities $P(\mathbf{X}(t + 1) = \mathbf{x} \mid \mathbf{X}(t) = \mathbf{x}(t); \beta, \phi)$ are given by (8) for $t = 2, 3, \dots, 40$ and the initial random graph $\mathbf{X}(1)$ is generated according to the sampling distribution in (9). The graphs in our Markov chain are undirected. Note that a single simulation requires us to specify the following values:

1. the number of vertices N ,
2. the centrality index to be used,
3. the potential force β ,
4. the penalty factor ϕ .

The above choices indicate a vast variety of possible simulations that we could use to study the properties of BCMCs.

As a starting point, we want to run ten different simulations, in particular: 5 simulations with (capped) closeness centrality index and various choices for N , and then 5 simulations with eigenvector centrality index and various choices for N . The eigenvector centrality is computed according to the same rules that we employed in Subsection 3.4 where we discussed the application of centralities to the Lawyers' data. That is, we take the largest eigenvalue λ and we select the eigenvector of unit length as our centrality vector $\mathbf{c}(t)$. For this first set of simulations we chose the parameter values $\beta = 1.37$ and $\phi = 2.20$ corresponding to the neutral rate $p_0 = 0.1$ and the point of indifference $s_0 = 1.6$.

In each simulation we compute and plot the following statistics:

$$Y(t) := \sum_{i=1}^N C_i(t)/N = \text{average centrality at time } t,$$

$$M(t) := |E(\mathbf{X}(t))|/\binom{N}{2} = \text{edge density at time } t.$$

Note that at this point we are not producing any MCMC approximations, i.e., we only want to observe a single simulation for each of the 10 combinations of the centrality index and the number of vertices. The idea here is that a single simulation is enough to informally suggest the following phenomenon. As the number N of vertices is increased while all other choices are held fixed, both the average centrality $Y(t)$ and the edge density $M(t)$ exhibit less and less variability and become closer and closer to some fixed values Y^* and M^* . This behaviour is apparent from the plots in Figure 4.4, where closeness was used, as well as Figure 4.5, where eigenvector centrality was used. For example, in the simulation where closeness was used as the centrality index and the number of vertices was $N = 80$, we see that $Y^* \approx 0.60$ and $M^* \approx 0.39$. (See the rightmost plots in Figure 4.4.)

We can not be sure that this behaviour was not observed entirely by chance, unless we generate a large number, say, $n = 500$, of simulations with the same set of all relevant choices. If the data from all the $n = 500$ simulations exhibit the phenomenon that we just described, then we can become more confident that a BCMC with a given set of parameters exhibits some sort of “stabilizing” behaviour for the values of $Y(t)$ and $M(t)$. This is where we start talking about Monte Carlo approximations.

The goal now is to use the Monte Carlo method to estimate the values

$$\mu(Y(t)) = \mathbb{E}_{\beta,\phi}(Y(t)), \quad \sigma(Y(t)) = \sqrt{\text{Var}_{\beta,\phi}(Y(t))},$$

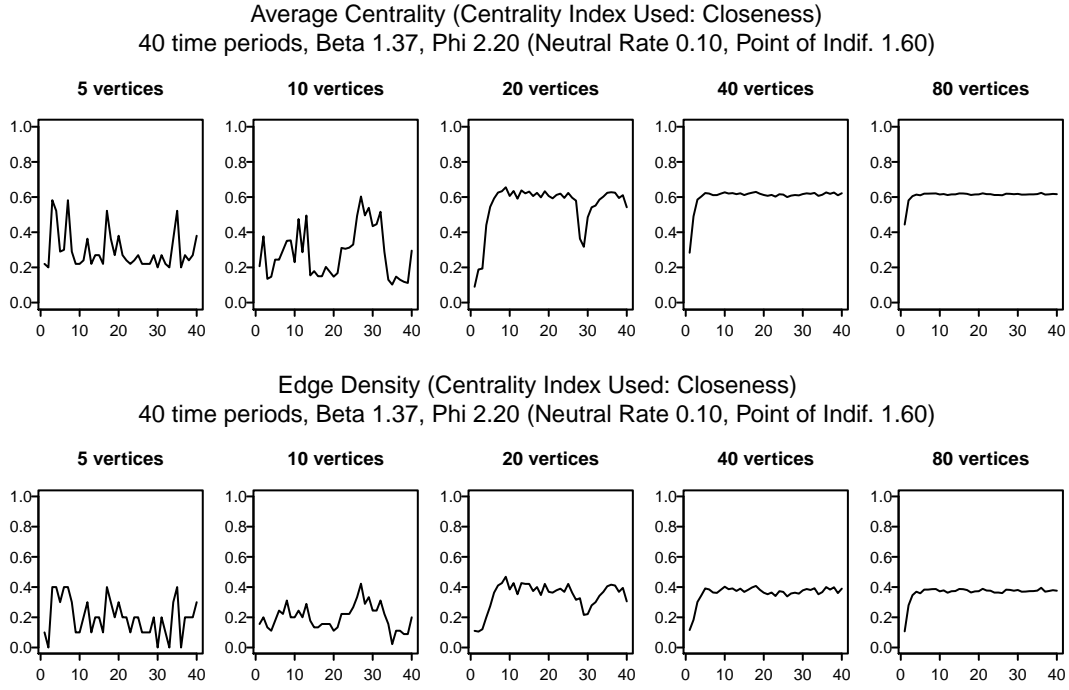


Figure 4.4: Five simulations of the BCMC with Closeness Centrality.

