UNIVERSITÀ DEGLI STUDI DI NAPOLI
"FEDERICO II"
DIPARTIMENTO DI MATEMATICA E STATISTICA

DOTTORATO IN STATISTICA
XXIII CICLO

ASSESSMENT OF STOCHASTIC APPROXIMATION METHODS AND OF DEGENERACY DIAGNOSTIC TOOLS IN EXPONENTIAL RANDOM GRAPH MODELS

Agnieszka Stawinoga

ANNO ACCADEMICO 2010 - 2011

Ai miei genitori
Moim Rodzicom

Contents

List of Figures

List of Tables

Introduction

In recent decades there has been an enormous growth of interest in the notion of social network and the methods of Social Network Analysis (SNA). Traditionally social network theory has been based on the social relations in order to understand the formation of social structures by the collection of simple rules operating on local relations. Social network perspectives provide a natural approach for the study of social, political and economic interaction structures and give formal definitions for answering standard social and behavioral science research questions.

The methodology developed in the field of network analysis has been categorized into descriptive methods and statistical methods. Descriptive techniques for network analysis drew inspiration from mathematical graph theory, using tools from linear algebra and focusing on issues of clustering and connectivity. This approach has become the heart of the field, providing a wide range of summary measures (paths, cycles, density, centrality, structural equivalence) to represent both the network position occupied by specific nodes, and the overall network structure. These techniques serve valuable purposes in describing and understanding network features that might bear on particular research questions. Because of the lack of adequate computing and statistical tools the statistical methods for estimation from samples, and quantifying the uncertainty in the estimates have been addressed in a limited way in this descriptive context.

Since applying descriptive or deterministic approaches do not require distributional assumptions about particular structural properties, if we decide to understand the uncertainty associated with observed outcomes, it is necessary to associate model-based

statistical methods which complement analyses based on descriptive techniques. In contrast to a descriptive approach, statistical techniques make the stochastic assumptions about actor behaviour and assume the existence of some probabilistic mechanism that describes the observed network. The statistical methods may be organized into two parts; the first group consists of dyadic and triadic methods which represent statistical models for subgraphs and the second group of statistical models for entire graphs and digraphs.

There are two important aspects of the statistical modeling of social networks:

- (1) the first one focuses on the process that gives rise to the formation of network ties among nodes in the population, and investigates the impact of local interactive processes on the global network structure,
- (2) the second one deals with the question how to infer the underlying process from the observed patterns and it requires statistical methods for sampling, estimation and inference.

Robins et al. [84] presented some important reasons for applying statistical models to analyze social networks. Stochastic models give the possibility to understand the complexity of social behavior and to investigate questions on how the global features observed in a network may be explained by a low number of local rules. The purpose of statistical models is the quantitative examination of the stochastic properties of social relations and the actors of a particular network. The assumption we made for the use of statistical models for networks is that the network is a self-organizing system of relational ties which means that dyadic relations are generated by the local social processes which depend on the existing relations. Since in network analysis, independence is not something one would want to assume and understanding the dependence among observations is actually the primary task, we need an appropriate model to enable us to describe the process and the dependencies in it. A well-specified model captures the regularities in the processes giving rise to network ties and at the same time recognizes the variability that we are unlikely to be able to model in detail. Since the deterministic approaches for analyzing single binary networks are not often appropriate for more complicated data, a properly formulated model is useful to obtain an efficient

representation of the complex network data. It gives the possibility to understand the significance of various kinds of local processes and then to reproduce the observed network at the global level.

Statistical models also allow inferences about whether certain network substructures are more commonly observed in the network than might be expected. Then, it is possible to develop hypotheses about the social processes that might produce these structural properties. Statistical modeling makes it possible to evaluate the differences in similar qualitative predictions and to understand by which different social processes they are made. Their utility is also recognized for examining more complex aspects such as longitudinal network [94] or multiple network structures [76].

As mentioned above (in point 2) it is very important to have an adequate method of estimation that works with dependent observations. Fortunately, the current theoretical developments, advances in statistical computing, and innovations in data collection give the possibility to make progress in model-based statistical methods for network analysis. For instance, the statistical methods that rely on resampling (i.e. the bootstrap, jackknife, and permutation tests) have been readily adopted by network analysits for tests like quadratic assignment procedures for sociomatrix regression. Development of Markov Chain Monte Carlo algorithms provided a very useful tool for solving complex estimation problems. and gave the possibility for advancement in the model-based framework for social network analysis.

In this work we pay attention to the Exponential Random Graph Models (ERGMs), the statistical models which provide a general framework for modeling dependent data where the dependence can be thought of as a neighborhood effect. The models can be used to decompose overall network structural properties into the effects of localized interaction rules: the traditional concern of the field of social network analysis. Here we concentrate on the modeling of a single binary social network but this class of models finds applications also in various more complex forms of relational data. Until recently, inference for ERGMs has been almost exclusively based on an alternative local approximation to the likelihood function referred to as the pseudo-likelihood [102] due to the dependence of the normalizing constant on the unknown parameter. The computational

tractability of the pseudo-likelihood function makes it an attractive alternative to the full likelihood function. Since this approach assumes conditional independence of the random variables representing the relational ties, it gives reasonable results only for dyadic independence models. In this case the Maximum Pseudo-Likelihood (MPLE) estimators correspond to the exact solution and the true maximum likelihood estimator may be found via an MPLE computation. For dyadic dependence models statistical properties for MPLE estimators are not well understood and in practice MPLE does not provide a good performance [104]. Currently the favored methods for statistical inference are stochastic procedures based on Markov Chain Monte Carlo (MCMC) such as the Netwon-Raphosn algorithm [27], an MCMC implementation of the Robbins-Monro algorithm [94], [95], an alternative steplength algorithm [53] was introduced recently.

The development of Monte Carlo estimation methods for ERGMs has allowed us to understand better model behavior, especially model degeneracy and stability ([33], [34]), two properties of random graph models that have an important consequence on the use of these models. Many of the model proposed in literature suffer from degeneracy but the problem of degeneracy and poor fitting can be resolved by using recently developed statistics which are able to capture high-order dependency structure in networks ([99], [56], [87]).

The present manuscript is based on two main motivations. Firstly, we are interested to examine model diagnostics and check for degeneracy of ERGMs using different methods and functions proposed in the literature and available in statnet for R, which is an integrated set of software tools for the representation, visualization, simulation and analysis of network data. Taking into account the stochastic nature of the estimation algorithm Newton-Raphson, we would like to evaluate whether the results of degeneracy diagnostics tools are also different for runs with different seed. Secondly, we aim to evaluate and compare results obtained for networks of various sizes from three different estimation procedures such as Newton-Raphson, Robbins-Monro and Stepping.

The present work is divided into four chapters. In particular in *Chapter* 1, we will start with the introduction of the fundamentals of how networks are represented, measured, and characterized. Then we will present the history of statistical models for social network and provide concepts and definitions that are the basis for the statistical modeling language of network research.

Chapter 2 is devoted to define the Exponential Random Graph Model. It begins with the concept of dependence graphs to underline their importance in defining different classes of ERGMs, to be followed by the explanation of how the ERGMs can be viewed as an autologistic regression model. It will conclude with a presentation of different classes of ERGMs (Bernoulli model, p1 model, Markov model and more recent models) and their importance in describing social phenomena.

Indeed, in *Chapter* 3, we focus the attention on the various concepts with regard to the estimation and interpretation of model parameters. This chapter contains the fundamental issues such as different estimation methods, simulation networks using MCMC algorithms and problem of degeneracy. These concepts are important to the issues addressed and discussed in the next chapter.

Finally, the goal of *Chapter* 4 is to illustrate the contributions of the research. In particular, the first contribution is based on the definition of the strategy for evaluation of the degeneracy diagnostic tools in the class of models ERGM. Considering the stochastic nature of the Newton-Raphson algorithm we are interested in examining whether for different runs of the same model the results of the diagnostic tools for degeneracy are different and whether this depends on the size of the network.

The second contribution is based on the comparison of the accuracy and precision of three approximation techniques, such as the Newton-Raphson, the Robbins-Monro, and the Stepping procedure used in the estimation of parameters for networks of different sizes.

CHAPTER 1

Elements of Theory of Social Network Analysis

The purpose of this chapter is to present definitions and notations necessary for applying statistical models for relational data discussed in the next chapters. Firstly, we present a brief history of evolution of statistical modeling for social network. After we will introduce some definitions and concepts fundamental for modeling approach. Then we will conclude with a general framework for exponential random graph model construction.

1.1 A brief history of statistical models for SNA

Statistical methods for social networks have been developed over a period of seventy years. Beginning in the late 1930's, the first generation of research dealt with the distributions of various network statistics, under a variety of null models. The work of Moreno and Jennings [68] presents one of the first statistical analysis of social network. They obtained the first simulation of a random digraph distribution by randomly assigning choices to individual actors. An important study about random directed graph distribution was obtained by Katz and Powell [62] who also introduced an index for mutuality [61].

The second generation, beginning in the 1970's and continuing into the 1980's

concerned models usually for probabilities of relational ties among very small subsets of actors, in which various simple substantive tendencies were parameterized. During that period Holland and Leinhardt presented empirical verification of probabilistic versions of deterministic network structural theories, which use graph theory to make predictions about network theory [43] [44]. In the mid 1970's the interest grew in the triadic analyses designed to study structural balance and transitivity theory. The research of Holland, Leinhardt and Davis was the first social network methodology to use sophisticated statistical models. These authors provided strong statistical evidence that transitivity is a very important structural tendency in social network [45]. Davis [15] showed that a basic feature of many social networks is the tendency toward transitivity ("friends of my friends are my friends"). Holland and Leinhardt in [46], [50], [51] and Johnsen in [59] [60]) showed that transitivity is one of many "null hypotheses" that can be tested by examining triads and the triples that they contain.
Holland and Leinhardt [47], [48] proposed a class of stochastic process models that could serve as the basis for extending a family of parametric probability models to time series data, later developed by Wasserman [106], [107] and Galaskiewicz and Wasserman [25]. Fienberg and Wasserman [19], [20] considered probability models for covariates and multiple relational data. Warner et al. [105] and Kenny and Nasby [63] used analysis of variance models to study networks with valued relations.

The third generation began with Exponential Random Graph Models (ERGMs), commonly called the p* class of models, introduced by Frank and Strauss [23] who viewed it as a generalization of model p1 introduced by Holland and Leinhardt [52] and all its relatives; but first of all, they gave importance to the Markov model. Wasserman and Pattison [109] and Pattison and Wasserman [76] further extended this class of models, describing them as p* models. They showed how a Markov parametric assumption provides just one of many possible sets of parameters. Most initial investigations focused on undirected and directed single, dichotomous relations. Generalizations of p* models to valued relations, and to more than one relation were proposed by Frank and Strauss [23], Robins et al. [86], Wasserman and Pattison [76] and Koehly and Pattison [64]. For this class of models Straus and Ikeda [102] developed an estimation

technique based on a local approximation to the likelihood function referred to as the pseudo-likelihood whose computational tractability makes it an attractive alternative to the full likelihood function. Recently van Duijn et al. [104] demonstrated that for some kinds of models statistical properties for MPLE estimators are not well understood and in practice MPLE does not provide a good performance. Currently the favored methods for statistical inference are Markov Chain Monte Carlo (MCMC) Maximum Likelihood Estimate (MLE) [27] and an MCMC implementation of the stochastic approximation Robbins-Monro algorithm [94] [95], both of which rely on the properties of the method of moments for exponential family distributions. There has been proposed by Hummel et al. [53] an alternative method, a partial stepping algorithm which give the possibility to find an approximate MLE in many cases where this was previously not possible. There has been proposed also an Bayesian approach for inference in ERGMs [9] which is an alternative to the classical inferential methods and it allows uncertainty about model parameters given the data to be explored through a posterior distribution. The introduction of recently developed statistics [99] and the implementation of curved exponential random graph models by Hunter and Handcock [56], [56] provide a very effective solution at overcoming the problems of degeneracy. Hoff et al. [42], assuming a unknown position for each node of the graph in a latent space and the probability of the edges as functions of those positions and node covariates, proposed the latent variable models. The further extension of these new modelling alternatives is the latent position cluster model of Handcock et al. [38] takes account of clustering. Nowicki and Snijders [73] proposed stochastic blockmodel methods involving block model structures whereby nodes of the graph are partitioned into latent classes and their relationship depends on block membership. More recently, Airoldi et al. [1] has developed the mixed membership approach extending the assumption of a single block membership.

A very important part of statistical modeling for social networks are statistical methods for longitudinal network data. Statistical studies of network evolution are especially important for obtaining insight into social networks but only just beginning to be developed. A natural starting point for modeling longitudinal network data are

continuous-time Markov chains proposed by Holland and Leinhardt [48]. They defined on the space of all digraphs with a given node set, for modeling social network dynamics even if the observations are made at a few discrete time points and not continuously. The dyad-independent models were elaborated by Wasserman [106] [107] and Leenders [67]. Their assupmtions that the ties between pairs of actors (dyads) develop according to processes that are mutually independent between dyads is not realistic for social processes, because dependence between the set of ties among three or more actors can be very strong. Models which do allow such higher-order dependencies and represent a variety of network effects are so-called actor-oriented models proposed by Snijders and van Duijn [96] and Snijders [94]. This class of models is based on a model of actors changing their outgoing ties as a consequence of stochastic optimization of an objective function. The actor-oriented models are too complicated for the calculation of likelihoods or estimators in closed form. The fact thet these models represent stochastic processes which can be easily simulated was exploited by the estimation method based on a Method of Moments (MoM) estimator implemented algorithmically by stochastic approximation [80]. Recently it has been proposed an MCMC algorithm for approximating the ML estimator for Social Network Dynamics [65] [98].

1.2 Representing Networks

Networks are a form of "relational data", data whose properties cannot be reduced to the attributes of the individuals (nodes) involved. The objects of analysis are the relations. In typical applications, the nodes in a network represent individuals, and the links (ties or edges) represent a specific relationship among individuals. Nodes can represent larger social units (groups, families, organizations), objects (physical resources, servers), abstract entities (concepts, texts, random variables). Relational data finds applications in social networks (friendship networks, co-authorship networks of scientists), the internet (the world-wide web) [8], transportation networks and biological networks [58].

A social network consists of a set of n social actors which interact with each other in

various ways. Social network analysis focuses on the pattern of r relationships between the actors and is widely used in social sciences. A social network can also contain a collection of attribute characteristics, measured on the actors. In the context of social network analysis, Holland and Leinhardt [52] classified relational data as follows:

- Single relationship data: A single relationship observed on a set of nodes at a single point in time.
- Time series data: There may be more than one point in time at which the relation on the set of nodes is observed.
- Covariates: There may be information about nodal attributes in addition to the relationship information.
- Valued relationship: Some types of relationship exist in warying degrees or strengths rather than in an all-or-none fashion.
- Multiple relationships: There may be more than one type of relationship studied on the same set of nodes

To understand the measurement of relations it is necessary to consider two important properties: whether the relation is directional or undirectional, and whether it is dichotomous or valued. In the directional relation, the relational tie between a pair of actors has an origin and a destination (i.e. one country imports alimentary goods from another country). In an undirectional relation the tie between a pair of actors does not have a direction (i.e. there is a tie between two countries if they share a border). Dichotomous relations are coded as either present or absent for each pair of actors (dichotomous: 1 or 0). Valued relations can take a range of values which indicate strength, frequency or intensity of the tie between each pair of actors (i.e the frequency of interaction among pairs of people).

In the following analysis we will use graphs and adjacency matrices to represent social network data mathematically.

1.2.1 Graphs

We let $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ denote a *graph*, which is described by the two sets $\mathcal{N}$ and $\mathcal{L}$. The set $\mathcal{N} = 1, 2, \ldots, n$ is a collection of nodes of a graphs, $\mathcal{N}$ in the network theory

defines a set of actors.

We consider an ordered pair of actors, i and j. If the relation under examination is a dichotomous we are interested if actor i relates to actor j or does not and in a directional one, the pair of actors i and j is distinct from the pair j and i. If a tie exists the ordered pair is an element of $\mathcal{L}$ which is a set of all relational ties between pairs of actors. The number of elements of $\mathcal{L}$ ranges from 0 (empty graph) to $n(n-1)$ (complete graph) which is the total number of ordered pairs in $\mathcal{L}$. We can represent the elements of $\mathcal{L}$ graphically by drawing a line. If there is a tie present between actors i and j we will write $i \rightarrow j$. In the literature such a graph is called a directed graph and directed lines are referred to as arcs. In the case of undirectional relations we cannot distinguish between the line from actor i to j and from j to i and the set $\mathcal{L}$ will contain at most $n(n-1)/2$ elements. The graph is called an undirected graph and the lines are called edges.

When we deal with valued relations the appropriate graph theoretic representation are valued graphs. A graph (or digraph) is a valued graph or a valued digraph if each line (or arc) carries a value from the set of real numbers. A valued graph $\mathcal{G}_{\mathcal{V}} = (\mathcal{N}, \mathcal{L}, \mathcal{V} =)$ consists of three sets of information: a set of actors $\mathcal{N}$, a set of lines $\mathcal{L}$ and a set of values $\mathcal{V} = v_1, v_2, \ldots, v_L$ attached to the lines. There are several special valued graphs: signed graphs in which positive lines have value $+1$ and negative lines have the value -1 or Markov chains whose values are probabilities and their corresponding sociometrix are referred to as transition matrices or stochastic matrices.

Figure 1.1 presents examples of directed and undirected graphs. The first data is a data set originally gathered by Sampson [89] in a study of the social interactions among 18 members of a monastery. Each node represents a monk and a directed edge from one to another indicates that the first nominated the second as one of his three best friends at any of the three time points when the survey was administered (18 nodes and 56 arcs). The second data is a data set of Padgett [74], consisting of marriage ties among leading Renaissance Florentines, it is a undirected network with 16 nodes (families) and 20 edges (marital ties).