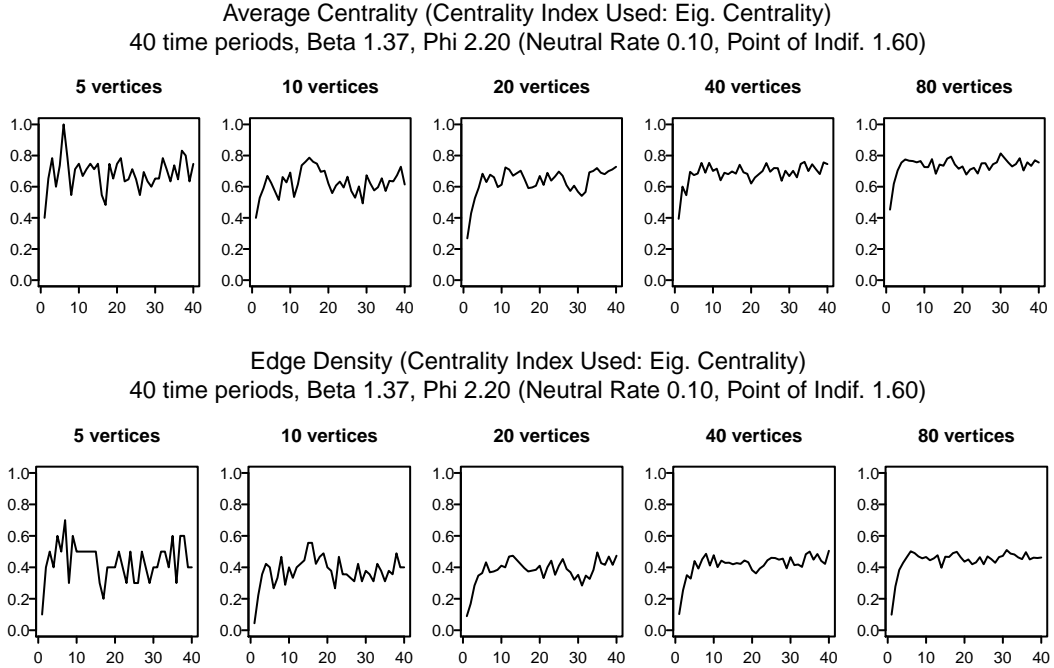


Figure 4.5: Five simulations of the BCMC with Eigenvector Centrality.

as well as

$$\mu(M(t)) = \mathbb{E}_{\beta, \phi}(M(t)), \quad \sigma(M(t)) = \sqrt{\text{Var}_{\beta, \phi}(M(t))}.$$

In other words, we are trying to answer the following question: what are the approximate expectations and standard deviations of the mean centrality $Y(t)$ and the edge density $M(t)$ at time t , given a certain centrality index, a fixed number N of vertices, and fixed parameter values β and ϕ ? To answer these questions, we produce $n = 500$ simulations with the same set of all relevant parameters (N , β , ϕ , centrality index). Thus, for example, we obtain $n = 500$ observations of the random variable $Y(t)$ for each t :

$$y^{(1)}(t), y^{(2)}(t), \dots, y^{(500)}(t).$$

The Monte Carlo approximation to $\mu(Y(t))$ is then given by

$$\hat{\mu}_{MC}(Y(t)) = (1/500) \sum_{k=1}^{500} y^{(k)}(t) \approx \mu(Y(t)),$$

and the approximation of $\sigma(Y(t))$ is given by

$$\hat{\sigma}_{MC}(Y(t)) = (1/\sqrt{500}) \sqrt{\sum_{k=1}^{500} [y^{(k)}(t) - \hat{\mu}_{MC}(Y(t))]^2} \approx \sigma(Y(t)).$$

Monte Carlo estimates of $\mu(M(t))$ and $\sigma(M(t))$ are obtained in an analogous way.

Before we present our first results, let us introduce two additional statistics:

$$Z(t) := (1/\sqrt{N}) \sqrt{\sum_{i=1}^N [C_i(t) - Y(t)]^2} = \text{standard deviation of vertex centralities at time } t$$

and

$$R(t) := \frac{\sum_i \sum_j x_{ij}(t)(C_i(t-1) + C_j(t-1))}{\sum_i \sum_j (C_i(t-1) + C_j(t-1))} = \text{realization rate at time } t \geq 2.$$

In the definition of the *realization rate* $R(t)$ above the denominator represents the maximum attainable potential at time t , which is obtained if and only if the graph $\mathbf{X}(t)$ is a complete graph. In the numerator we have the actual realized potential at time t , thus $R(t)$ represents the “success rate” by which the graph $X(t)$ has attained its maximum possible potential. The reason for introducing these additional statistics $Z(t)$ and $R(t)$ is that the Monte Carlo approximations of $\mu(Z(t))$ and $\mu(R(t))$ will exhibit the same stabilizing behaviour as the one shown by $\mu(Y(t))$ and $\mu(M(t))$.

We are now ready to present our simulation summaries in Figures 4.6–4.9 on pages 74–77. The following notes will help the reader to understand these four summaries.

- Figures 4.6 and 4.7 contain summary plots from the Monte Carlo simulations obtained with *closeness* centrality index. In the first simulation, we used $N = 10$ vertices. In the second simulation, we used $N = 40$ vertices.
- Figures 4.8 and 4.9 contain summary plots from the Monte Carlo simulations obtained with *eigenvector* centrality index. Again, we used $N = 10$ vertices for the first simulation and $N = 40$ vertices for the second simulation.
- Each summary contains four plots, one for each of the random variables $Y(t)$, $Z(t)$, $M(t)$, and $R(t)$. The plots are labelled accordingly and contain the Monte Carlo approximations $\hat{\mu}_{MC}(\cdot)$ for each value of t .
- The plot of the estimates $\hat{\mu}_{MC}(Y(t))$ of the expected average centralities and the plot of the estimates $\hat{\mu}_{MC}(M(t))$ of the expected densities contain additional dashed lines. These dashed lines represent the values

$$\hat{\mu}_{MC}(Y(t)) \pm \hat{\sigma}_{MC}(Y(t))$$

and

$$\hat{\mu}_{MC}(M(t)) \pm \hat{\sigma}_{MC}(M(t)).$$

By plotting these lines we achieve a better illustration of the sample distributions of $Y(t)$ and $M(t)$. For example, we can see that the variability in both $Y(t)$ and $M(t)$ is significantly reduced when the number N of vertices is increased from 10 vertices to 40 vertices. This behaviour is observed for both the closeness and the eigenvector centrality indices.

- The main property to observe in all of the four summaries is the following one. Starting from some time point in the beginning of our Markov chain, say $T_0 = 6$, all of the expectations

$$\mu(Y(t)), \mu(M(t)), \mu(Z(t)), \mu(R(t))$$

seem to remain fixed for $t \geq T_0$.

The last point in the above list describes an interesting phenomenon exhibited by the BCMCs. This “stabilizing” behaviour can be observed for different values of the parameters β and ϕ and not just for the specific values. To illustrate this point we include several additional simulation summaries in Appendix A, with parameters set to values different from the ones that we have already used.

Although we do not exactly know why the expectations of the statistics $Y(t)$, $Z(t)$, $M(t)$, and $R(t)$ remain fixed after some initial period, we are nevertheless able to conclude from our simulations that BCMCs are not appropriate in situations where the average centrality and the density of a dynamic network show high variability as time passes.

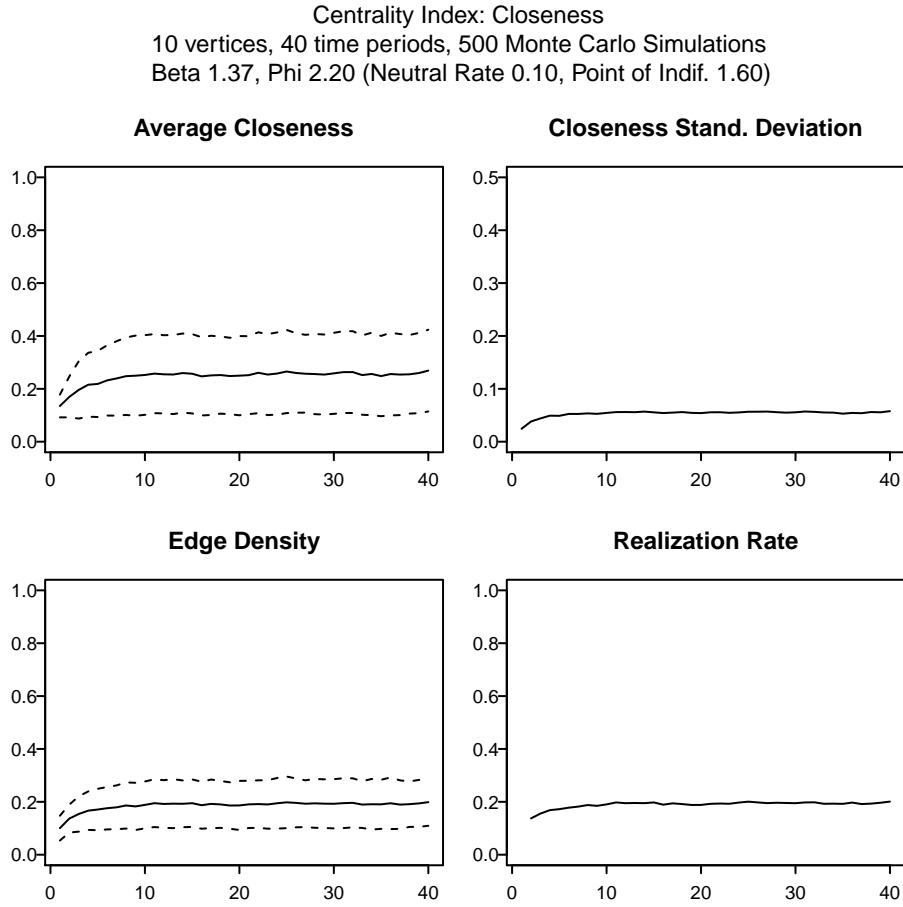


Figure 4.6: Summary of Monte Carlo simulations with Closeness centrality index and $N = 10$.