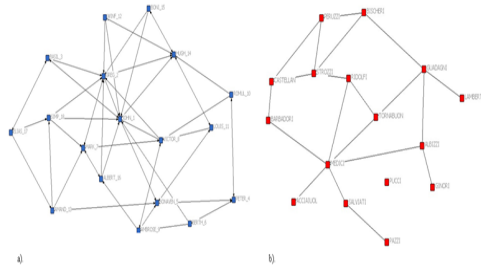

Figure 1.1: Examples of a) directed graph and b) undirected graph

1.2.2 Matrices

A social relation can also be represented by an $n \times n$ *sociomatrix* (the analogue of the *adjacency matriax* in graph theory), $\mathbf{Y}$, which with its elements is assumed for the following analysis as random variables. We denote by

$$Y_{ij} = \begin{cases} 1 \text{ if } (i,j) \in \mathcal{L} \\ 0 \text{ otherwise} \end{cases} \tag{1.1}$$

a random variable where Y_{ij} = 1 if there is a network tie from actor i to actor j, and where Y_{ij} = 0 if there is no tie (Figure 1.2 a). We specify y_{ij} as the observed value of the variable Y_{ij} and we let $\mathbf{Y}$ be the matrix of all variables, with y the observed matrix with elements y_{ij}. For most social relations, one does not have a relationship with oneself, so that on the diagonal of the adjacency matrix there are structural zeros Y_{ii} = 0. In the case of non-directional relations, where $Y_{ij} = Y_{ji}$ and the two variables are not distinguished, the adjacency matrix will be a symmetric one (Figure 1.2 b).
For the following analysis the observed network y will refer to the network data collected by the researcher and which modeling the researcher is interested in. The observed network is seen as one particular pattern of ties out of a large set of possible patterns. It is regarded as a realization from a set of possible networks with similar important characteristics (at the very least, the same number of actors).

a)

	R	B	A	B	P	L	V	W	J	G	H	B	M	A	A	B	E	S
ROMUL_10	0	1	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0
BONAVEN_5	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
AMBROSE_9	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
BERTH_6	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
PETER_4	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
LOUIS_11	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
VICTOR_8	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
WINF_12	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0
JOHN_1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0
GREG_2	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0
HUGH_14	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0
BONI_15	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0
MARK_7	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0
ALBERT_16	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0
AMAND_13	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
BASIL_3	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	1	1
ELIAS_17	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	1
SIMP_18	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	0	0

b)

	A	A	B	B	C	G	G	L	M	P	P	P	R	S	S	T
ACCIAIUOL	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
ALBIZZI	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0
BARBADORI	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
BISCHERI	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0
CASTELLAN	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0
GINORI	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GUADAGNI	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	1
LAMBERTES	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
MEDICI	1	1	1	0	0	0	0	0	0	0	0	1	1	0	1	1
PAZZI	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
PERUZZI	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0
PUCCI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RIDOLFI	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0
SALVIATI	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
STROZZI	0	0	0	1	1	0	0	0	0	1	0	1	0	0	0	0
TORNABUON	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0

Figure 1.2: Adjacency matrices for respectively a) directed graph and b) undirected graph presented in Figure 1.1

1.3 Notations in SNA

In this paragraph firstly we will present some principal definitions and notations which have the fundamental role in social network analysis. Then we will introduce dyadic and triadic theories which are essential for the methods discussed in the following chapters. We will also present some classes of probability distributions which are important in statistical modeling for relational data.

1.3.1 Local and global measures

One of the properties which have an important effect in the modeling of social networks is the *nodal degree*. The *degree* of a node $i \in \mathcal{N}$, denoted by $d(i)$, can be very informative in statistical modeling to study some tendencies toward higher-order network properties such as reciprocity. The degree of a node i is defined as the number of edges incident with i. A node of degree 0 is an isolated node. The nodal degrees are equal to either the row sums or the column sums,

$$d(i) = \sum_j y_{ij} = \sum_i y_{ij} = y_{i+} = y_{+j}. \tag{1.2}$$

In the case of a directed graph the nodes have both *out-degrees* and *in-degrees*. *Out-degree* of the node i, $d_O(i)$, defined as the total of the row i of the adjacency matrix $\mathbf{Y}$, explains how many arcs originate with the actor i,

$$d_O(i) = \sum_j y_{ij} = y_{i+}. \tag{1.3}$$

The *in-degree* of the node i, $d_I(i)$, is defined by the total of the column i of the matrix $\mathbf{Y}$ and explains how many arcs end with the actor i,

$$d_I(i) = \sum_j y_{ji} = y_{+i}. \tag{1.4}$$

Both the out-degree and in-degree take on integer values ranging 0 and $n-1$ and sum to L [108].

The *degree distribution*, denoted by (d_0,d_1,...,d_{n-1}), is a distribution of the counts of nodes with different degree k where $k = 1, \ldots, n-1$. It is a fundamental characteristic, which makes it possible to compare networks and classify them according to their properties.

While we consider an individual node in a network, nodal degree is the simplest characteristic. In the case of considering a network as a whole, the simplest structural characteristic is *density* which involves the number (and the proportion) of the edges in the whole graph.

Let L be the number of edges present in a graph which can take on any integer value from 0 to $n(n-1)$. We define the *density* of a graph, Δ, as the proportion of the number of edges present, L, to the maximum possible number of edges in a graph. It can be calculated as:

$$\Delta = L/n(n-1)/2, \tag{1.5}$$

The density of a network keeps track of the relative fraction of links that are present in a network, and because the average degree equals $2L/n$ the density is simply the average degree divided by $(n-1)$.

Much of the interest in network analysis deals with indirect relationships among

nodes. To capture the indirect interactions in a network, it is important to model *paths* through the network. In the case of an undirected network, a *path* from $v \in \mathcal{N}$ to $v \in \mathcal{N}$ is an alternating sequence of vertices and edges, beginning with v and ending with s, so that each edge connects its preceding node to its succeeding one. The length of a path is the sum of the weights of its edges. A connected graph has paths between all pairs of nodes. The *geodesic*, $d(v, u)$, is the the shortest path from v to u, taken to have infinite length if there is no path between the two nodes. The *geodesic distribution* is defined by the number of actor pairs for which the shortest path between them is of length k, for each value of $k = 1, 2, \ldots$. Pairs of nodes that are not connected are classified as $k = \infty$.

1.3.2 Definition and importance of dyads and triads

In the social network analysis statistical methods are referred to as local methods since the basic unit for statistical analysis are the subgraphs rather than the whole network. In SNA a local structure is defined as the regularities in a social system of actors and relations that can be studied at the level of subgraphs, rather than the entire set of actors involved in the network. A *subgraph* is a subset of nodes taken from $\mathcal{N}$ and all the arcs or edges (digraph and graph) between them. Dyads and triads, the node-generated subgraphs, are two important structures which provide a local view of the entire network. Since dyadic triadic methods operate at the analytic level of these structures and rely on the isomorphism of them, it is necessary to introduce isomorphic subgraphs.
Briefly, two subgraphs, $\mathcal{G}_{s1}$ and $\mathcal{G}_{s2}$ are *isomorphic* if there is a one-to-one mapping from nodes of $\mathcal{G}_{s1}$ to nodes of $\mathcal{G}_{s2}$ that preserves the adjacency of nodes. Further, they are identical, except for possibly different labeling of the nodes [108].

- Dyads

Dyadic analysis provides some important methods to study the idea of reciprocity. The interests of researchers in the strength of tendency for an actor to reciprocate choices of other actors were evaluated from the 1930's and plays very important role

in social network analysis. A *dyad* (2-subgraph) is an unordered pair of actors and the arcs between them. Since a 2-subgraph is the smallest possible social group it is a fundamental element of connectivity and frequently the basic unit for the statistical analysis of social network. For actors i and j where $i \neq j$ and $i < j$ there are four possible dyads divided in three dyadic isomorphism classes.

- D_{ij}=(Y_{ij}, Y_{ij})=(1,1)- a mutual dyad;
- D_{ij}=(Y_{ij}, Y_{ij})=(0,0) - a null dyad;
- D_{ij}=(Y_{ij}, Y_{ij})=(1,0) - an asymmetric dyad;
- D_{ij}=(Y_{ij}, Y_{ij})=(0,1) - an asymmetric dyad.

The first class consists of mutual dyads, in which both actors in a pair relate to each other. The second one is a class of asymmetric dyads in which the single choice is not reciprocated, either i relates to j or j relates to i, but not both. The last class consists of null dyads, in which neither actor has a tie to the other. We should underline that, for local methods (dyadic and triadic methods), it is necessary to consider the direction of the ties since the states of local structures depend on them dramatically. In a graph an unordered pair of nodes can be in only one of two states: either two nodes are adjacent or they are not adjacent. Thus, there are only two dyadic states for an undirected relation represented as a graph; either the actors in the dyad have a tie or, they do not. In a directed network there are $n(n-1)$ ordered pairs of actors but the number of dyads is $\binom{n}{2} = n(n-1)/2$ because every pair of actors is considered just once. In an undirected network the number of dyads is equal to the number of edges.
Dyadic methods define the triple $< MAN >$ as the dyad census, where M, A, N are the numbers of mutual, asymmetric and null dyads. We obtain it by examining all dyads in the digraph and categorizing each into its classes.
The frequencies M, A and N can be calculated both from the elements of $\mathbf{Y}$ or using matrix operations on $\mathbf{Y}$: $M = \sum_{i<j} Y_{ij}Y_{ji}$, $A = L - 2M$, $N = \binom{n}{2} - A - M$.
A statistical dyadic analysis is only possible if we consider each of the elements of the matrix $\mathbf{Y}$ to be a random variable. This implies that $\binom{n}{2}$ are also random variables and also the counts of the dyad census to be random variables. There are

many dyadic methods which allow us to examine different theoretical concepts. One of these techniques is the index ρ_{KP} proposed by [61]. It provides an important tool for measuring the tendency of actors in a network to reciprocate choices more frequently than would happen simply by chance. To calculate the value of ρ_{KP} we use the expectation of the number of mutual dyads, assuming that actors make choices in some random way. Further, $-\infty < \rho_{KP} \leq 1$, where the value less than 0 means that there are tendencies away from mutual dyads, toward assymetrics and nulls. If ρ_{KP} equals to 0 there is no tendency for reciprocity and, 1 indicates maximal tendency. There two different estimates of ρ_{KP} dependent on the network data collection designs: fix choice and free choice procedures (see more in chapter 13 of [108]).

- Triads

Triadic methods allow us to examine fundamental theoretical concepts such as structural balance and transitivity. The first one was introduced by Heider [40] who examined individuals' viewpoints of social situations. Mathematical formalization and quantification of his idea were made with help of graph theory by Hararay [39] and Cartwright and Hararay [11]. The analysis of structural balance is based on measuring relations for a set of actors considering ties with a sign or a valance. As Holland and Leinhardt [43] [44] recognized, structural balance essentially involves the notion of transitivity in graphs which has been shown to be a key structural property in network data.

A *triad*(a 3-subgraph), T_{ijk}, is defined as a set of three actors i, j, k where $i \neq j \neq k$ and the relationships between them. We consider $i < j < k$ since the order of the actors is important. The number of triads derives from $\binom{n}{3}$ which explains the number of ways that we can take n actors, three at a time. In a graph, a triad may be in one of four possible states, depending on whether, zero, one, two or three lines are present among the three nodes in the triad (Figure 1.3). In a digraph there are 2^6=64 possible values for a triad because each of three actors can relate to two other actors and each of these six arcs can be present or absent. If we ignore the actor labels, then there are sixteen possible isomorphism classes of triads (Figure 1.4). In the MAN labeling of triads proposed by Holland and Leinhardt [43], each of these sixteen classes denoted: 003, 012, 102, 021D, 021U, 021C, 111D, 111U, 030T, 030C, 201, 120D, 120U, 102C,

Figure 1.3: The four possible triadic states in a graph

210 and 300. Here, the numbers correspond to the number of mutual (M), asymmetric (A) and null (N) dyads in the triad and the letters correspond to the orientation of the asymmetric dyads: D for down, U for up, T for transitive and C for cyclic. The vector $T = (T_{003}, T_{012}, , T_{300})^t$, which contains the frequencies for each type of isomorphism triad classes, is called the triad census, introduced by Davis and Leinhardt [16] [17]. It furnishes a complete classification of all triads in a directed graph and provides a convenient way to reduce the entire adjacency matrix **Y** to a smaller set of summary statistics.

As underlined previously, transitivity is one of the most important concepts studied by triadic methods Holland and Leinhardt [45] [51] Johnsen [59] [59].

To represent transitivity in undirected graphs we will use the presence of triangles and 2-stars where a *triangle* is a subgraph of three nodes (a triad), each pair of which is connected by an edge. A *k-star* is a subgraph of $(k+1)$ nodes in which one central node is connected by an edge to exactly k other nodes. A triad will be defined intransitive if a 2-star is not part of a triangle. Since structural balance is the outcome of a local process, it is not necessary to control the whole graph to determine whether a particular triad of nodes is balanced or not. The proportion of transitive to intransitive triads in the graph is an aggregate measure of the tendency toward transitivity and structural balance, and hence that of clustering which refers to a tendency for pairs of nodes to be connected to each other if they are connected to a common third node. The local *clustering coefficient* C_i of a node i is the proportion of pairs of nodes to which node i is connected, and those are connected to one other. Because any triangle contains three 2-stars, the clustering coefficient C can be defined as the proportion $3T/S_2$, where T is the number of triangles and S_2 the number of 2-stars in the graph as a whole [72].

As will be seen in Section 2.2.3, the statistics T and S_2 are also very important in a particular class of ERGMs - Markov graphs.

In the case of directed graph a triad involving actors i, j and k is transitive if whenever i relates to j ($i \rightarrow j$) and j relates to k ($j \rightarrow k$) and so i relates to k ($i \rightarrow k$). Intransitive triads are those which are not transitive, and triads which are neither transitive nor intransitive are called vacuously transitive. Considering the Figure 1.4 it is worth noting that only four triads are transitive (030T, 120U, 120D, 300), five are vacuously transitive (003, 012, 021U, 021D, 102) and seven are intransitive (021C, 111D, 111U, 030C, 201, 120C, 210). Some of the intransitive triads are more transitive and some less transitive e. g. 210 is more transitive than 201. Statistical analysis of the sixteen frequencies of the triad census can be used to test theories of structural balance and transitivity. To examine the triad census we can use random directed graph distributions, assuming that the census has been drawn from a random directed graph. It is worth noting that the distribution of the triad census is complicated, except for its mean and covariance matrix. Fortunately, it has been proven that the distribution of triad census can be well-approximated by the multivariate normal distribution [43] [46] [51].

1.3.3 Probability distributions

Statistical methods usually begin with an assumption that the data under investigation are the realizations or observations on a collection of random variables. Random graph probability distributions make it possible to study subgraphs statistically. The basic questions we want to answer is what kind of distribution the random variables follow. Then these distributions allow researchers to test hypotheses about various properties of a graph and a digraph. The fundamental family of probability distributions for directed graphs is the class of uniform distributions. Those range from the simplest possible distribution for a directed graph with n nodes to conditional distributions which fix or condition on a number of graph characteristics. The statistical methods for the dyad census focus on the number of arcs present in a digraph (inference using

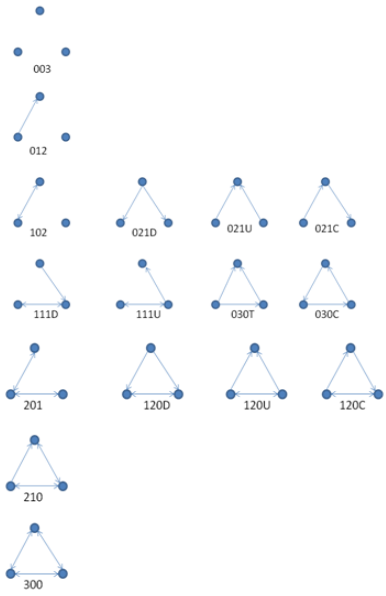

Figure 1.4: The triad isomorphism classes

the simple distributions) and the number of mutual dyads present in a digraph (which require conditional uniform distributions).
Below we present definitions of the Uniform and Bernoulli distributions.

Let $G_d(\mathcal{N})$ be the set of all possible labeled directed graphs with n nodes. We record labels of the nodes to be able to consider all possible digraphs with various characteristics. Without labels assigned to the nodes we can only consider the number of isomorphism classes. We assumed that we analyze a single dichotomous relation so each element of adjacency matrix has two possibilities 0 or 1. As we stated above in a digraph there are $n(n-1)$ arcs which provide $2^{(n(n-1))}$ different labeled adjacency matrices. It is worth noting that the number of elements of $G_d(\mathcal{N})$ grows exponentially with n.

- Uniform and Bernoulli distributions

The simplest distribution on $\mathcal{G}_d(\mathcal{N})$ is the uniform distribution denoted by U and $\mathbf{Y} \sim U$ means that an adjacency matrix representing a particular digraph is distributed as a uniform random variable so every realization is equally likely. The sample space is exactly $G_d(\mathcal{N})$ which contains $2^{(n(n-1))}$ labeled digraphs so the uniform probability function is $P(\mathbf{Y}= y) = 1/2^{n(n-1)}$. To describe the arcs of the digraph under the uniform distribution it is possible to use statistically independent Bernoulli random variables with all probabilities of the presence of arc being equal to 1/2 for

$$P(Y_{ij} = 1) = \begin{cases} 1/2, i \neq j \\ 0, i = j \end{cases} \tag{1.6}$$

All elements of adjacency matrix are independent of all other elements, and the probability distribution of any one of the elements is the simplest possible distribution, the Bernoulli distribution with equal probabilities [22]. This distribution assumes that all actors choose about one-half of the other actors, so that the expected degree is $(n-1)/2$ for all actors and the expected density is 0.5. The uniform distribution can be generalized to a family of Bernoulli distributions by altering the probability that any element of the random adjacency matrix is equal to 1. These models can also generate parametric classes of distributions.