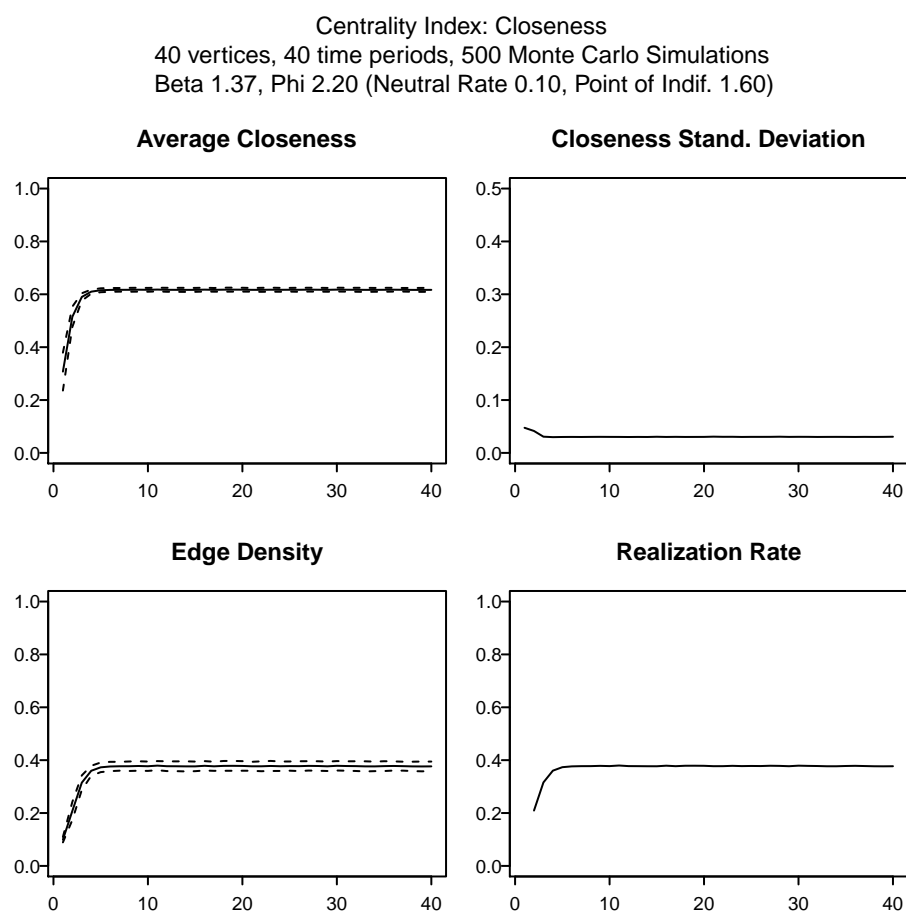


Figure 4.7: Summary of Monte Carlo simulations with Closeness centrality index and $N = 40$.

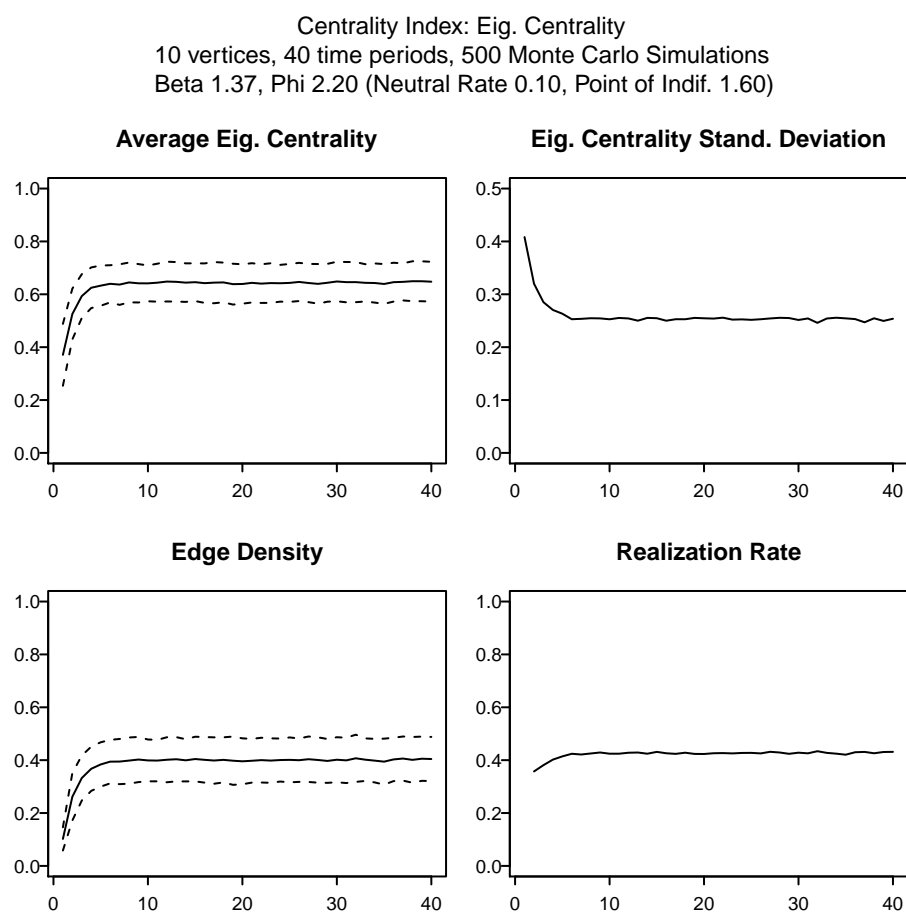


Figure 4.8: Summary of Monte Carlo simulations with Eigenvector centrality index and $N = 10$.