The general Bernoulli distribution is described by

$$P(Y_{ij} = 1) = \begin{cases} P_{ij}, i \neq j \\ 0, i = j \end{cases}, \tag{1.7}$$

where $0 < P_{ij} < 1$ and P_{ij} may differ from element to element to let some actors relate to other actors with different probabilities tendencies. P_{ij} will denote $P(Y_{ij} = 1)$ the probability that a specific arc is present in the digraph. The Bernoulli distribution assumes that the elements of $\mathbf{Y}$ are statistically independent. This distribution permits some of arcs in a random digraph to have greater probabilities of being present than other arcs. If a random digraph is distributed as Bernoulli with P_{ij}=1/2 for all $i \neq j$ then the random digraph is uniformly distributed. In the case of all P_{ij} being equal but different from 1/2 the distribution is not uniform.

An important concept is statistical conditioning which implies the restriction of attention to only those random graphs with the particular properties that are conditioned upon. It is based on taking a random variable and then we derive the conditional distribution of that random variable conditioning on the specific graph properties. Several families of uniform distributions on digraphs can be formed by considering conditional uniform distributions. To obtain the sample space of a conditional uniform distribution we can take all the possible digraphs with n nodes and from this set remove all those digraphs that do not have the specific characteristics fixed before. Then we obtain the revised sample space with digraphs which are equally likely to occur under a conditional uniform distribution.

- Conditional Uniform Distributions

To form some families of uniform distributions on directed graphs we can use conditional uniform distrubutions of which some examples will be presented below (for more details and other examples see [108]. To obtain the sample space of a conditional uniform distribution we can consider all the possible directed graphs with n nodes and then restrict attention to digraphs with certain characteristic fixed before. Then after removing all those digraphs without the specific characteristic we will obtain the desired sample space in which all the remaining digraphs are equally likely to occur.

To study the randomness of choice made by each actor we can use a uniform distribution, conditional on the number of arcs, $U|L = y_{++}$. It is the simlest conditional uniform distribution which conditions the graph property L [62]. This distribution gives equal probability to all digraps with L arcs, and zero probability to all digraphs which do not have y_{++} arcs. Its sample space includes only those digraphs with L arcs. This conditional uniform distribution has a probability mass function

$$P(Y = y) = \begin{cases} 1/\binom{n(n-1)}{l} \text{ if } y_{++} = 1 \\ 0, \text{ otherwise} \end{cases}, \tag{1.8}$$

where $\binom{n(n-1)}{l}$ is the number of ways to distribute l arcs to the $n(n-1)$ possible arc locations.

Since there are $2^{n(n-1)}$ possible digraphs for a set with n nodes and there are $\binom{n(n-1)}{l}$ digraphs in the sample space for $U|L$ so the number of random digraphs that have exactly l arcs is usually quite a bit smaller than the total number of all possible digraphs with n nodes.

Katz and Powell [62] proposed a unifrom conditional distribution which gives the possibility to control for the outdegrees of each node. $U|Y_{(i+)}$ is the uniform distribution which conditions the fixed set of outdegrees of the nodes in the directed graph. In this case every digraph which presents specified outdegrees, $Y_{1+} = y_{1+}$,$Y_{2+} = y_{2+}$,...,$Y_{n+} = y_{n+}$ occurs with equal probability, and digraphs with outdegrees different from the set of specified outdegrees are not in the sample space. To understand the probability function for $U|Y_{(i+)}$ we have to know how many digraphs exist with a specified collection of outdegrees which means understanding how a fixed number of 1's can be allocated for the ith row of a sociomatrix. This fixed number of 1's can be placed in certain positions at random while each of the remaining entries in the row equals 0. Since the rows themselves are independent with the row sums of a sociomatrix fixed, each row can be treated separately to calculate the number of possibilities of distributing a fixed number of 1's to the $n-1$ positions in each row. The probability function for this distribution is defined

$$P(Y=y) = \begin{cases} \prod_{i=1}^{n} 1/\binom{n-1}{y_{i+}}, \text{ if } Y_{i+} = y_{i+} \text{ for all } i \\ 0, \text{ otherwise} \end{cases}, \tag{1.9}$$

where $\binom{n-1}{y_{i+}}$ is the number of ways to place the correct number of 1's in the ith row of the sociomatrix. Thus the sample space has $\prod_{i=1}^{n} \binom{n-1}{y_{i+}}$ elements and it depends on both the number of nodes n and the outdegrees that the random directed graphs are forced to have.

A very important distribution for statistical modeling for social network is the distribution referred to as, $U|Y_{(i+)}, Y_{(+j)}$ which conditions both choices made by actors and choices received. It means that it combines the conditioning of $U|Y_{(i+)}$ and $U|Y_{(+j)}$ and all digraphs with the specified values of the outdegrees and the indegrees are equally

likely. Digraphs that do not satisfy these constraints are absent from the sample space and cannot occur. Although a lot of researche has been proposed regarding this distribution, unfortunately it is impossible to write down its probability mass function. Katz and Powell [62] studied relation between this distribution and other the simpler conditional uniform distributions. Snijders [92] presented how to enumerate all digraphs with fixed indegrees and outdegrees which makes it possible to conduct permutation tests of graph properties such as the number of mutual dyads. Another important distribution is the distribution which controls the choices made, the choices received and the types of dyads. It is the uniform distribution that conditions on the dyad census, the out-degrees and the in-degrees, referred to as $U|M, Y_{(i+)}, Y_{(+j)}$.

In this chapter we have presented some standard terminology and notations for SNA. In the next chapter we will discuss Exponential Random Graph Models, the class of models extensively used to model global social network structure as a function of their "local features". We will deepen the concept of dependence assumptions and illustrate particular classes of ERGMs implied by them. We will also explain the concept of homogeneity and other constraints which make it possible to define a well-specified model.

CHAPTER 2

Exponential Random Graph Models

In this chapter we will present logic and definition of the Exponential Random Graph Models (p*) and two general ways of viewing this class of models. Then to present the development of ERGMs we will focus on a range of different models, from the simpler to more analytically complicated, and their utility in explaining social processes.

2.1 Definition of Exponential Random Graph Models

The Exponential Random Graph Models for social networks are the stochastic models for graphs and which attempt to represent the stochastic mechanisms that produce ties, and the complex dependencies that this induces. This class of models forms a statistical exponential family and expresses each relational tie as a stochastic function of actor or network structural properties. Using ERGMs the principal goal is to estimate model parameters from an observed network and then evaluate how adequately the model represents the data. In general, we do not know what stochastic process generated the observed network, and our goal in formulating a model is to propose a plausible and theoretically principled hypothesis for this process. One of the research questions may be whether the structure of the observed network presents significantly more, or less, characteristics of interest than expected by chance.
The Exponetial Random Graph Models, also known as the p* class of models, was first

discussed in 1986 by Frank and Strauss [23] who viewed it as a generalization of model p1 introduced by Holland and Leinhardt in 1981 [52]. Frank and Strauss presented the general specification of p* but, first of all, they paid attention to the Markov model, the special case of p* models. Further developments of the p* family of models has been given by Wasserman and Pattison [109] and Pattison and Wasserman [76] who showed how a Markov parametric assumption provides just one of many possible sets of parameters. Most of the initial investigations focused on nondirected and directed single, dichotomous relations. In the literature there are a lot of works which present extensions of Exponential Random Graph models to valued [86] and multiple relations [76] [64].
Because the p* model has an exponential term in the right hand side, such distributions have been referred to as Exponential Random Graph Models (ERGMs) and this name will be used in the following analysis.

There are two different approaches to arrive at the Exponential Random Graph family class of models. On the one hand this class of models could be derived from the Hammersley-Clifford theorem [5] as a natural consequence of dependence graphs [23] [76]. On the other hand, it could be viewed as an autologistic regression model [109] [3].
Robins et a. [84] proposed the five steps which are fundamental not only because they make it possible to connect theoretical decisions to data analysis but also because they help to locate certain earlier network models within the class of exponential random graph models.
A general framework for exponential random graph model construction is:

- Step 1: each network tie is regarded as a random variable,
- Step 2: a dependence hypothesis is proposed, defining contingencies among the network variables,
- Step 3: the dependence hypothesis implies a particular form to the model,
- Step 4: simplification of parameters through homogeneity or other constraints,
- Step 5: estimate and interpret model parameters.

The first step provides an assumption that a tie is a random variable which implies that there is a stochastic framework with a fixed node set and there is no a full knowledge about relationship formation. It states that our model is not going to make perfect deterministic predictions and as a result there is going to be some statistical "noise", or lack of regularity, that cannot be successfully explained. Below we introduce dependence graphs and the importance of dependence assumptions in statistical modeling of social network. Then we will explain how dependence assumptions and constraints such as homogeneity influence the form of different models. Details about the final step which is estimation will be presented in Chapter 3.

2.1.1 Dependence assumptions and dependence graph

An important aspect of any probabilistic model of a network is to consider the statistical dependencies among the ties. Construction of a dependence graph gives the possibility to do it. In the context of a social network a dependence graph indicates which relational ties are conditionally independent. As suggested by Frank and Strauss [23], it is fundamental to define dependence structures for the random variables Y_{ij} because the entries of adjacency matrix $\mathbf{Y}$ cannot be assumed to be independent. The dependence assumptions may be represented in a dependence graph, first introduced into the network literature by Frank and Strauss, following the approach of nearest neighbors described by Besag [5]. The dependence structure for random variables Y_{ij} is determined by a dependence graph $\mathcal{D}$ of the random array $\mathbf{Y}$, whose nodes, $\mathcal{N}_{\mathcal{D}}$, are all possible relational ties in the original relation and whose ties specify which ties in the relation are conditionally dependent, giving the remaining relational ties. Two ties are conditionally dependent if the conditional probability that the ties are both present, given the other ties in the network, is not equal to the product of their marginal conditional probabilities. The dependence graph has lines connecting all pairs of conditionally dependent ties. Since there are a lot of ways of specifying conditional dependence between a pair of ties it leads to a variety of distinct dependence graphs. It is worth noting the importance of dependence assumptions in distinguishing among a variety of different random graph types and in constraining types of configurations

relevant to the different models [83]. The definition of dependence graph provides the basis for the Hammersley-Clifford theorem [5], presented below, which establishes that a random graph has a probability that depends only on the complete subgraphs, or cliques of the dependence graph.

2.1.2 The Hammersley-Clifford Theorem

The theorem presented in this section, the Hammersley-Clifford theorem, provides the important link between dependence structures and interactions. It establishes that a probability model for $\mathbf{Y}$ depends only on the cliques of the dependence graph $\mathcal{D}$. The Hammersley-Clifford theorem demonstrates that suffiecient statistics for a loglinear model for the random graph are of the form $\prod_{(i,j)\in\mathcal{A}} y_{ij}$ where $\mathcal{A}$ is a complete subgraph of the dependence graph $\mathcal{D}$ and the product is computed across a set of edges in the original graph that are pairwise conditionally dependent, given the rest of the graph. A subset $\mathcal{A}$ of the set of nodes $\mathcal{N}_\mathcal{D}$, which represent relational ties in the original relation, is complete if every pair of nodes in $\mathcal{A}$ is linked by an edge of $\mathcal{D}$. A subset comprising a single node is also regarded as complete.
According to the Hammersley-Clifford theorem, $P(\mathbf{Y} = y)$ may be characterized by an exponential family of distributions:

$$P(\mathbf{Y} = \mathbf{y}) = \exp \sum\nolimits_{A\subseteq\mathcal{N}_\mathcal{D}} \varpi_\mathcal{A} u_A(y)/\kappa, \tag{2.1}$$

where:

- $\kappa = \sum_Y \exp \sum_{\mathcal{A}\subseteq\mathcal{N}_\mathcal{D}} \varpi_\mathcal{A} u_A(y)$ is normalizing factor;
- $\mathcal{D}$ is the dependence graph for $\mathbf{Y}$;
- $u_A(y) = \prod_{(i,j)\in\mathcal{A}} y_{ij}$ is the sufficient statistic corresponding to the parameter $\varpi_\mathcal{A}$;
- $\varpi_\mathcal{A}$=0 whenever the subgraph induced by the nodes in $\mathcal{A}$ is not a clique of $\mathcal{D}$.

The set of nonzero parameters in this probability distribution for $P(\mathbf{Y} = y)$ depends on the maximal cliques of the dependence graph. Any subgraph of a complete subgraph is

also complete but not maximal, so if $\mathcal{A}$ is maximal clique of $\mathcal{D}$, then the probability distribution for the graph will contain nonzero parameters for $\mathcal{A}$ and all its subgraphs. Using this theorem Frank and Strauss [23] proved how log-linear probability models can describe random graphs with a general dependence structure. Examples of different dependence assumptions and their associated models are given in Section 2.2.

2.1.3 Definition of ERGMs

The assumptions about dependencies among the random tie variables in a network and the Hammersley-Clifford theorem determine the general form of the probability distribution of Exponential Random Graph Model:

$$P(\mathbf{Y} = y) = \frac{\exp\{\theta' u(y)\}}{\kappa(\theta, \Upsilon)} \tag{2.2}$$

where $\mathbf{Y}$ is an adjacency matrix of a random network on n nodes, Υ is the support of $\mathbf{Y}$, the set of all possible networks with n nodes.
The vector θ, $\theta \in \Omega \subset \Re^q$ $\Omega = \{\theta \in \Re^q : \kappa(\theta, \Upsilon) < \infty\}$ is a vector of parameters which are the weights of the linear combination, and are usually unknown and must be estimated.
The vector $u(y)$ is a known q-vector of network statistics based on y. The goal in defining the vector of statistics $u(y)$ is to choose statistics that summarize the social structure of the network (e.g. number of edges, number of triangles). These statistics should match the purpose for which model parameters are being estimated and networks are being simulated. The model itself depends on the hypothesized structural features of the network because the vector of statistics is chosen a priori and this choice depends on the questions of interests. It is worth noting that ERGM terms cannot be simply taken and used in one field (i.e. social networks), to be then used to represent processes in another field (i.e. biological networks). The effective choice of terms for an ERGM depends on theory and context. Which network statistics appear in the model depends on the structure of the hypothesized dependence graph, and on whether any homogeneity

constraints have been proposed.

The denominator $\kappa(\theta, \Upsilon)$ is the normalizing factor. It makes all the probabilities sum to 1 and guarantees that the distribution is proper.

$\kappa(\theta, \Upsilon)$ depends on both θ and the support Υ and it is defined as:

$$\kappa(\theta, \Upsilon) = \sum_{z \in \Upsilon} \exp\{\theta' u(z)\} \tag{2.3}$$

Specification of 2.3, including the number of nodes, n, is an important aspect of model 2.2. For a fixed n, Υ may contain up to $2^{n(n-1)}$ networks, a very large number even for moderated-size n. The dependence of $\kappa(\theta, \Upsilon)$ on the unknown parameter vector carries the primary barrier to inference when using this model.

The ERGMs can be used also to analyze models in which networks of relational ties depend on the actor attributes which allow for additional covariate information about the network. To incorporate attribute information into the ERGM we need to replace $u(y)$ with $u(y, X)$ in 2.2 to allow X, which is a matrix of attributes of the nodes in the network, to influence the probability distribution of $\mathbf{Y}$. In this case we are interested in the probability of the graph y given the observations of attributes x, that is, $P(\mathbf{Y} = y | X = x)$. Important examples have been proposed where two different processes relate attribute and network variables. The first one is social selection processes [81] in which individual attributes may contribute to the formation or change of network ties what means that individuals may change their relationships on the basis of the attributes of other individuals. The second is the social influence processes [82] where network structure can affect individual characteristics in that individuals may be influenced by others with whom they have network ties.

- Structural parameters: interpretations and constraints

Applying ERGMs we expect that local statistics can capture the global structure in the network. The structure of a typical graph in this distribution can be interpreted as the result of a cumulation of these particular local configurations. The goal in defining the vector of statistics $u(y)$ is to choose statistics that summarize the social structure of the network and match the purpose for which themodel parameters are being estimated.

In the ERGMs the parameters are assumed to govern the probabilistic nature of the social or/and behavioral process of interest and reflect structural concerns. Parameters corresponding to the network statistics may be interpreted in terms of their contribution to the "likelihood" of occurrence of network with the relevant feature of interests. For a given observed network y, parameter estimates indicate the strength of effects in the data. If the parameter is large and positive, we expect to observe the corresponding configuration in graphs in distribution 2.2 more frequently than if the parameter were zero (defined to be zero when configuration occurs by chance).

There are several ways in which constraints on the parameters of ERGMs may be applied. As defined in the Section 2.1.3, Ω is the parameter space whose dimension, q, is at most $N - 1$ for the saturated model. In order to define a model clearly, we need to simplify it and reduce the number of parameters. Some parameters need to be set to zero, equated or otherwise constrained by introducing hypothesized constraints on the values of parameters associated with larger configurations [99].

According to Frank and Strauss [23], the reduction in the number of parameters is often done by imposing standard isomorphic homogeneity constraint that all isomorphic graphs $\mathcal{G}$ have the same probability $P(\mathcal{G})$. In this case the parameters are equated if the configurations are the same when we ignore the labels (a prori indistinguishable) on the nodes (in which case the configurations are said to be isomorphic). We do not introduce parameters specific to the different vertices. One of the examples of homogeneity constraint is proposing a single tendency for reciprocity across the entire network, by assuming that the reciprocity parameters for each possible reciprocated tie are all equal.