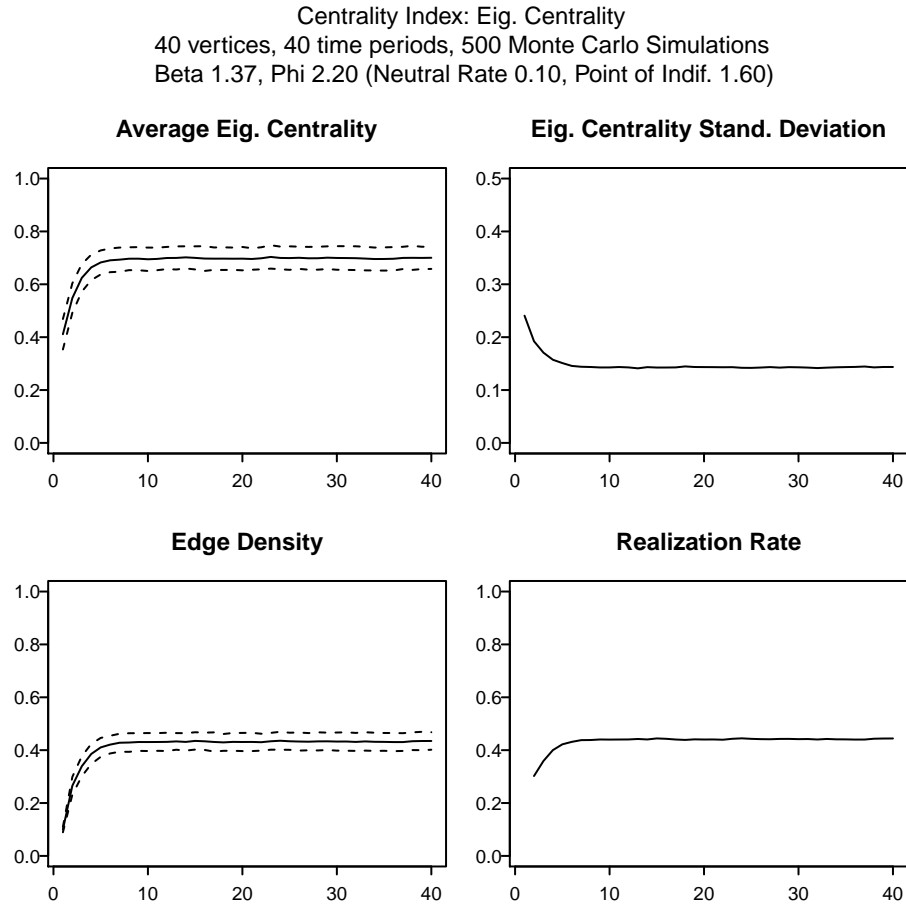


Figure 4.9: Summary of Monte Carlo simulations with Eigenvector centrality index and $N = 40$.

Chapter 5

Directions of Future Research

The Balanced Potential Model as well as the Balanced Centrality Markov Chains require additional investigation of their properties. The following list contains some suggestions for future research.

- For BPMs as well as BCMCs estimation of unknown parameters β and ϕ is performed in the same way as for any logistic regression model. However, model fitting as well as model selection methods have to be tested with various real datasets in order to gain a more thorough understanding of BPMs and BCMCs.
- It is important to investigate further the phenomena described in Subsection 4.2.2. In particular, we need to find a theoretical basis for the observed “stabilizing” behaviour of the simulated BCMCs.
- Suppose that s_{ij} represents antipotentials in a BPM. Then $\beta < 0$. But then the inverted values $s_{ij}^* := \frac{1}{s_{ij}}$ represent potentials, and we can alternatively use the BPM with s_{ij}^* and $\beta > 0$. How do we choose between the two approaches?

For example, suppose that we have N vertices that represents points in a Euclidean space. We know that two points i and j are more likely to form an edge e_{ij} if the Euclidean distance d_{ij} between them is small. In this problem the distances d_{ij} represent antipotentials. However, we can choose between two alternative models

$$\log \pi_{ij}(\beta_1, \phi) = \beta_1 d_{ij} + \phi, \quad \beta_1 < 0,$$

and

$$\log \pi_{ij}(\beta_2, \phi) = \beta_2 / d_{ij} - \phi, \quad \beta_2 > 0,$$

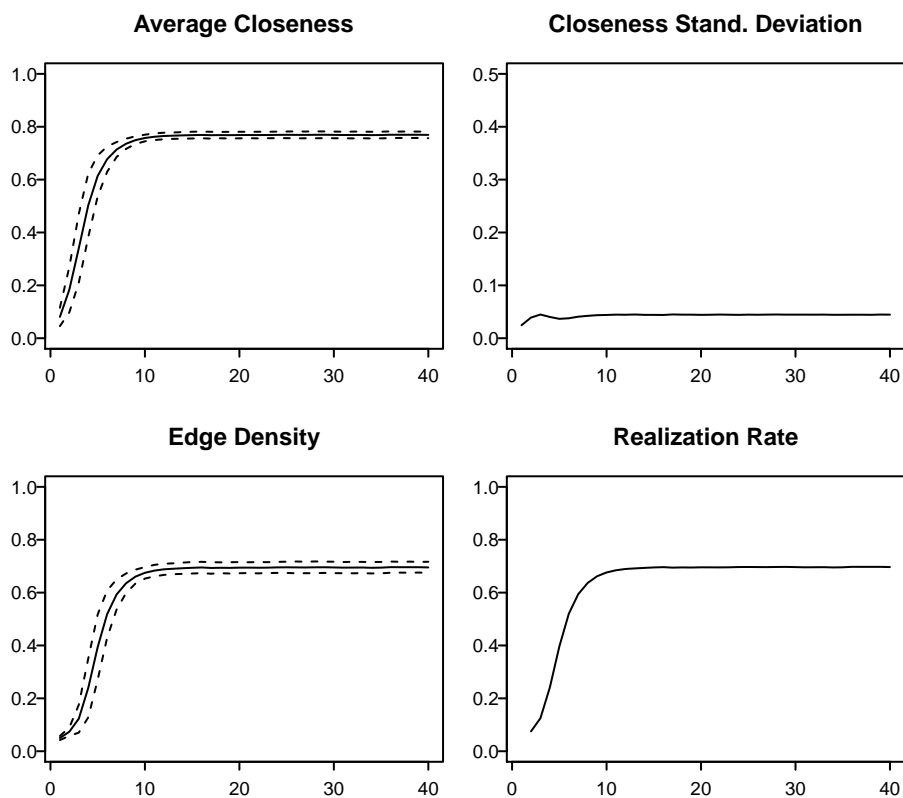
both of which have the same interpretation (points that are far apart are less likely to connect). However, it is not clear which of the two models would be a better fit to a hypothetical dataset.