Another method of applying constraints may be obtained by equating parameters for isomorphic configurations involving similar types of actors. For instance, in the case of reciprocity we could allow the reciprocity effect to depend on node characteristics. In school friendship networks, one reciprocity parameter for boy-boy configurations, one for girlgirl and another for boy-girl configurations could be proposed.

During the process of choosing the vector of statistics $u(y)$, it is important to ensure that there are no linear dependencies among the specific terms. For instance, we should avoid using both edges and density in the same model because they summarize the

same structural property.

2.1.4 ERGMs as logit model

The formulation of log linear model closely relates the ERGMs to logistic regression and makes it possible to view this class of models in a very standard response/explanatory variables. In logistic regression the response variable is a logit [the logit is a function defined as $logit(p) = log(p) - log(1-p)$], or log odds of the probability that a relational tie is present, and the explanatory variables can be quite general.

- *Change statistics*

In order to specify an alternative version of model 2.2 three new relations have been proposed [109]:
Y_{ij}^{+}, Y_{ij}^{-}, Y_{ij}^{c}, where Y_{ij}^{+} is a matrix formed from $\mathbf{Y}$ where the tie from i to j is forced to be present:

$$(Y_{ij}^{+})_{mk} = \begin{cases} Y_{ij}, \text{ if } (m,k) \neq (i,j) \\ 1, \text{ if } (m,k) = (i,j). \end{cases} \tag{2.4}$$

Y_{ij}^{-} is defined as a matrix formed from $\mathbf{Y}$ where the tie from i to j is forced to be absent:

$$(Y_{ij}^{-})_{mk} = \begin{cases} Y_{ij}, \text{ if } (m,k) \neq (i,j) \\ 0, \text{ if } (m,k) = (i,j). \end{cases} \tag{2.5}$$

Note that Y_{ij}^{+} and Y_{ij}^{-} differ at most from $\mathbf{Y}$ by the (i,j) the entry, which is forced to be 1 or 0 respectively. We define Y_{ij}^{c} as a matrix for the complement of Y_{ij}:

$$(Y_{ij}^{c})_{mk} = \begin{cases} Y_{ij}, \text{ if } (m,k) \neq (i,j) \\ undefined, \text{ if } (m,k) = (i,j) \end{cases} \tag{2.6}$$

Y_{ij}^{c} represents the rest of the network other than the single variable Y_{ij}. Now we can introduce the vector of change statistics which is a function of a particular choice $u(\cdot)$

of network statistics, an observed network y, and a particular pair of different nodes (i,j). It is defined as:

$$\delta(y)_{ij} = u(y_{ij}^{+}) - u(y_{ij}^{-}) \tag{2.7}$$

where, following definitions presented above, y_{ij}^{+} and y_{ij}^{-} represent the networks realized by fixing y_{ij}=1 or y_{ij}=0, respectively, while keeping all the rest of the network exactly as in y itself. In other words, $\delta(y)_{ij}$ is the change in the value of the network statistic $u(y)$ that would occur if y_{ij} were changed from 0 to 1 while leaving all of the rest of y fixed. When the network statistics involve covariates X in addition to y we may add X to the notation and write $\delta(y, X)_{ij}$.

Holland and Leinhardt [52] were the first to develop a log-linear model for network data. Using the dichotomous nature of the random variable Y_{ij} the model 2.2 can be turned into a logit model not for the probability of the graph, but for the conditional probabilities of the relational ties. The expression in an alternative way of the full conditional distributions of Y_{ij} has the advantage of not depending on the normalizing constant κ. Using the vector of change statistics, the probability that the tie from i to j is present, conditional on the complement of Y_{ij}, is the following:

$$P(Y_{ij} = 1|Y_{ij}^{c}) = \frac{P(Y = y_{ij}^{+})}{P(Y = y_{ij}^{+}) + P(Y = y_{ij}^{-})} = \frac{\exp(\theta' u(y_{ij}^{+}))}{\exp(\theta' u(y_{ij}^{+})) + \exp(\theta' u(y_{ij}^{-}))}. \tag{2.8}$$

We next consider the odds ratio of the presence of a tie from actor i to actor j to its absence, which simplifies model 2.8:

$$\frac{P(Y_{ij} = 1|Y_{ij}^{c})}{P(Y_{ij} = 0|Y_{ij}^{c})} = \frac{\exp(\theta' u(y_{ij}^{+}))}{\exp(\theta' u(y_{ij}^{-}))} = \exp\left\{\theta' \left[u(y_{ij}^{+}) - u(y_{ij}^{-})\right]\right\} \tag{2.9}$$

Taking the natural log of both sides we will obtain the conditional log-odds of an edge:

$$\log \frac{P(Y_{ij} = 1|Y_{ij}^{c})}{P(Y_{ij} = 0|Y_{ij}^{c})} = \theta' \left[u(y_{ij}^{+}) - u(y_{ij}^{-})\right] = \theta' \delta(y)_{ij}, \tag{2.10}$$

where θ is a vector of effect parameters to be estimated and $\delta(y)_{ij} = u(y_{ij}^{+}) - u(y_{ij}^{-})$ is the change defined above. It is worth noting that logits are simple and well defined

when the relation is dichotomous. In the case of valued relation one must be careful which logits to model. It was discussed at length in [86].

2.2 Different types of ERGMs

As mentioned above dependence assumptions have the consequence of picking out different types of configurations as relevant to the model. Below we present some types of models which derive from different dependence assuptions.

2.2.1 Bernoulli random graph model

The first type is the Bernoulli random graph [18] [24], which represents the simplest dependence assumption that all possible distinct ties are independent of one another. This class of graphs derived from the assuptions of conditional independence for all edges (Y_{ij} and Y_{kl} are independent whenever $i \neq k$ and $j \neq l$). A Bernoulli graph assumes complete independence of relational ties and its dependence graph is an empty graph which comprises a collection of isolates nodes, each corresponding to an edge in the Bernoulli graph. According to the Hammersley-Clifford theorem, the only configurations relevant to the model are those in which all possible ties in the configuration are conditionally dependent on each other. When all possible ties are independent, the only possible configurations relate to single edges Y_{ij} so the general form of Bernoulli model is:

$$P(\mathbf{Y} = y) = \frac{\exp \sum_{i,j} \theta_{ij} y_{ij}}{\kappa\left(\theta, \Upsilon\right)}, \tag{2.11}$$

where

$$\log \kappa\left(\theta, \Upsilon\right) = \sum_{i,j} \log(1 + \exp(\theta_{ij})). \tag{2.12}$$

A parameter for each of these configurations is $\theta_{ij} = logitP(\mathbf{Y} = y)$ which is the log-odds of a tie in the dyad D_{ij}. For model 2.2, q is equal to $n(n-1)/2$. The elements of the vector of network statistic, $u(y)$, are just y_{ij} and tell us whether that configuration

is observed or not.

We obtain a special case of the Bernoulli random graph model if we impose a homogeneity assumption which means that there is a fixed probability for all possible edges across the graph. It implies that $q = 1$. The parameter $\theta_{ij} = \theta$ for all i and j is called edge or density parameter and is related to the probability of a tie being observed. In this spacial case the network statistic $u(y) = L(y) = \sum_{i,j} y_{ij}$ is the number of ties in the graph y.

$$P(Y = y) = \frac{\exp(\theta L(y))}{\kappa(\theta, \Upsilon)} \tag{2.13}$$

This homogeneous Bernoulli model is also known as the Erdös-Rényi model [18] and it has been extensively studied in [71] and [2].

The simplicity and homogeneity that make the Bernoulli model tractable also make it less useful as a realistic model for social phenomenon. For instance, if friendship networks could be reasonably described as Bernoulli graphs, we could claim that any pair of humans has a given propensity to friendship, irrespective of whatever other friendships may occur. In practice it is not possible because relations among friends are much more complex.

2.2.2 p1 model (dyadic independence model)

When we deal with directed networks, more complicated assumption have to be adopted. In this case dyads, rather than edges, are indepedent of one another. The dyadic independence models are models in which, regardless of the nature of the relation, dyads are assumed to be statistically independent, so that the joint probability distribution, and hence the likelihood function, is simply a product of dyadic probabilities. Unfortunately, this class of models, because of their simply independence assumption, present some limitations in describing real social phenomena.

The dyadic independence models derive from the assumption of dyad independence in which Y_{ij} is conditionally dependent only on Y_{ji}, given the rest of the graph. For undirected networks, dyadic independence models are defined as those in which $P(Y_{ij} = y_{ij})$

is independent of $P(Y_{kl} = y_{kl})\ \forall (i,j) \neq (k,l)$ conditional on the actor attributes. One of the examples of dyadic idependence models is the p1 model which expresses the two elementary social tendencies of reciprocation and differential atraction (including popularity and expansiveness effects) [47] [52]. Holland and Leinhard denoted the probability $P(\mathbf{Y} = y)$ by p1 to emphesize this probability distribution as the first or simplest family of distributions on digraphs that might have been considered for social network data.

Let Υ be the set of all directed graphs with independent dyads. This model can represent arbitrary nodal indegree and outdegree marginal distributions and strength of reciprocity (mutuality) within dyads. It can be written as:

$$p1(y) = P(\mathbf{Y} = y) = \exp(\rho m + \theta y_{++} + \sum_i \alpha_i y_{i+} + \sum_j \beta_j y_{+j})/\kappa(\rho, \theta, \alpha_i, \beta_j) \tag{2.14}$$

where m, y_{++}, y_{i+} and y_{+j} are the values of M, Y_{++}, Y_{i+} and Y_{+j} computed from y. In this model ρ can be considered the mutuality effect; α_i is the sender effect of the ith node and β_j represents the receiver effects of the jth node.The θ represents the overall edge effect (like intercept in a linear regression).

One important question to consider is if p1 is a conditional uniform distribution for random digraphs. The answer is yes. The p1 is an exponential family of distributions with minimal sufficient statistics consisting of the indegrees, the outdegrees and the number of mutual dyads. If we condition on specific values of $Y_{(i+)}$,$Y_{(+j)}$ and M, all random digraph with these values have exactly the same probability of occurring. Thus the p1 model is identical to the random digraph distribution $U|M, Y_{(i+)}, Y_{(+j)}$. Since the basic modeling unit of p1 consists of the dyad, not of the individual tie or arc, the p1 can thus be viewed as a Bernoulli dyad distribution rather than a Bernoulli arc distribution, where all dyads, not ties or arcs, are assumed to be independent.

An extension of the p1 model is the p2 which assumes dyadic independence but conditional on node-level attribute effects [66] [103]. The p2 model is more realistic. It works well, especially, when attribute effects are expected to be strong and the structure is expected to arise from the attributes.

2.2.3 Markov model (dyadic dependence model)

The Markov random graphs of Frank and Strauss [23] are a particular class of ERGMs which are more realistic than simple probabilistic models such as the Bernoulli model and dyadic independence models. These models are generalizations of Markov random fields designed for spatial interaction models [5] which are based on the Ising models of rectangular arrays of binary variables or lattice. The simplest examples represent observations as points on a lattice, and assume that only the nearest neighbors have an influence on the status of a site.

Frank and Strauss [23] defined a graph $\mathcal{G}$ to be a Markov graph (or graph with Markov dependence) if $\mathcal{D}$ contains no edge between disjoint sets i, j and k, m in $\mathcal{L}$. Markov dependence is one, in which a possible tie from i to j is assumed to be contingent on any other possible tie involving i or j, even if the status of all the other ties in the network is known. In this case, the two ties are said to be conditionally dependent, given the values of all the other ties which means that if the value of one tie changes, the probability of the other tie is affected, even if all the other ties in the network remain the same. Markov dependence can be characterized as the assumption that two possible network ties are conditionally dependent when they have a common actor. This implies that non-incident edges in $\mathcal{G}$ are conditionally independent.

Figure 2.1presents a Markov graph $\mathcal{G}$ with 4 nodes and 6 edges and its dependence graph $\mathcal{D}$. It illustrates the correspondence between cliques of $\mathcal{D}$ and sufficient subgraphs of $\mathcal{G}$ (i.e. in Figure 2.1the clique $(12, 13, 23)$ of the dependence graph $\mathcal{D}$ corresponds to the triangle $(1, 2, 3)$ in the graph $\mathcal{G}$).

For general Markov graphs the cliques of $\mathcal{D}$ correspond to sets of edges so that any pair of edges within the set must be incident. According to the Hammersley-Clifford theorem, Frank and Strauss [23] proved that such sets are just triangles and stars, defined respectively as:

$$T(y) = \frac{1}{6}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} y_{ij}y_{jk}y_{ki} \tag{2.15}$$

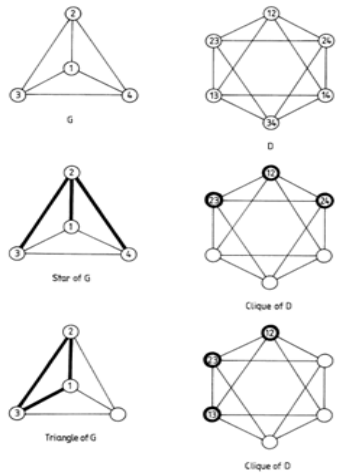

Figure 2.1: A Markov graph $\mathcal{G}$ and its dependence graph $\mathcal{D}$. Source: [23]

and

$$S_k(y) = \frac{1}{k!} \sum_{i_0,\ldots,i_k} \cdots \sum y_{i_0 i_1} \cdots y_{i_0 i_k}, \tag{2.16}$$

where k=1, ..., $n-1$.

Therefore, there are $\binom{n}{3}$ distinct triangles, $\binom{n}{2}$ distinct 1-stars (edges) and $n\binom{n-1}{k}$ distinct k-stars for k=2, ..., $n-1$.

The number of sufficient subgraphs are in total:

$$\left(\begin{array}{c} n \\ 3 \end{array} \right) + \left(\begin{array}{c} n \\ 2 \end{array} \right) + \sum_{k=2}^{n-1} n \left(\begin{array}{c} n-1 \\ k \end{array} \right) = n2^{n-1} + \left(\begin{array}{c} n \\ 3 \end{array} \right) - \left(\begin{array}{c} n+1 \\ 2 \end{array} \right) \tag{2.17}$$

When we deal with directed graph the cliques of $\mathcal{D}$ correspond to arcs, mutual arcs, stars of order 3 or more and triangles of various kinds. A homogeneous Markov random graph model for undirected network is defined as

$$P(Y = y) = \exp \sum_{k=1}^{n-1} \sigma_k S_k(y) + \tau T(y) / \kappa(\sigma_k, \tau) \tag{2.18}$$

where $T(y)$ is a count of triangles and $S_k(y)$ is the number of k-stars in y for $1 \leq k \leq n-1$. For a given observed network y, parameter estimates indicate the strength of effects in the data. In the case of the Markov model, a large and positive estimate for τ suggests that, given the observed number of edges and stars, networks with more triangles are more likely, that is, there is a strong transitivity effect in the network.
Figure 2.2 illustrates configurations of the Markov model for directed and undirected networks. Note that the terms in the Markov model are often related to each other, in

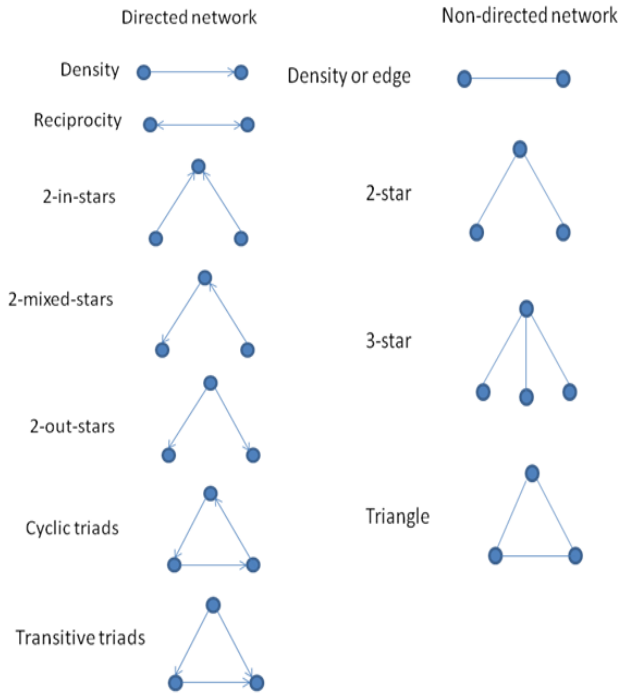

Figure 2.2: Configurations and parameters for Markov random graph models

the sense that some are higher-order of the others (i.e. a 3-star in a nondirected network centered on node i is composed of three 2-stars (and three edges) also centered on i.) This feature has its importance when we interpret the model for describing social

phenomenon. In the case of a network with many 2-stars, some of them will form triangles just by chance. But if there is a substantial triangle effect in a Markov random graph model, it is over and above any 2-star effect. We can infer that triangulation did not occur simply because of the chance overlapping of many 2-stars (or indeed of many edges) but that it is an important process in this network, independent of other effects. In practice, the model defined above has been simplified reducing the terms to edges, 2-stars and triangles appropriate for capturing, both, transitivity and clustering. This special case is called triad model and is of the form:

$$P(Y = y) = \exp(\theta L(y) + \sigma_2 S_2(y) + \tau T(y))/\kappa(\theta, \sigma_2, \tau) \tag{2.19}$$

where $S_2(y)$, $S_2(y)$ are the numbers of 2-stars and $T(y)$ is the number of triangles, in the observed network y.