APPENDICES

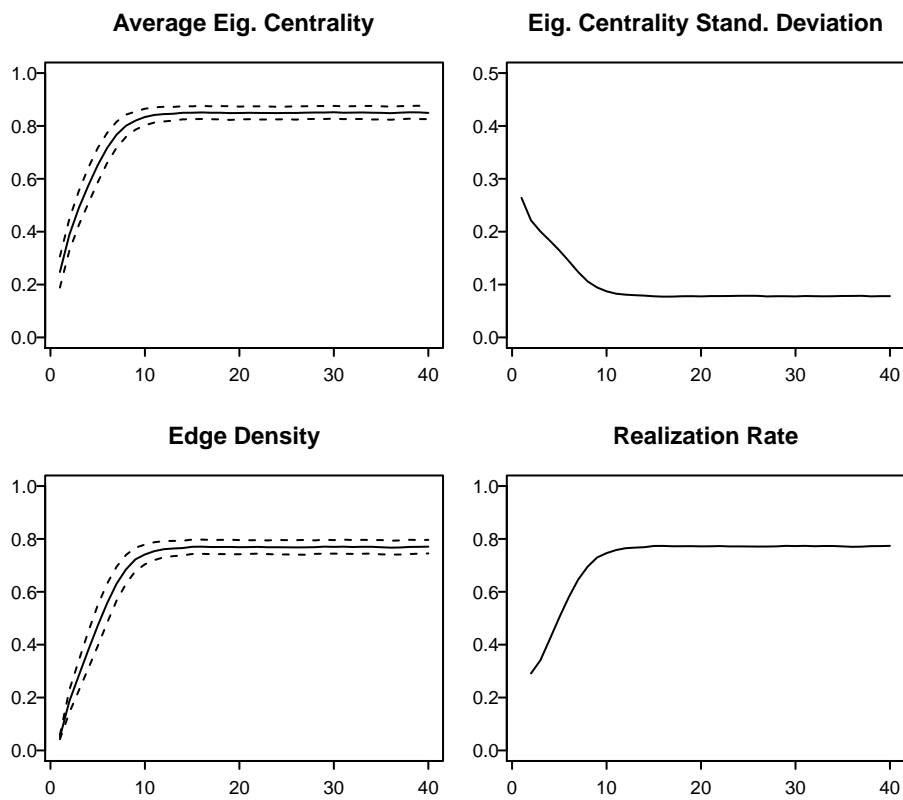
Appendix A

BCMC Simulations

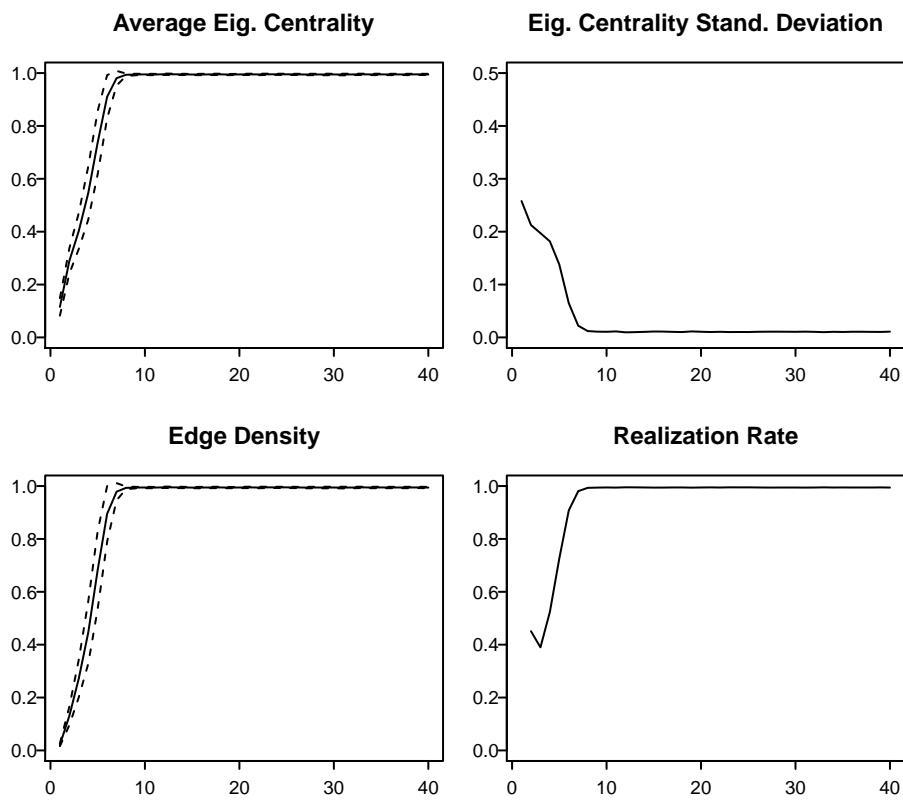
Centrality Index: Closeness
40 vertices, 40 time periods, 500 Monte Carlo Simulations
Beta 2.45, Phi 2.94 (Neutral Rate 0.05, Point of Indif. 1.20)