Unfortunately, the triad model presents some shortcomings in producing reasonable networks and describing real social phenomenon. The reason is related to the problem of degeneracy [34] which will be introduced and explained in chapter 3. Robins et al. [85] suggested that to solve this problem it may be important to include also at least the 3-star and 4-star effects, that is, at least first three moments of the degree distribution. An alternative but useful approach, proposed by Snijders et al. [99] Robins et al. [87], assumes constraints on the relationship between all higher-order star parameters and lower-order star parameters (see Section 2.2.4). Some developements of the Markov random graph models have been proposed for multivariate networks [76], for valued networks [86] and for affiliation networks [91] [78].

2.2.4 Higher order model specifications

In this section we focus on "higher order" statistics proposed by Snijders et al [99] such as the alternating k-triangle, alternating k-two path and alternating k-star and their reparametrized versions such as the geometrically weighted degree, the geometrically weighted edgewise shared partner and the geometrically weighted dyadwise shared

partner [56]. These terms demonstrate improvements over the homogeneous Markov model and have been proven to be more robust and very effective in overcoming the problems of degeneracy ([99] [56], [87]). We refer to these parameters and their associated models as "higher order" because they include configurations with more than 3 nodes.

- Alternating k-stars, alternating k-triangles and alternating k-paths

One of the statistics proposed by Snijders, et al. [99] is an alternating k-star which combines counts of all the Markov star parameters with geometrically decreasing weights on the higher order star counts so that they did not come to dominate the calculation. The alternating k-star assumption proposes that, rather than setting higher order star parameters to equal 0, all star parameters be retained in the model but with a linear constraint among parameter values such that, for all $k \geq 2$, $\sigma_{k+1} = -\sigma_k/\lambda$ for some λ greater than 1.
The alternating k-star statistic is defined as:

$$u = \sum_{k=2}^{n-1} (-1)^k \frac{S_k}{\lambda^{k-2}}. \tag{2.20}$$

The statistic 2.20 includes stars of all orders but, for λ greater than 1, the impact of higher order stars is reduced for higher k and they have alternating signs. In that sense, this assumption is rather more general than simply forcing higher order star parameters to have value 0.
The combination of star parameters into the alternating k-star parameter attempts to express a given functional form to the degree distribution. If the alternating k-star parameter is positive, then highly probable networks are likely to contain some higher degree nodes ("hubs"), whereas a negative parameter suggests that networks with high degree nodes are improbable. This parameter helps with degeneracy because presence of alternating signs addresses the balance of a model between positive and negative star parameters to prevent the resulting graph distribution from being forced towards containing largely complete or largely empty graphs.

As new higher order structures of transitivity for nondirected graphs Snijders et al. [99] proposed a k-triangle which is a combination of k separate triangles that all share one edge, the common base of the k triangles (Fig. 2.3).

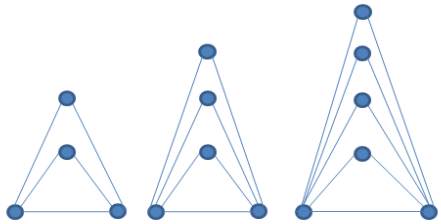

Figure 2.3: 2-, 3- and 4-triangles

Counts of the k-triangle configurations are combined into one statistic producing the alternating k-triangle defined as:

$$v = \sum_{k=2}^{n-1} (-1)^k \frac{T_k}{\lambda^{k-2}} \tag{2.21}$$

where T_k is the number of k-triangles in the network. The corresponding k-triangle parameter $\tau = \tau_1$ corresponds to a ordinary triangle configuration, with the additional constraint that $\tau_{k+1} = -\tau_k/\lambda$, where τ_k is the parameter corresponding to a k-triangle. The alternating k-triangle statistic does not simply represent triangulation in the network but additionally is a measure of the extent to which triangles themselves group together in larger higher order "clumps" in the network. The parameter associated with the alternating k-triangle statistic models the triangulation in the network but permit more heterogeneity. It is better than the Markov single triangle parameter in dealing with clumps of triangles that form the denser regions of the network. A large positive parameter value indicates that there is substantial transitivity effects in the network, and that this is likely to be expressed in the formation of denser regions.

The alternating k-triangle assumption moves beyond the dependence assumptions underlying Markov random graph models, utilizing instead the partial dependence concept

[77]. The additional dependence assumption is presented by Snijders et al. [99] as social circuit dependence, where the presence of two edges in the observed graph creates dependence among two other possible edges, assuming the four edges constitute a 4-cycle. One of the situations, where such social circuit dependence might occur, can be in an organization. Suppose Tim usually worked with Katy and John with Mike. Then if John and Tim commenced work on a new project, the chances of Katy and Mike also working together might increase. In other words, the chances of Katy and Mike working together are increased by John and Tim working together, but only when John already works with Mike and Tim with Katy [87].

The third configuration introduced by Snijders [99] is a k-two-path which is a lower order configuration for a k-triangle and it is identical to the k-triangle except that the edge at the base of the k-triangle is not necessarily present (Figure 2.4). This configuration quantifies multiple independent paths between pairs of nodes.

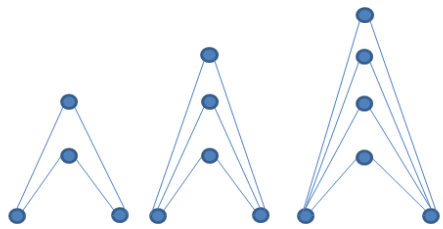

Figure 2.4: 2-, 3- and 4-paths

The motivation of creating the k-two-paths is to introduce a parameter that, when used in conjunction with k-triangles, will make it possible to distinguish between tendencies to form edges at the sides, or at the base of a k-triangle. It is worth noting that the sides of a k-triangle in the absence of the base represent a type of edge clustering that is only the precondition to transitivity, while the presence of the base edge reflects transitive closure. A combination of k-two-paths are incorporated into one statistic

which is defined by:

$$w = U_1 - \frac{2}{\lambda}U_2 + \sum_{k=3}^{n-2}\left(\frac{-1}{\lambda}\right)^{k-1} U_k \tag{2.22}$$

where U_k is the number of k-two-paths equal $\sum_{i<j}\binom{L_{2ij}}{k}$ for $k \neq 2$ and for $k = 2$ $\frac{1}{2}\sum_{i<j}\binom{L_{2ij}}{2}$. L_{2ij} is the number of two-paths connecting nodes i and j.
It is worth noting that the choice of suitable values of λ depends on the data set but it should be the same for the k-triangles and the alternating k-two-paths, but may differ from the value used for the alternating k-stars.

- Geometrically weighted degree, geometrically weighted edgewise shared partner and geometrically weighted dyadwise shared partner

The geometrically weighted edgewise shared partner, the geometrically weighted dyadwise shared partner and the geometrically weighted degree [56] are the reparameterized higher order terms presented above. From a modeling perspective, these geometrically weighted terms are useful because they are not merely counts of local network configurations, like the degree of k-star statistics; instead, they are particular linear combinations of an entire distribution of degree or shared partner statistics.
The geometrically weighted degree statistic which is a reparameterized version of the alternating k-star statistic can be defined by:

$$u(y, \phi_s) = e^{\phi_s}\sum_{i=1}^{n-1}\left\{1 - (1 - e^{\phi_s})^i\right\} d_i(y) \tag{2.23}$$

It is a scalar for a fixed network y and parameter ϕ_s, obtained by a linear combination of the degree $d_i(y)$ that depends on the tuning parameter ϕ_s [54]. This statistic models the degree distribution but puts more weight on the numbers of nodes with lower degrees, with weights decreasing geometrically as the degrees increase.
Models containing the alternating k-star or its reparameterized version geometrically weighted degree distribution can capture degree distributions well in the right circumstances (i.e. [29]). They permit greater heterogeneity in the degree distribution, so that

it is more capable of modeling high degree nodes than a small number of low order Markov star parameters.

The configurations of k-triangles also introduce a new distribution of graph features: the edgewise shared partner distribution [54]. The geometrically weighted edgewise shared partner statistic is the weighted sum of number of edges in the network that have exactly k shared partners, weighted by the geometric sequence, $(1 - e^{-\phi})^k$ and it is defined as:

$$v(y, \phi_t) = e^{\phi_t} \sum_{i=1}^{n-2} \left\{1 - (1 - e^{\phi_t})^i\right\} EP_i(y) \tag{2.24}$$

where $EP_i(y)$ is the number of edges in y between two nodes that share exactly i neighbours in common-the number of edges that serves as the common base for exactly i distinct triangles. It models the distribution of shared partners of tied actors, but with weights decreasing geometrically as the number of shared partners increase.

The geometrically weighted dyadwise shared partner is the weighted sum of number of pairs (i, j) such that i and j share exactly k neighbors in common, whether or not y_{ij} =1; weighted by the geometric sequence, $(1-e^{-\phi})^k$, where ϕ is a decay parameter [56].

$$w(y, \phi_p) = e^{\phi_p} \sum_{i=1}^{n-2} \left\{1 - (1 - e^{\phi_p})^i\right\} DP_i(y) \tag{2.25}$$

where $DP_i(y)$, that is the number of $pairs(i, j)$ such that i and j share exactly i neighbours in common, whether or not $y_{ij} = 1$. The parameter associated with this statistic models the distribution of shared partners of actors who may or may not be tied, but with weights decreasing geometrically as the number of shared partners increase.

Both, the alternating statistics and the geometrically weighted statistics depend on parameters, λ and ϕ, respectively. If these parameters are fixed and known, then these statistics do not present any special difficulties because they can be easily included as components of $u(y)$. However, if values of λ or ϕ are considered uknown it would be desirable to estimate an optimal value by maximum likelihood method. In this case, the model resulting from including one or more of these terms in $u(y)$ is not one of the standard ERGMs but it forms a curved exponential family, which makes more difficult

the estimation procedure [56]. Hunter and Handcock in the work [56] present how to extend estimation based on approximations derived from a Markov chain Monte Carlo scheme to curved exponential family models.

In practice a lot of dyadic dependence models in the estimation produce serious numerical problems for many networks of different size. For this reason it is impossible to fit them well. The models with the new statistics seem to be more robust and adequate also for large networks. Goodreau [29] fits the new specifications to a network of over thousand nodes and demonstrates how to evaluate model fit. Hunter et al. [57] managed to obtain realistic parameter estimates and appropriately fit models containing these new statistics for network with more than two thousand nodes.

CHAPTER 3

Statistical inference for ERGMs

In this chapter we focus on topics important for statistical modeling for social network such as different estimation methods, simulation networks using MCMC algorithms and problem of degeneracy. These concepts are fundamental to the issues addressed and discussed in the next chapter.

3.1 Estimation methods

The goal of applying a statistical model 2.2 for the network, regardeless if the model is constructed from specific dependence assumptions or viewed as autologistic regression, is to be able to describe the global features by a low number of local structures. To do it we need to estimate the unknown parameters θ of a set of configurations $u(y)$ and then interprete them. The criterion used for estimation is the maximum likelihood criterion whose the basic idea is to obtain the best parameter estimates in such a way that the observed network (which is a particular graph in the distribution) presents the highest possible probability of being replicated by the given model when the model is used to simulate a network.

Parameter estimates, as well as estimates of the uncertainty of estimation make possible to explore the range of network outcomes predicted by the model and to infer whether any model parameter is significantly different from zero and so whether the corresponding configuration is present in the observed graph.

3.1.1 Likelihood inference for ERGMs

In the case of dyadic independence models the likelihood function is easy to write down and the estimation of the model parameters is not very difficult. For dyadic dependence models because of their complex dependence structures the estimation becomes more complicated.
The likelihood function (which is the product of the probabilities for each dyad) for the general form of ERGM (2.2) is:

$$L(\theta) = \frac{\exp(\theta' u(y))}{\kappa(\theta, \Upsilon)}. \tag{3.1}$$

Differentiating the loglikelihood function:

$$\ell(\theta) = \theta^t u(y) - \log \kappa(\theta, \Upsilon) \tag{3.2}$$

shows that the maximum likelihood estimate $\hat{\theta}$ satisfies

$$E_{\hat{\theta}}[u(Y)] = u(y_{obs}), \tag{3.3}$$

where $u(y_{obs})$ is the vector of observed network statsitics, E_θ is the expectation under P, so that it at least ensures that the probability mass of the ERGM is centered at $u(y_{obs})$. Although the likelihood function presents a simple expression, its dependence on the normalizing constant $\kappa(\theta, \Upsilon)$ which makes difficult to obtain the maximum likelihood estimation for an exponential family random graph model. Table 3.1 presents the number of elements in Υ ($2^{n(n-1)/2}$ in the case of undirected network) as the number of actors grows.

Number of actors n	**Number of elements in Υ**
7	2097152
10	3.518437e+13
40	6.359114e+234

Table 3.1: Number of network in Υ for different number of actors n

It is worth noting that in the case of networks the number of nodes cannot be treated as a

traditional sample size. So colled "effective sample size" for a graph of n nodes depend on model specification. In the case of the Bernoulli model for undirected network edges are independent and the true sample size is $\binom{n}{2}$. For more complex models two networks with qualitatively similar features but with different size might have a tottaly different MLE.

Handcock [33] presented condition for existence and uniqueness MLE for social network models and he underlined that many properties of the MLE can be derived from statistical exponential family theory [4].
The maximum likelihood estimator (MLE) for θ is:

$$\hat{\theta} = argmax_{\theta \in \Omega} P(Y = y_{obs} | u, \Upsilon) \tag{3.4}$$

Denote the relative interior of C by rint(C) and the relative boundary by $rbd(C) = cl(C)/rint(C)$ where C is the convex hull of $\{u(y) : y \in \Upsilon\}$ [33].
Result:

- The MLE exists if, and only if, $u(y_{obs}) \in rint(C)$
- If it exists, it is unique. In addition, when it exists, it can be found as the unique solution to 3.3 or, equivalently, as 3.4, the unique local minima of 3.2.
- A necessary and sufficient condition for the MLE not to exist is that $u(y_{obs}) \in rbd(C)$. This occurs with positive probability.

Handcock summarized that, for MLE, in practice attempting to numerically maximize the likelihood leads to unbounded estimates when the observed graph has statistics falling on the relatively boundary of C. This typically means the optimization algorithm does not converge, or otherwise converges to a false maxima (for more details see [33].

Until recently, inference for ERG models has been almost exclusively based on an alternative local approximation to the likelihood function referred to as the pseudo-likelihood, proposed by Besag [6] and suggested by Strauss and Ikeda [102] for parameter estimation in Markov random graph models. Currently the favored methods for statistical inference are Markov Chain Monte Carlo (MCMC) Maximum Likelihood Estimate (MLE) proposed by Geyer and Thompson [27] applied to ERGMs by Corander

et al. [12], Crouch and Wasserman [14], Snijders [95], Handcock [34]. Therefore, algorithms to compute the MPLE for ERGMs are typically deterministic while the algorithms to compute their MLEs are typically stochastic. ERGMs have usually been expressed in the natural θ-parameterization (called also canonical parameterization) but there have been considered for the model 2.2 two alternative types of parameterization [33].

- Mean parameterization

First one is the mean value parameterization (the μ-parameterization), in which the natural parameter θ is replaced by parameter $\mu(\theta)$, which corresponds to the expected value of the sufficient statistic $u(Y)$ under the model with natural parameter θ [4]. The μ-parameterization is defined by

$$\mu(\theta) = E_\theta(u(Y)) \tag{3.5}$$

where where $\mu : \Omega \to rint(C)$ and C is the relative interior of the convex hull of the sample space of $u(Y)$. The mapping is strictly increasing in the sense that

$$(\theta_a - \theta_b)^t (\mu(\theta_a) - \mu(\theta_b)) \geq 0 \tag{3.6}$$

with equality only if $P_{\theta_a}(Y = y) = P_{\theta_b}(Y = y) \forall y$. The mapping is also injective in the sense that

$$\mu(\theta_a) = \mu(\theta_b) \to P_{\theta_a}(Y = y) = P_{\theta_b}(Y = y) \forall y. \tag{3.7}$$

Using 2.2, $\mu(\theta)$ can be reexpressed as

$$\mu(\theta) = \left[\frac{\partial \log[\kappa(\theta)]}{\partial \theta}\right](\theta) \tag{3.8}$$

and its gradient as

$$\nu(\theta) = \left[\frac{\partial^2 \log[\kappa(\theta)]}{\partial \theta^t \partial \theta}\right](\theta). \tag{3.9}$$

For each natural parameter θ for the model 2.2 there is a unique μ-parameter corresponding to that model (and vice versa). In practice the value $\mu(\theta)$ can be obtained by computing the average of $u(Y)$ for graphs Y simulated from the model with natural parameter θ. One advantage of this parameterization is its interpretability. Since it is defined on the scale of network statistics it gives superior performance than the θ-parameterization. The parameter corresponding to each graph statistic is just its expected value over the population of graphs and the MLE parameter estimates coincide with the value of the observed network statistics: $\hat{\mu} = g(y_{obs})$. The capacity to reproduce the observed network statistics and the population-average interpretation make it possible to evaluate immediately a certain model specification and understand better the degeneracy problem. However, this fact that $\hat{\mu} = g(y_{obs})$ is of little practical value, since we cannot estimate standard errors or even simulate from the fitted model without knowing the inverse image θ of μ. For more details and examples see [33] and [104].

- Mixed parameterization

Second alternative type of parameterization is the mixed parameterization (the η-parametization) that has similar properties to the mean value parameterization. It may be considered in the case where a sub-set of the statistics are most directly interpretable in terms of their mean values and others in their natural parameterization (Handcock 2003a).
Let $(u^{(1)}, u^{(2)})$ be a partition of u such that the first component is that of the statistics interpretable in terms of their mean values. Consider similar partitions $(\theta^{(1)}, \theta^{(2)})$ of θ and $(\mu^{(1)}(\theta), \mu^{(2)}(\theta))$ of $\mu(\theta)$. Let $\Omega^{(2)}$ be the set of values of $\theta^{(2)}$ for θ varying in Ω and $C^{(1)}$ be the convex hull of $\left\{t^{(1)}(y) : y \in \Upsilon\right\}$.
The mapping $\eta : \Omega \longrightarrow \Omega^{(2)} \times \text{rint}(C^{(1)})$ defined by

$$\eta(\theta) = (\mu^{(1)}(\theta), \theta^{(2)}) \tag{3.10}$$

For many random graph models the mixed parameterization presents better interpretability and inferential properties than the natural or mean value parameterizations [33].

Handcock underlined that mean value parameterization and mixed parameterization are a natural way to represent ERGM. Although, they seem to suffer from more indirect specification of the likelihood, their interpretability, ease of estimation and the stability and functionality of their parameter space make them one of solutions to resolve problem of lack of convergence when MCMC methods are used for estimation and simulation networks.

3.1.2 Maximum Pseudolikelihood Estimation

Estimation based on the maximum pseudolikelihood technigue was first proposed by Besag [6] for spatial models. Then this approach was adopted by Strauss [101] to estimate different interactive models and Strauss and Ikeda [102] in order to estimate the parameter of Markov models. The computational tractability of the pseudolikelihood function makes it an attractive alternative to the full likelihood function and it has made ERGMs computationally available.
Consider the conditional formulation 2.10 of the model 2.2, the pseudolikelihood for the ERGM can be explained by

$$\ell_P(\theta) = \prod_{i \neq j} P(Y_{ij} = 1|Y_{ij}^c)^{y_{ij}} P(Y_{ij} = 0|Y_{ij}^c)^{1-y_{ij}} \tag{3.11}$$

and

$$\ell_P(\theta) = \theta \sum_{ij} \delta_g(y)_{ij} y_{ij} - \sum_{ij} \log[1 + \exp\{\theta' \delta_g(y)_{ij}\}], \tag{3.12}$$

where the value of θ that maximizes 3.12 is a maximum pseudolieklihood estimator (MPLE).

Maximizing the pseudolikelihood is equivalent to maximizing the likelihood function for the fit of logistic regression in which the response data consist of each unique element of the adjacency matrix, y_{ij} (treated as an independent observation), and the predictor vectors are given by the change statistics so the maximum pseudolikelihood

estimates of the corresponding ERGM can be obtained using logistic regression computing packages.

Although the maximum pseudlikelihood procedure looks like a logistic regression (which assumes independent observations) but this is not because of the dependencies within the relational data. Since the approach of maximum pseudolikelihood assumes conditional independence of the random variables representing the relational ties, it gives reasonable results only for dyadic independence models (i.e. Bernoulli model) which unfortunately do not fit the data well. In this case the Maximum Pseudolikelihood estimators (MPLE) correspond to the exact solution and the true maximum likelihood estimator may be found via an MPLE computation.

Since the formation of edges in a network has the existing network structure as a condition, it is necessary to construct more complex models that may fit the dependencies among ties well. When the dependence among observations becomes stronger (the dyadic dependence models), it is generally that the statistical properties for MPLE estimators are not well understood and in practice MPLE does not provide a good performance [104]. In this case the parameter estimates may be biased and it could be risky to interprete standard errors from logistic regression output as though they are reasonable estimates of the standard deviations of the pseudolikelihood estimators. It is also difficult to apply usual tests of model fit, for example the pseudolikelihood deviance is not asymptotically distributed as Chi-squared (which would be the case in normal logistic regression). It is worth noting that even though the pseudolikelihood estimation has been used to data as a pragmatic convenience (its speed and determinism), formal statistical inferences should not be made with MPLE technique because of its shortcomings. This approach may be recommended as a first selection criterion for a dyadic independence model but the preferred option is to use Monte Carlo estimation procedures and any final model should be checked with MLE.

Van Duijn et al. [104] proposed an alternative pseudolikelihood estimator, the penalized pseudolikelihood estimator (MBLE). This method was originally introduced by Firth [21] as a general approach to reducing the asymptotic bias of maximum likelihood estimates by penelizing the likelihood function. Then Heinze and Schemper

[41] suggested a penalized likelihood approach in the context of logistic regression. The penalized pseudo-likelihood for the model 2.2 is defined by

$$\ell_{BP}(\theta) = \ell_P(\theta) + \frac{1}{2}\log|I(\theta)| \tag{3.13}$$

where $I(\theta)$ denotes the expected Fisher information matrix for the formal logistic model underlying the pseudo-likelihood evaluated at θ. The estimator that maximizes $\ell_{BP}(\theta, y_{obs})$ is referred to as the maximum bias-corrected pseudo-likelihood estimator (MBLE).

3.2 Markov Chain Monte Carlo Maximum Likelihood Estimation

Only recently there have been developed methods for solving a problem of parameters estimation of ERGMs for social networks. The idea of Markov Chain Monte Carlo Maximum Likelihood Estimation (MCMCMLE) consists of randomly sampling networks $Y_1,\ldots,Y_m$ from the distribution $P_{\theta,\Upsilon}$ simulated using an MCMC procedure and taking the empirical distribution of these networks as an approximation to the true distribution. These methods use a stochastic approximation to built and maximize the likelihood function and make possible to overcome a problem of calculating approximate maximum likelihood estimates of the ERGM parameters. They were first suggested in Geyer and Thompson [27] and later developed for ERGMs by Snijders [95], Robins and Pattison [83] and Hunter and Handcock [56].
An indirect alternative can arise if we arbitrarily choose parameter vector θ_0 and rather than maximize 3.2 directly, we consider instead the log-ratio of likelihood values

$$\ell(\theta) - \ell(\theta_0) = (\theta - \theta_0)^T g(y_{obs}) - \log\left[\frac{\kappa(\theta, \Upsilon)}{\kappa(\theta_0, \Upsilon)}\right], \tag{3.14}$$

where θ_0 is arbitrarily chosen parameter vector.
The approximation of ratios of normalizing constant can be described by [27]:

$$\frac{\kappa(\theta, \Upsilon)}{\kappa(\theta_0, \Upsilon)} = E_{\theta_0} \exp\left\{(\theta - \theta_0)^T g(Y)\right\} \tag{3.15}$$

where E_{θ_0} denotes the expectation assuming that Y has distribution given by $P_{\theta,\Upsilon}$.
Therefore, we may exploit the law of large numbers and approximate the logarithm of the likelihood ratio $r(\theta, \theta_0) = \ell(\theta) - \ell(\theta_0)$ by :

$$\hat{r_m}(\theta, \theta_0) \approx (\theta - \theta_0)^T g(y_{obs}) - \log\left[\frac{1}{m}\sum_{i=1}^{m} \exp\left\{(\theta - \theta_0)^T g(Y_i)\right\}\right], \tag{3.16}$$

where Y_1,...,Y_m is a sequence of graphs simulated from the model $P_{\theta,\Upsilon}$. The strong convergence of $\hat{r_m}(\theta, \theta_0)$ to $r(\theta, \theta_0)$ as $m \to \infty$ is guaranteed by a Markov chain version of the strong law of large numbers. Thus for a fixed sample size m, maximization of $\hat{r_m}(\theta, \theta_0)$ as a function of θ gives an approximation to the maximum likelihood estimator. Since 3.16 simply approximates a population mean by the sample mean it is called "naive"' approximation.
Handcock [33] [34] underlined that the existence and uniqueness of the MCMCMLE can be understood in terms of the statistical exponential family with respect to counting measure on $u(Y_1)$,...,$u(Y_M)$.
Result:
Let CO be the convex hull of sampled sufficient statistics. In practice, there are two situations:

- $u(y_{obs}) \in rint(CO)$ the MCMCMLE exists and is unique. It is found as the unique maximum of the MCMC likelihood.
- $u(y_{obs}) \notin rint(CO)$ but is in $rint(C)$ then the MCMCMLE does not exist, even though MLE exists and is unique.
- $u(y_{obs}) \notin rint(C)$ Neither the MCMCMLE nor the MLE exists.

Handcock explained that the MCMCMLE may not exist for at least two reasons. First, when the MLE does not exist, neither the MCMCMLE will exist. Second,

the MCMCMLE will not exist even in cases where the MLE does, when the model used to simulate the graphs is not close enough to produce realizations that cover the observed values of the network statistics. The stochastic estimation technique described above requires one to select a parameter value θ_0. While the approximation of Equation 3.16 may in theory be made arbitrarily precise by choosing the MCMC sample size m to be large enough [27]. In practice it is extremely difficult to use this approximation technique unless the value θ_0 is "close enough" to the true maximum likelihood estimator θ. The method usually used to choose θ_0 is pseudolikelihood estimation method.

- Newton-Raphson algorithm

One of the methods for the approximation of maximum likelihood estimator used for ERGMs is the Newton-Raphson method [27] [56]. The main idea of this algorithm is to obtain a maximizer of the approximate log-likelihood ratio 3.16 while iterating an approximate Fisher scoring method until convergence [56]. It applies MCMC algorithms to generate a random sample from a particular probability distribution on the sample space of all networks and then this sample is used to approximate the true likelihood function. Finally, the approximated likelihood function is maximized, which yields the parameter estimates.
The scheme of the algorithm is as follows:

1. Select an initial value of θ_0.
2. Generate an MCMC sample $u(Y_1)$,...,$u(Y_m)$ using the parameter θ_0.
3. Iterate the approximate Fishing scoring method 3.17 until convergence, obtaining a maximizer $\tilde{\theta}$ of $\hat{r_m}$.

$$\theta^{(k+1)} = \theta^{(k)} + \left\{ \hat{I}(\theta^{(k)}) \right\}^{-1} \nabla(\theta)^t \left[u(y_{obs}) - \sum_i u_{w_i} \right], \tag{3.17}$$

where $w_i^{(k)} = \frac{\exp\left\{ \left[\theta^{(k)} - \theta^{(0)}\right]^t u(Y_i) \right\}}{\sum_{j=1}^{n} \exp\left\{ \left[\theta^{(k)} - \theta^{(0)}\right]^t u(Y_j) \right\}}$, $u_{w_i} = w_i^{(k)} u(Y_i)$

and

$$\hat{I}(\theta^{(k)}) = \nabla(\theta^{(k)})^t \left\{ \sum_{i=1}^{m} w_i^{(k)} u(Y_i) u(Y_i)^t - \left(\sum_{i=1}^{m} u_{w_i} \right) \left(\sum_{i=1}^{m} u_{w_i} \right)^t \right\} \nabla(\theta^{(k)}).$$

4. If $Var_{MC}\hat{r_m}$ of equation 3.18 is too large compared to $\hat{\ell}(\tilde{\theta})$, then set θ_0 equal to $\tilde{\theta}$ and return to step 2.

$$Var_{MC}[\hat{r_m}] = \frac{1}{m^2 \bar{U}^2} \sum_{k=-K}^{K} (m - |k|)\hat{\gamma}_k \tag{3.18}$$

   where $\bar{U} = \frac{1}{m}\sum_{i=1}^{m} U_i = \frac{1}{m}\sum_{i=1}^{m} \exp\{[\theta - \theta^0]^t u(Y_i)\}$ and γ_k denotes the sample lag-k autocovariance of the sequence U_1, U_2,..., which is assumed to be stationary.

5. Take $\tilde{\theta}$ to be the MCMCMLE.

After the algorithm has converged it is necessary to obtain errors in approximating the true MLE, $\hat{\theta}$, by the MCMCMLE, $\tilde{\theta}$ which are calculated using an estimated MCMC covariance matrxi for θ. The usual error inherent in using the MLE to approximation reality are obtained from the standard asymptotic results and the estimated Fisher infromation matrix $\hat{I}(\theta^{(k)})$ used to compute an estimate $[\hat{I}(\theta^{(k)})]^{-1}$ of the covariance matrix.
More detailed description of this method and its application to fitting curved exponential random graph models can be find in the work of Hunter and Handcock [56].

- Robbins-Monro algorithm

The Robbins-Monro algorithm is a stochastic iterative algorithm and can be used to compute moment estimates, and therefore also maximum likelihood estimates in the ERGM [95]. This method is a stochastic iterative algorithm intended to solve equations of the form

$$E\{Z_\theta\} = 0, \tag{3.19}$$

where Z_θ is a random variable of which the probability distribution is governed by a parameter θ and where realizations of Z_θ can be observed for arbitrary values of θ. In the case of maximum likelihood estimates in ERGM the aim is to solve 3.19 , where Z_θ

is given by $Z_\theta = u(Y) - u_0$ where Y has probability distribution of form 2.2 and u_0 is the observed value of the network statistics.

The algorithm proposed by Snijders [95] distinguishes three phases. In phases 1 and 3 a generation of networks is required by simulating random draws from the exponentail random graph distribution with parameters that depend on the algorithm's phase. In phase 1 generated networks are used to determine a diagonal matrix $D_0 = diag(cov_{\theta_0}(u(Y)))$ to be used in the successive phase. Its diagonal elements are estimates of the derivatives $d_{kk} = \partial E_\theta u_k(Y)/\partial\theta_k, k = 1, 2, ..., p$ evaluated in the initial value θ_0 of the estimation algorithm. The number of steps in this phase is $N_1 = 7 + 3q$. In phase 3 the estimate covariance matrix $\Sigma(\theta) = cov(u(Y))$ of $u(Y)$ is used to estimate the standard error of the model parameters estimates. The number of steps in the third phase is ptoposed to be $N_3 = 1000$.

The phase 2 is the most important one which consists of several subphases. The main goal of this phase is to determine iteratively the estimates parameter, according to the updating step:

$$\widehat{\theta}^{(t+1)} = \widehat{\theta}^{(t)} - a_t D_0^{-1} Z(t), \tag{3.20}$$

where a_t is the step size, D_0 is the diagonal matrix computed in phase 1, and $Z(t)$ for $t = 1, 2, ...$ are random variables so that the conditional distribution of $Z(t)$ given $Z(1), ..., Z(t-1)$ is the distribution of Z_θ (a random variable with probability distribution governed by parameter θ) obtained for $\theta = \widehat{\theta}^{(t)}$. The step size a_t is a sequence of positive numbers converging to 0 and is constant within each subphase. The initial value of a_t in phase 2 is 0.1 and the number of subphases is 4. The number of iteration steps per subphase is determined by a stopping rule, but bounded for subphase k by a minimum value $N_{2k}^- = 2^{4(k-1)/3}(7 + q)$ and a maximum value $N_{2k}^+ = N_{2k}^- + 200$.

The use of this method was proposed also by Snijders [94] for parameter estimation in an actor-oriented model for network evolution.

- Stepping algorithm

As stated above it is extremely difficult to use this approximation techniques unless the value θ_0 is "close enough" to the true maximum likelihood estimator θ. Since the

quality of approximation 3.16 degrades quickly as θ moves away from θ_0, Hummel et al. [53] proposed a sistematic method for moving θ_0 closer to $\hat{\theta}$ step by step.
The idea of this method is to take partial steps toward $\hat{\mu}$ in mean value parameter space by pretending that the MLE $\hat{\mu}$ is not $u(y_{obs})$ but rather some point in between $u(y_{obs})$ and the estimate of $\mu(\theta_0)$. This makes it possible to restrict the search for a maximizer of the approximate log-likelihood ratio to a region where this approximation is reasonably accurate. Conceptually, there is an iterative jump from canonical parameter space to mean value parameter space (by taking means of MCMC samples) and vice versa (by maximizing approximate log-likelihood functions) until a value of θ_0 is obtained close enough to θ to allow one final MCMC-based maximization step. The procedure begins with an initial θ_0, such as $\theta_0 = 0$ or $\theta_0 = MPLE$. Then the vector μ_1 is defined by

$$\hat{\mu}_1 = \gamma_{step}\hat{\mu} + (1 - \gamma_{step})\bar{\mu}_0, \tag{3.21}$$

where $\gamma_{step} = [0, 1]$, $\bar{\mu}_0$ the sample mean of the vectors $u(Y_1)$,...,$u(Y_m)$, where Y_1,...,Y_m is an MCMC sample from the model defined by θ_0. The idea is to treat $\hat{\mu}_1$ as the MLE in this maximization step because the former $\hat{\mu}_1$ is closer to $\bar{\mu}_0$ than the latter, which means that the approximation to the log-likelihood ratio should be better. The maximizer of the resulting approximation will be called θ_1. Since θ_1 is thus "closer" to the MLE than θ_0, in the sense that its corresponding mean-value parameter is closer, then the process is repeated, with θ_1 taking the place of θ_0. The maximiser can be treated as the final MLE when $\hat{\mu}$ is in the convex hull. For more details and example of application see [53].

There are currently three programs available for Monte Carlo maximum likelihood estimation for Exponential Random Graph Models such as "statnet", "pnet" or "SIENA version 3". The "SIENA version 3" in the "StOCNET" suit of programs and the "pnet" program implement the same stochastic approximation algorithm Robbins-Monro described in Snijders [95]. Additional information about these two programs is available from the SIENA website (http://stat.gamma.rug.nl/siena.html) and the Melnet website (http://www.sna.unimelb.edu.au/). The third program is the "statnet" package for R

[35], an integrated set of software tools for the representation, visualization, simulation and analysis of network data. The "ergm", one of the "statnet" packages, allows maximum likelihood estimates of ERGMs to be calculated using Markov Chain Monte Carlo. The "statnet" package provides not only parameter estimates but also various tools for plotting and simulating networks, assessing model goodness of fit and monitoring graphically and numerically the performance of the MCMC algorithms. All computational procedures are sophisticated, incorporating recent advances in MetropolisHastings algorithms and are intended to optimize computational efficiency which means that the program is able to fit many kinds of models for large graphs in a reasonably short time frame. The "ergm" package give the possibility to use three estimation procedures described above the Newton-Raphson, the Robbins-Monro and the Stepping procedures.