Centrality Index: Eig. Centrality
40 vertices, 40 time periods, 500 Monte Carlo Simulations
Beta 2.45, Phi 2.94 (Neutral Rate 0.05, Point of Indif. 1.20)

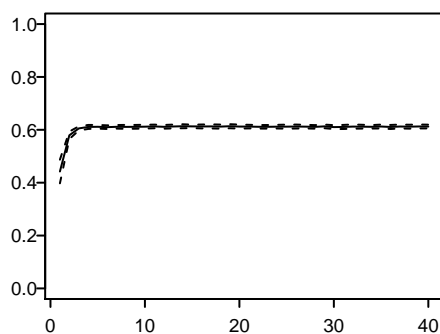


Centrality Index: Eig. Centrality
 40 vertices, 40 time periods, 100 Monte Carlo Simulations
 Beta 4.58, Phi 3.89 (Neutral Rate 0.02, Point of Indif. 0.85)

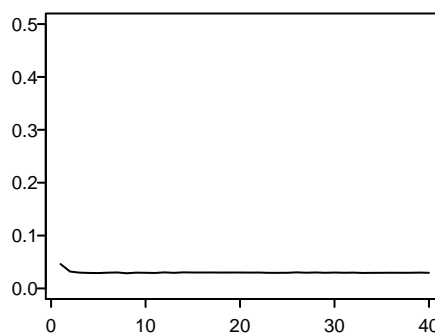


Centrality Index: Closeness
40 vertices, 40 time periods, 100 Monte Carlo Simulations
Beta 0.96, Phi 1.73 (Neutral Rate 0.15, Point of Indif. 1.80)

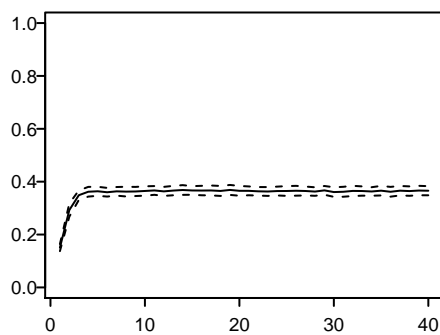
Average Closeness



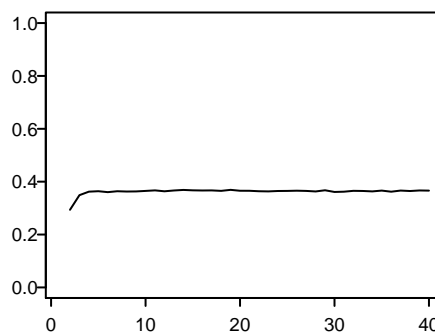
Closeness Stand. Deviation



Edge Density



Realization Rate



References

- Anderson, C. J., Wasserman, S., & Crouch, B. (1999). A p* primer: logit models for social networks. *Social Networks*, 15, 29
- Arnold, B. C., & Strauss, D. (1991). Pseudolikelihood estimation: some examples. *Sankhya*, 53(2), 233–243. 29, 31
- Barabasi, A., & Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286, 509–512. 37
- Bartz, K. C. (2011). *Efficient Monte Carlo Methods for Sampling and Inference: Networks, Brains, Proteins*. Ph.D. thesis, Department of Statistics, Harvard University. 29
- Bartz, K. C., Blitzstein, J. K., & Liu, J. S. (2010). Monte carlo maximum likelihood for exponential random graph models: From snowballs to umbrella densities. *Social networks*. Paper submitted. Currently under review. 29
- Besag, J. E. (1975). Statistical analysis of non-lattice data. *The Statistician*, 24, 179–195. 29
- Bollobas, B. (1985). *Random Graphs*. London: Academic Press. 20
- Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2(1), 113–120. 45
- Brandes, U., & Erlebach, T. (2009). *Statistical Analysis of Network Data. Methods and Models..* Lecture Notes in Computer Science. Springer. 42, 43, 45
- Cessie, S. L., & van Houwelingen, J. (1994). Logistic regression for correlated binary data. *Journal of the Royal Statistical Society*, 43, 95–108. 30
- Cohen, W. W. (2009). Enron email dataset. <http://www.cs.cmu.edu/~enron/>. 38

- Cox, D. R. (1970). *The Analysis of Binary Data*. Methuen's Monographs on Applied Probability and Statistics. Methuen and Co. 4
- Cramér, H. (1946). *Mathematical Methods of Statistics*. Princeton University Press. 24
- Frank, O., & Strauss, D. (1986). Markov graphs. *Journal of the American Statistical Association*, 81(395), 832–842. 21, 22, 29, 30
- Freeman, L. (1979). Centrality in social networks: I. conceptual clarification. *Social Networks*, 1, 215–239. 51
- Geyer, C. J., & Thompson, E. A. (1992). Constrained monte carlo maximum likelihood for dependend data. *Journal of the Royal Statistical Society*, 54, 657–699. 25
- Geys, H., Molenberghs, G., & Ryan, L. M. (1997). Communications in statistics—theory and methods. *Social Networks*, 26, 2743–2767. 32
- Gilks, Richardson, & Spiegelhalter (1996). *Markov Chain Monte Carlo in Practice*. Chapman and Hall. 24
- Hakimi, S. L. (1965). Optimal locations of switching centers and the absolute centers and medians of a graph. *Operations Research*, 12, 450–459. 51, 55
- Handcock, M. S. (2003). Statistical models for social networks: Inference and degeneracy. In *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers*, (pp. 229–240). Board on Behavioral, Cognitive, and Sensory Sciences. 29
- Handcock, M. S., Robins, G., Snijders, T., Moody, J., & Besag, J. (2003). Assessing degeneracy in statistical models of social networks. *Journal of the American Statistical Association*, 76, 33–50. 29
- Hanneke, S., Fu, W., & Xing, E. P. (2010). Electronic journal of statistics. *Internet Mathematics*, 4, 585–605. 39
- Holland, P. W., & Leinhardt, S. (1977a). A dynamic model for social networks. *Journal of Mathematical Sociology*, 5, 5–20. 36
- Holland, P. W., & Leinhardt, S. (1977b). Social structure as a network process. *Z. Soziol.*, 6, 386–402. 36

- Holland, P. W., & Leinhardt, S. (1981). An exponential family of probability distributions for directed graphs. *Journal of the American Statistical Association*, 76(737). 20, 21
- Holme, P., & Ghoshal, G. (2006). Dynamics of networking agents competing for high centrality and low degree. *Physics Review Letters*. 69
- Huisman, M., & Snijders, T. A. B. (2003). Statistical analysis of longitudinal network data with changing composition. *Sociological Methods and Research*, 32, 253–287. 35, 36
- Hunter, D. (2007). Curved exponential family models for social networks. *Social Networks*, 29, 216–230. 15
- Hunter, D. R., & Handcock, M. S. (2006). Inference in curved exponential family models for networks. *Journal of Computational and Graphical Statistics*, 15, 565–583. 26
- Katz, L. (1953). A new status index derived from sociometric analysis. *Psychometrika*, 18(1), 39–43. 45
- Kolaczyk, E. D. (2009). *Statistical Analysis of Network Data*. Series in Statistics. Springer. 11, 58
- Lai, T. L. (2003). Stochastic approximation. *The Annals of Statistics*, 31(2), 391–406. 28
- Lazega, E. (2001). *The collegial phenomenon: the social mechanisms of cooperation among peers in a corporate law partnership*. Oxford University Press. 2
- Le Cam, L. (1960). Locally asymptotically normal families of distributions. *University of California Publications in Statistics*, 3, 37–98. 24
- Lehmann, E. L. (1983). *Theory of Point Estimation*.. Wiley. 26, 32
- Lubbers, M. J., & Snijders, T. (2007). A comparison of various approaches to the exponential random graph model: A reanalysis of 102 student networks in school classes. *Social Networks*. 30
- Minieka, E. (1977). The centers and medians of a graph. *Operations Research*, 25, 641–650. 55
- Mitzenmacher, M. (2004). A brief history of generative models for power law and lognormal distributions. *Internet Mathematics*, 1(2), 226–251. 37

- Nevel'son, M. B., & Hasminski, R. A. (1973). An adaptive robbins-monro procedure. *Automatic and Remote Control*, *34*, 1594–1607. 26
- Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, *45*, 167–256. 1, 37
- Robbins, H., & Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, *22*, 400–407. 26
- Robins, G., & Pattison, P. (2001). Random graph models for temporal processes in social networks. *Journal of Mathematical Sociology*, *25*, 5–41. 39
- Robins, G., Snijders, T., Wang, P., & Handcock, M. (2007). Recent developments in exponential random graph (p^*) models for social networks. *Social Networks*, *29*, 192–215. 30
- Sabidussi, L. (1966). The centrality index of a graph. *Psychometrika*, *31*, 581–683. 51
- Schweinberger, M. (2011). Instability, sensitivity, and degeneracy of discrete exponential families. *Journal of the American Statistical Association*, *In Press*. 29
- Small, C. (1997). Multidimensional medians arising from geodesics on graphs. *The Annals of Statistics*, *25*(2), 478–494. 55
- Snijders, T. A. B. (1996). Stochastic actor-oriented models for network change. *Journal of Mathematical Sociology*, *21*, 149–172. 27, 35
- Snijders, T. A. B. (2002). Markov chain monte carlo estimation of exponential random graph models. *Journal of Social Structure*, *3*. 26, 27, 28, 30
- Snijders, T. A. B. (2005). Models for longitudinal network data. *Models and methods in social network analysis*. 35
- Snijders, T. A. B., Pattison, P. E., Robins, G. L., & Handcock, M. S. (2006). New specifications for exponential random graph models. 22, 28, 58
- Strauss, D., & Ikeda, M. (1990). Pseudolikelihood estimation for social networks. *Journal of the American Statistical Association*, (pp. 204–212). 29
- van Duijn, M., Gile, K., & Handcock, M. S. (2009). Comparison of maximum pseudo likelihood and maximum likelihood estimation of exponential family random graph models. *Social Networks*, *31*, 52–62. 24, 30

- Venter, J. H. (1967). An extension of the robbins-monro procedure. *Annals of Mathematical Statistics*, 38, 181–190. 26
- Wasserman, S. (1978). Models for binary directed graphs and their applications. *Advances in Applied Probability*, 10, 803–818. 36
- Wasserman, S., & Faust, K. (1994). *Social Network Analysis. Methods and Applications..* Structural Analysis in the Social Sciences. Cambridge University Press. 45
- Wasserman, S., & Robins, G. (2005). An introduction to random graphs, dependence graphs, and p^* . *Models and methods in social network analysis*. 23, 28, 30, 32
- White, S., & Smyth, P. (2003). Algorithms for estimating relative importance in networks. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, (pp. 266–275). ACM. 55