3.3 Simulating exponential random graphs using MCMC

Markov Chain Monte Carlo (MCMC) algorithms provide a natural way to simulate social networks [28]. These methods [27] effectively simulate the network over the space of possible graphs in order to maximize the likelihood and was first utilized in this field by Strauss [101]. The MCMC algorithm can be used not only for simulation of the network given the parameters but also to estimate the network model from data and then use the same model, with the empirically based parameter estimates, to drive a simulation of the network. These methods gives the possibility to control the network structures in a simulation and to test how well the estimated models capture the network structure.

One of the convenient ways to generate random graphs is by Gibbs sampling (called also "heat bath") applied to the elements of the adjacency matrix. An initial matrix $Y^{(1)}$ is chosen, and the elements of this matrix are stochastically updated. This updating mechanism circles through the whole matrix again and again, thus defining a stochastic process $Y^{(t)}$ which is a Markov chain; the distribution of $Y^{(t)}$ tends asymptotically to the desired random graph distribution. This procedure implies that the matrices

$Y^{(t)}$ and $Y^{(t+1)}$ differ at most in only one element. At each iteration t, for some choice of (i, j), Y_{ij} is set to zero or one according to the conditional probabilities $P_{\theta_0,\Upsilon}(Y_{ij} = 1|Y^c_{ij} = y^c_{ij})$ and $P_{\theta_0,\Upsilon}(Y_{ij} = 0|Y^c_{ij} = y^c_{ij})$ implied by Equation 2.10. This algorithm chooses the pairs (i, j) uniformly at random, sequentially, or using some mixture of the two. Each update requires the change statistics $\delta(y)_{ij}$ of Equation 2.10 to be determined. A general theorem [26] implies that the distribution of the graphs $Y^{(t)}$ converges for $t \to \infty$ to the exponential random graph distribution.

A simple alternative to Gibbs sampling is a pure Metropolis algorithm in which the proposal is always to change the value of y_{ij}. This proposal is accepted with probability defined by

$$\min\left\{1, \frac{P_{\theta_0,\Upsilon}(Y_{ij} = 1 - y_{ij}|Y^c_{ij} = y^c_{ij})}{P_{\theta_0,\Upsilon}(Y_{ij} = y_{ij}|Y^c_{ij} = y^c_{ij})}\right\} \tag{3.22}$$

The Metropolis scheme is usually preferred over the Gibbs scheme because it results in a greater probability of changing the value of y_{ij}, a property thought to produce better-mixing chains.

The third methods are Metropolis-Hastings algorithms which choose the candidate graph y_{prop} from an auxiliary distribution dependent on the current graph y_{curr} and are aimed at either focusing the transitions or spreading them more broadly throughout Υ. Thus, if $q(y_1, y_2)$ denotes the probability that $Y_{prop} = y_1$ given that $Y_{curr} = y_2$ under this auxiliary distribution, then the proposal is accepted with probability defined by

$$\min\left\{1, \frac{P_{\theta_0,\Upsilon}(Y = y_{prop})q(y_{curr}, y_{prop})}{P_{\theta_0,\Upsilon}(Y = y_{curr})q(y_{prop}, y_{curr})}\right\}, \tag{3.23}$$

where

$$\frac{P_{\theta_0,\Upsilon}(Y = y_{prop})}{P_{\theta_0,\Upsilon}(Y = y_{curr})} = \exp\left\{\theta_0\left[u(y_{prop}) - u(y_{curr})\right]\right\}. \tag{3.24}$$

The Metropolis algorithm of Equation 3.22 may be viewed as a special case in which the auxiliary distribution is symmetric in the sense that $q(y_1, y_2) = q(y_2, y_1)$. If y_{prop} differs from y_{curr} by exactly a single edge toggle, replacing y_{ij} by $1 - y_{ij}$, then $u(y_{prop}) - u(y_{curr})$ is just $\pm\delta(y)_{ij}$. On the other hand, if y_{prop} differs substantially from y_{curr} for a particular type of Metropolis-Hastings proposal, then the ratio of Equation

3.24 can be calculated by considering a sequence of networks, each with one dyad different from the last, starting from the current network and ending at the proposed network. At each step, the ratio is a simple function of the change statistic vector [55].

Metropolis-Hastings algorithms can converge more efficiently than Gibbs sampling to the target distribution when the proposal density $q(\cdot;\cdot)$ is well-chosen. The behavior of MCMC algorithms is also very dependent on the choice of statistics $u(y)$. Although it can be proven that in principle the Metropolis algorithm yields convergence to the desired probability distribution, in practice convergence may take an exceptionally large number of steps. For example if y is a high-probability graph, and its neighboring graphs much less probable, then the algorithm may retain y for a very large number of steps. When this happens, we refer to y as a frozen structure, and to the Markov chain as showing poor mixing [28]. These circumstances are discussed further below in Section 3.4.

3.4 Problem of degeneracy

Development of Monte Carlo estimation methods for ERGMs has allowed to understand better model behavior, especially the model degeneracy and model stability. Handcock [33] [34] conducted a very wide study on degeneracy for ERGMs extensively investigating degenerate parameter regions for two-star models for very small networks. As the basic property Handcock considered that the useful stochastic models should place a significant proportion of their probability mass on graphs that have high probability of being produced by the underlying social process.

A random graph model is defined to be *degenerate* if it places almost all its probability mass on a small number of graph configurations in $\mathbf{Y}$. Hence degeneracy of a graph model occurs when the model places disproportionate probability mass on only a few of the possible graph configurations [34]. A common case is where the distribution places almost all its mass on the empty graph (i.e., $Y_{ij} = 0\ \forall i, j$), and/or the complete graph (i.e., $Y_{ij} = 1\ \forall i, j$), which are not interesting in practical point of view. Basing on the geometry of the mean value parameterization the near degeneracy [33] occurs when

the expected sufficient statistics are close to a boundary of the hull and the model place much probabilities mass on the graphs in the deg(Υ); the set of graph on the boundary of the convex hull $\deg(\Upsilon) = \{y \in \Upsilon : u(y) \in rbd(C)\}$.
For any estimation procedure, it is very important that the model is nondegenerate one, otherwise it is difficult to obtain satisfactory convergence of parameter estimates. Unfortunately, if degeneracy model is used for simulation and MCMC likelihood inference, the approximations to the true model will be very poor. Further, algorithms used for inferential purposes are often inadvertently based on degenerate forms resulting in inferential degeneracy. Thus, for most applications, we should look for ($\mathbf{Y}$, u, θ) which are far from degenerate and we should seek to limit our space of viable models accordingly. Since the effective parameter space of ERGMs is a small, bounded subset of the theoretical parameter space, Handcock [33] indicated the utility of the Bayesian framework as a very promissing approach for network modeling.

Another property mentioned above which have an important consequence on the use of these models in practice is the stability. A random graph model is stable if small changes in the parameter values result in small changes in the probabilistic structure of the model. If this is not the case, then very similar parameter values can describe very different graph structures. Unstable models often have bad statistical properties and do not represent realistic graphs. Conditions for the stability of a model are related to those for degeneracy. The stability of model within the mean value parameterization is geometrically simpler than that of the natural parameterization.

For the ERG class of models, the issue of model degeneracy for certain parameter values was first discussed by Strauss [101]. Strauss was the first who used the Metropolis-Hastings algorithm to simulate random distributions. He observed that it is possible asymptotically that there may be no finite normalizing constant for a distribution with certain parameters version. For these regions of the parameter space, simulations are thus not adequate in producing a stationary distribution.
By model degeneracy, Strauss meant situations where the energy $H(x)$ of a graph y, defined as $H(y) = -\sum_A \lambda_A u(y)$, tended in probability to a minimum as the number of nodes became large. This notion of degeneracy relates to the second limit problem

introduced by Grenander [31] which reffered to the behavior of the probability distribution as the order of the graph (and hence the number of random edges in the model) tends to infinity.

Consider the energy $H(y)$ of a graph y and the minimum energy m_H as the minimum value of $H(x)$ over all graphs in the state space. If K denotes the set of graphs with minimum energy m_H and $1/\gamma$ ($\gamma > 0$) is a scaling constant for parameters, then a scaling of the parameters of model 2.1 yields the model given by

$$P(Y = y) = (1/\kappa) \exp\left\{\sum\nolimits_{A \subseteq \mathcal{N}_{\mathcal{D}}} (\frac{\lambda_A}{\gamma}) u_A(y)\right\}. \tag{3.25}$$

This model can be rewritten in the form:

$$P(Y = y) = \exp\left\{-\left(H(y) - m_H\right)/\gamma\right\} / \left[|K| + \sum\nolimits_{y \notin K} \left\{-(H(y) - m_H)/\gamma\right\}\right], \tag{3.26}$$

from which it is clear that as γ becomes smaller (and the model parameters become larger), the probability of graphs in the minimum energy set K approaches $1/|K|$ and the probability of all other graphs approaches zero. It means that for some choices of parameter values the simulation may reach a particular high-probability graph that remains as the current graph for a large number of steps in the simulation. Such behavior is common as the size of the parameters increases. Grenander [31] described such behavior as freezing and stated the first limit problem about the behavior of the probability distribution while the interactions among variables are made stronger. When γ increases in magnitude, the parameters of the model become very small and the graph distribution approximates the Bernoulli distribution. Conversely, scaling the parameters by an increasingly large constant positive value $1/\gamma$ (small values of γ) inevitably leads to freezing. In practice, as the parameters become larger, the probability density becomes concentrated uniformly on a subset of structures which are often very regular in form and are referred to as frozen patterns. Frozen patterns often occur in "degenerate" regions of the parameter space.

Following Grenander [31], Robins et al. [85] demonstrated that increasing all parame-

ters values by the same factor results in movement toward degenerate regions. Although most human social structures are stochastic, because of the presence of tendencies such as transitivity and structural balance, it is possible that stochastic social systems may be "enough close" to determinism. Robins et al. [85] interpreted degenerate regions as areas where "stochasticism " breaks down and deterministic structures emerge. Simulating Markov random graph models with a judicious choice of parameters Robins et al. examined how "close" an observed graph is to this phase transition from stochasticism to determinism and dimonstrated that this subclass of models can produce the properties of a small world. Contrary to Handcock [34] who argued that degenerate regions imply uninteresting models which are not good at representing the social network data, Robins et al. [85] proved that some degenerate models can be typical of complex systems with phase transitions and result in graphs of theoretical importance (for instance: the caveman graphs of Watts [111] or complete bipartite graphs).

In practice, many of the models proposed in the literature and used for representing real social network data suffer from degeneracy in the sens defined by Handcock [34], especially Markov graph models tend to these phenomena. The problem of degeneracy and poor fitting can be resolved with using of the social circuit models based on recently developed statistics (Section 2.2.4) which have been proven to be more robust and very effective in representing real network data. These statistics seem to capture high-order dependency structure in networks and to contain a lot of significant information about network [99] [56] [87].

Goodreau et al. in [30] warned against assuming model degeneracy too quickly. The problems with convergence can in some cases arise not because the model is a degenerate one, but simply because the Markov chain has not been allowed to run long enough to cover the sample space sufficiently. This is especially true if the starting values $\theta_0$for the estimation are very far from the true MLE. There have been proposed some options that one may explore before going on with the diagnosis of degeneracy. One approach is to increase the size of MCMC sample, or interval, or both. Another option is to decrease the length of steps, increase the maximum iterations through the

simulation-estimation cycle, or both. A final choice is to introduce some constraints on the estimation process, for instance to fix the number of edges in Monte Carlo estimation procedures. In such models there are no density parameters and edges have minor effects on other parameter estimates. Although fixing the number of edges can be helpful to decrease the risk of degeneracy problems but the decision should be based on a theoretical belief that this constraint appropriately captures the stochastic process behind the network, if not, it may simply mask degeneracy rather than overcoming it.

3.5 Identifying model degeneracy and goodness of fit

It is quite complex to say which models are useful for a social network. Next step, after we have fitted the model, is drawing graphs at random from the parameter estimates. It makes it possible to compare the sampled graphs to the observed one and decide if the fitted model is a good one for the data. A good statistical model should capture the significance of various kinds of local processes and still be able to reproduce the observed network at the global level. If our model can generate a distribution of networks that have consistent network measures we may say it is a good model.
An observed graph which had a high likelihood of being replicated by a given set of ERGM parameters is said to be a "good fit". If it happens we may even hypothesize that the global features of the network may be explained by the modeled structural effects. It allows us to study the properties of the sampled networks in order to understand the nature of networks that are likely to emerge from these effects.

Goodreau et al. [30] and Hunter et al. [55] presented some tools for diagnosis of degeneracy in the model estimation implemented in the "statnet" package for R. Apart from the tools such as check.degeneracy function, mcmc.diagnostics function and graphical goodness of fit presented below, the authors suggested to analyze the initial and final value of coefficients and the improvements in the log-likelihood values, that should be relatively low numbers (e.g., less than 1 and generally decreasing). If a number approaches 20.0, it indicates problems with convergence.

- *Function check.degeneracy*

The *check.degeneracy* function is the control function of "ergm" estimation function [55]. This function gives as the output the *degeneracy.value* calculated to assess the degree of degeneracy in the model. If this value is equal to or larger than 1 indicates high instabilities and suggests that the model is degenerate. If *degeneracy.value* is equal to Inf (Infinity) diagnostics indicate that the model is very unstable and degenerate, and that the numerical summaries are suspected. The value smaller than 1 means the model is a good one.

- *Function mcmc.diagnostic*

Another function proposed in "statnet" package is the *mcmc.diagnostics* function, which examines the diagnostics for the MCMC model fitting process. This function creates simple diagnostic plots for the MCMC sampled statistics produced from a fit. The plots illustrate what is happening to the model statistics during the last iteration of the MCMC estimation procedure.

Figure 3.1 illustrates how chain behaves during the last iteration in the model which

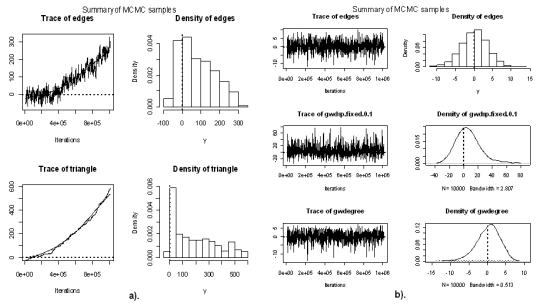

Figure 3.1: The plots created by the mcmc.diagnostics function

fit data well (Figure 3.1 b) and in the degenerate model (Figure 3.1 a). On the left of the diagnsotic plots the chain is represented as a time series for each model statistic, while the right-hand side summarizes this chain in a histogram. Both are normalized to the observed data, represented by 0. In a converged model, these statistics should vary stochastically around the mean, but will not trend steadily away from the mean. For each variable within each chain this function also prints the Raftery-Lewis diagnostics,

which is a run length control diagnostic based on a criterion of accuracy of estimation of the quantile q. It indicates about the number of iterations required to estimate the quantile q to within an accuracy of +/- r with probability p. It also indicates if runs are sufficient long and suggests which run length is required.

- Simulations and graphical goodness of fit

Another method for diagnosis of degeneracy is to evaluate if the model class is capable to produce networks that resemble the observed network. It can happen that although a maximum likelighood estimator $\hat{\theta}$, while providing in some sense the best model from the particular class of models for a particular choice of $u(y)$, does not necessarily result in a partocularly good model in a practical sense [55]. The method proposed by Hunter et al. [55] and Goodreau et al. [30] consists of simulating many networks (at least 100) from the final coefficient estimates and compare features of the observed network with the same features of a set of simulated networks 3.2. If the original network is inconsistent with the networks generated from the model, this suggests that the structure of the network differs from those predicted by the model, and the model is not fitted well. Figure 3.3 presents a way for comparing the observed network with generated

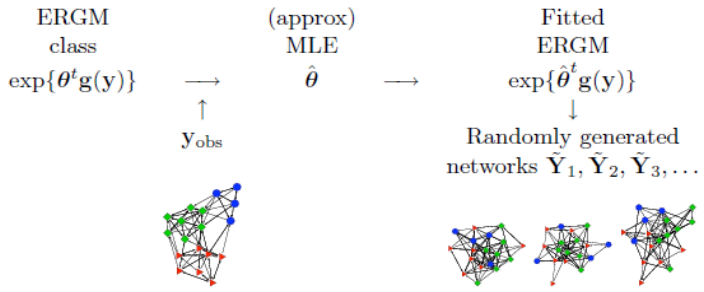

Figure 3.2: Principal idea of goodness of fit in ERGMs. Source: [54].

networks by plotting the full distribution of the statistic of interest represented by a histogram and the value of this statistic in the observed network represented by an arrow. Figure 3.3 a illustrates a model which is capable to generate networks with the number of edges comparable to that of the original network. The right-hand plot (Figure 3.3

b) illustrates a situation when the model is not able to riconstruct networks with the number of triangles of the observed network. For methods based on simulations we

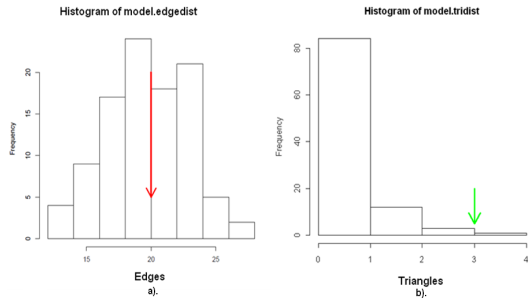

Figure 3.3: The number of edges (a) and the number of triangles (b) across 100 simulations compared to these statistics of the observed network (arrow)

have to be sure that we use sufficiently large interval between our samples, in order to minimize the autocorrelation between consecutive samples. It is important to state burnin (the number of steps in the simulation chain before the simulated network is drawn) at large number because it lets a chain moves from the starting network so that the output is approximately independent of initial conditions; otherwise we may obtain a plot that suggest that the model was well fitting, but in fact simply did not have enough time to move away from the original network.

Since the method presented above provides some limitations because it gives possibility to compare only a single outcome (i.e. number of edges) from the simulations, Hunter et al. [57] proposed graphical tests of goodness of fit (the *gof* function).
Idea of this method is based on comparison of certain observed network statistics difficult to be repressented with a single outcome (i.e. degree distribution) with the values of these statistics for a large number of networks (the default number is 100) simulated according to the fitted ERGM. The choice of the statistics determines which structural aspects of the networks are important in assessing fit. In practice there are considered three sets of statistics: the edgewise shared partner distribution, the geodesic distance distribution and the degree distribution.
The graphical goodness of fit depicts the boxplots with results across 100 networks

simulated from the fitted model. The vertical axes in plot can be explained as the relative frequency or its logit. The solid line represents the statistics for the observed network. The boxplots include the median and interquartile range. The light gray lines represent the range in which 95 percent of simulated observation fall. The gof function prints also the actual values with associated p-values. Figure 3.4 illustrates three different

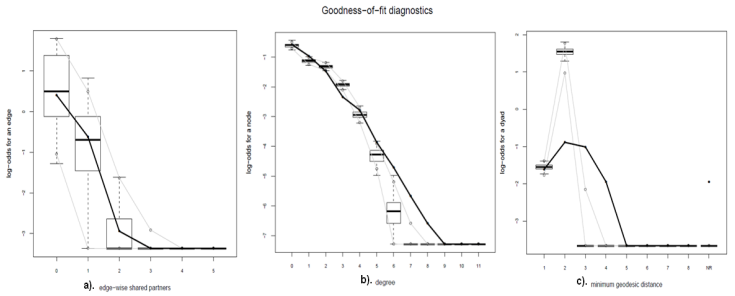

Figure 3.4: Goodness of fit diagnostics

situation; plot a) shows how well simulated network resamble the edgewise shared partner distribution of observed network; in the case b) simulated networks overestimate low degrees and underestimate high degrees of degree distribution of observed network. The plot c) presents that the network simulated according fitted model are not able to represent the geodesic distance distribution observed in the network.

Hunter et al. [57], using the graphical tool for goodness of fit, distinguished two situations for identifying if the model fits well the original network. First, if the observed network is not typical of the simulated network for a particular statistic included in the ERGM vector$u(y, X)$, then the model is either degenerate. Second, if it happens for the statistics not included in $u(y, X)$ the model is poorly-fitting.

With regard to the values of the approximate log-likelihood for the MLE and AIC it is worth noting that as the models become increasingly complex and include more dyadic dependence terms, the approximations for the likelihood become increasingly less precise [29]. Hunter et al.(2006) underlined that there are numerous problems with AIC index because it is based on the approximate loglikelihood values and the assumptions that justify the use of AIC for assessing goodness of fit are not appropriate

in the case of social networks because the observations are not an independent and identically distributed sample. If we consider a situation when we have to choose from two different models a better one, we will choose the model with maximized values of the log-likelihood function (or smaller value of AIC because lower AIC implying a significant increase in model fit). In the work of Hunter et al. [57], the authors compared their graphical methods for assessing goodness of fit with more traditional methods such as AIC or BIC. In this case the model selection via AIC and BIC is approximate at best and the goodness of fit plots provide a richer picture than AIC alone. The graphical goodness of fit procedures are confirmed in the sense that models that produce large reductions in (roughly approximate) AIC also seem to yield considerably better fits in the graphical plots. The graphical goodness of fit is more informative than the AIC results because the plots show which structural features are fit well and which are not.

CHAPTER 4

Evaluation of degeneracy diagnostic tools and parameter estimation methods

As it has already emerged from the previous chapters, estimation and interpretation are two critical points of research applications of statistical models in social networks analysis. After obtained parameter estimates, as well as estimates of the uncertainty of estimation, we may then take full advantage of having a statistical model for the network that is constructed from specific dependence assumptions and that is estimated from observed network data. According to the meaning of model parameters in describing relational phenomena in social networks, the obtained estimates should be useful for further investigation.

It is worth noting that the estimation step becomes complicated if the dependence structure is complex, as it probably needs to be for any realistic model. As discussed in detail in Section 3.1, only in the case of ERGMs with dyadic independent terms the estimation can be based on the standard logistic regression method which means that each run of these models exploits the same exact maximum likelihood calculation. In the case of dyadic dependence models we need to use stochastic algorithms based on Markov Chain Monte Carlo (MCMC) which provide the approximate maximum likelihood estimates and require random number generation. [55].

Apart from estimation, an important research goal is to evaluate whether the fitted model is useful for describing the structure of the network under examination or whether it is a degenerate model (3.4). It is quite complex to say which models are useful for a social network. A good statistical model should capture the significance of various kinds of local processes and still be able to reproduce the observed network at the global level. It allows us to study the global properties of the sampled networks in order to understand the nature of networks that are likely to emerge from the local effects. In the literature there have been proposed some degeneracy and model diagnostic tools which help to choose the model which fits the data well (see Section 3.5).
Considering the stochastic nature of the estimation procedures and the importance of the diagnostic tools in the phase of evaluation of the fitted model, two research questions have emerged that we would like to analyze:

- **Results of diagnostic tools**: Since the estimation algorithms do not provide exactly the same results each time the algorithm is run, we are interested in evaluating the information provided by degeneracy diagnostic tools implemented in "statnet" for different runs of the same model. We assess if for different runs the results of diagnostic tools are different and if it depends on network size.
- **Results of estimation procedures**: We compare the results from three estimation procedures such as Newton-Raphson, Robbins-Monro and Stepping for networks of different sizes to evaluate which of these methods exhibits the best accuracy and gives the best approximation.

4.1 Study design

As stated above there are two important issues under examination which divide the study into two parts which in some way are closely related to each other. The first one deals with the assessment of the degeneracy diagnostic tools and the second one focuses on comparison of three estimation methods based on MCMC technique which provide the approximation of MLE. To our aims we focus on the "ergm" package which is one of the "statnet" packages implemented in R [33] [36].

- Network data

The study is performed using both, simulated and realistic networks. We use three real network data such as the FLOMARRIAGE network, the ECOLI2 network and the FAUX.MAGNOLIA.HIGH network. These networks are available in "ergm" package (Figure 4.1). Table 4.1 illustrates principal statistics observed in the networks chosen for the analysis.

To simulate networks we will use one of the examples presented in the work of Robins et al. [85] where the authors clearly illustrate how Markov graph distributions based on a locally specified social process can be used to produce small world graphs through a judicious choice of parameter values.

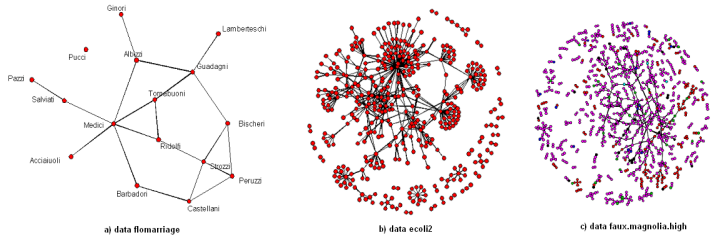

Figure 4.1: Networks used for the analysis a) FLOMARRIAGE, b) ECOLI2 and c) FAUX.MAGNOLIA.HIGH

	FLOMARRIAGE	ECOLI2	FAUX.MAGNOLIA.HIGH
nodes	16	418	1461
edges	20	519	974
density	0.167	0.06	0.0009
2 stars	47	5290	1821
triangles	3	42	169
clust. coeff.	0.2784	0.0238	0.0306

Table 4.1: Graph statistics for real network data

- The data FLOMARRIAGE represents a data set analyzed by Padgett [74], consisting of weddings among leading Renaissance Florentine families. It is an undirected network with 16 nodes (families) and 20 edges (marital ties).

- The data ECOLI2 is the undirected version of a biological network in which the nodes are operons in Escherichia Coli and an edge from one node to another indicates a regulating relationship between them. The data set is based on the RegulonDB network [88] and was modified by Shen-Orr et al. [90]. It is a network with 418 nodes, 519 edges.

- The network FAUX.MAGNOLIA.HIGH represents simulations of an in-school friendship network with 1461 nodes (students), 974 undirected edges (mutual friendship) [79]. There are three different attributes associated with the nodes in this network Race, Sex and Grade.

- As simulated networks, we consider four networks of different sizes: Network 1 with 100 nodes, Network 2 with 300, Network 3 with 500 and Network4 with 1000 nodes. All of them are generated from the Markov random graph distribution based on the vector of parameter values (-4.0, 0.1, -0.05, 1.0), where the values are for the number of edges, the number of 2-stars, the number of 3-stars, and the number of triangles, respectively. Table 4.2 presents principal characteristics from simulated networks. We observe that densities and clustering coefficients decrease with the increase the number of nodes for the Markov random graph with these specific parameter values. It is worth noting how slowly the number of triangles increases in comparison to the increase of number of 2-stars and 3-stars in respective networks.

	Network1	**Network2**	**Network3**	**Network4**
nodes	100	300	500	1000
density	0.0285	0.0174	0.0125	0.0075
2 stars	411	3822	9008	25825
3 stars	409	5751	15914	53769
triangles	11	39	77	158
clust. coeff.	0.0803	0.0306	0.0256	0.0184

Table 4.2: Graph statistics from simulated networks

- Technical considerations

The main aim of the first part of the analysis is to evaluate practical methods for diagnosis of degeneracy. We focus on the four degeneracy diagnostic tools in R such as the *check.degeneracy* function, the *mcmc.diagnostics* function, simulations method, graphical goodness of fit (the *gof* function) (3.5). We propose to evaluate information provided for different runs of the same model. More details and results are presented in Section 4.2.

The general strategy is as follows:

For FLOMARRIAGE and FAUX.MAGNOLA.HIGH networks we obtain 500 MLE estimates based on the Newton-Raphson algorithm a large number of times. We analyze the distribution of values provided by the *check.degeneracy* function and on the basis of these values we choose seven sets of estimates obtained from different runs which will be used for further evaluation. Then for each of the chosen configurations of estimates, we evaluate the information provided by the other diagnostic tools to assess whether for different runs of the model the results of diagnostic tools for degeneracy are different and whether they depend on network size.

The main interest of the second analysis is to compare the performance of the maximumlikelihood (MLE) approximation techniques such as Newton-Raphson, Robbins-Monro, Stepping.

We propose the following strategy:

We obtain 1000 the MLE estimates basing on the procedures under comparison to obtain the distribution of estimates for each parameter. The first step is to investigate the simple Bernoulli model for both, simulated and real networks. In this case we evaluate the accuracy of the estimation procedures in approximating the true value of MLE which is equal to the MPLE.

The next step is to examine for simulated networks the performance of the procedures in fitting the model used for network simulation. In this case the values used for network simulation are treated as the true values of parameters. In the case when the true value of parameter is known, we evaluate accuracy and precision by computing: (1) the bias of the estimates from the true values of parameters, (2) the sample variance of

the estimates and (3) the mean squared errors (MSE) of the estimates. In the case of complex models for real networks neither the true value of MLE nor the process which creates the network is known. In this case we examine the variability of estimates provided by the procedures under comparison and then we try to assess the capacity of estimates in producing networks which resemble the observed networks. To do it we use the expected values of estimate distributions as arguments for simulating a graph distribution. We compare the generated networks to the observed graphs and decide if the fitted model is a good one for the data.
Details about particular models, values of MCMC arguments and, finally, the results are presented in Section 4.3.

One of the important aspects of analysis is to notice of some fundamental considerations such as estimation procedures, values of θ_0, MCMC arguments.

- For each network we investigate different model because as stated in Section 2.1.3 the effective choice of terms for an ERGM depends on the context and on the structure of the observed network.
- For the goal of this analysis we consider three estimation procedures. The primary method is a simple Newton-Raphson algorithm of maximum likelihood estimation. Alternative methods are a stochastic approximation Robbins-Monro and a partial Stepping algorithm. All three procedures have been described in Section 3.2.
- As stated in Section 3.2 the stochastic estimation techniques requires one to select an initial parameter value θ_0. In practice, to obtain the convergence of the algorithm the value θ_0 has to be "close enough" to the true maximum likelihood estimator. For goals of this analysis we chose the MPLE values obtained through logistic regression - the usual method for fixing θ_0 - as initial values of estimation procedures.
- For each model the estimation procedure is repeated a large number of times. For each run we change only the value of seed by fixing the random number generator

to obtain exact reproducibility of estimates and to be sure that the same run will not be repeated.

- The MCMC-based estimation methods require one to control initialization values of a chain arguments such as a burnin (denoted `burnin`), an MCMC sample size (denoted `MCMCsamplesize`), a number of iterations (denoted `maxit`), an interval (denoted `interval`) and a length of steps (denoted `steplength`). They determine:
  - `burnin`: the number of initial samples discarded to avoid any possible bias of the original network,
  - `MCMCsamplesize`: the total number of samples taken,
  - `maxit`: the number of times the parameter for the MCMC should be updated by maximizing the MCMC likelihood,
  - `interval`: the number of Markov Chain steps between successive samples in order to minimize the autocorrelation between consecutive samples,
  - `steplength`: the argument which modifies the NewtonRaphson optimization of the Monte Carlo approximation to the loglikelihood to account for the uncertainty in the approximation to the actual log-likelihood.

  The values of MCMC arguments will vary from application to application to obtain the chains sufficiently long for models fitted for the networks.
- For purposes of this analysis for each run we control estimates, standard errors, estimates of the error attributable to the MCMC algorithm (MCMC standard errors), the value of maximum likelihood, the value of *check.degeneracy* function and the value of AIC.

The R code of the function used for this analysis is given in Appendix A.

4.2 Contribution to analysis of degeneracy diagnostic tools

For this analysis we choose two real networks, a small one FLOMARRIAGE and a large one FAUX.MAGNOLIA.HIGH. We focus on the four degeneracy diagnostic tools in R such as the *check.degeneracy* function, the *mcmc.diagnostics* function, simulations method, graphical goodness of fit (the *gof* function), which make it possible to understand what is happening in the chains during estimation procedure and provide good information about which structural features are fitted well and which are not. The main interest is to analyze whether for different runs of the model the results of diagnostic tools for degeneracy are different and whether they depend on network size [100].

To avoid the problem of degeneracy we decided to consider recently developed statistics described in Section 2.2.4. For the network FLOMARRIAGE we focus on a model with statistics such as:

- the number of edges (denoted “edges”),
- the geometrically weighted dyadwise shared partner with value of ϕ equal 0.5 (denoted “gwdsp(0.5,fixed=TRUE)”),
- the geometrically weighted degree distribution with value of ϕ equal 1 (denoted “gwdegree (1,fixed=TRUE)”).

For the network FAUX.MAGNOLIA.HIGH we focus on the model with the seven structural parameters, related to network statistics such as:

- the number of edges (denoted “edges”),
- the geometrically weighted edgwise shared partner with value of ϕ equal 0.1 (denoted “gwesp(0.1,fixed=TRUE)”),
- the geometrically weighted dyadwise shared partner with value of ϕ equal 0.1 (denoted “gwdsp(0.1,fixed=TRUE)”),
- the geometrically weighted degree distribution with value of ϕ equal 0.1 (denoted “gwdegree (0.1,fixed=TRUE)”).

To represent the nodal attribute effect, we use uniform homophily terms such as "nodematch" which adds one network statistic to the model, which counts the number of edges (i, j) for which nodes i and j has the same attribute name [69]:

- "nodematch("Grade", diff=FALSE)",
- "nodematch("Race", diff=FALSE)",
- "nodematch("Sex", diff=FALSE)").

This analysis is carried out on the version 2.2-4 of the ergm package (22 April 2010). For each network we repeat 500 times the estimation procedure for two models with the same terms but different values of MCMC arguments such as number of iterations (`maxit`), `burnin`, `MCMCsamplesize` and `steplength` (Table 4.3). To monitor the statistical properties of the MCMC algorithm, we use the function *mcmc.diagnostics* which gives the possibility to control the performance of the MCMC algorithms graphically and numerically and provides both, plots and some numerical diagnostics which indicate if the chain is sufficiently long to converge (see Section 3.5).

	Model 1.1	**Model 1.2**	**Model 2.1**	**Model 2.2**
Burnin	1e+4	1e+5	1e+4	1e+5
MCMCsample	1e+4	1e+5	1e+4	1e+5
Steplength	0.5	0.25	0.5	0.25
Interval	100	100	100	100
maxit	3	10	3	15

Table 4.3: Values of MCMC procedure arguments

In models 1.1 and 2.1 we focus on default argument values proposed in the "ergm" estimation function. Since for MCMC-based estimation methods it is important to obtain a chain sufficiently long [30] we decide to change values of some MCMC arguments. In the models 1.2 and 2.2 we increase the number of iterations from 3 to 10 for model for FLOMARRIAGE, network and from 3 to 15 for model for FAUX.MAGNOLIA.HIGH. We also increase the value of `burnin` from 1e+4 to 1e+5 which means that we discard more initial samples to avoid any possible bias of the original network. We also increase `MCMCsamplesize` from 1e+4 to 1e+5 to obtain bigger sample size. We decrease the value of `steplength` from 0.5 to 0.25 to make fitting more stable. In the models 1.1

and 2.1 chains have 1.01 million steps (10,000 + 10,000100) and in the models 1.2 and 2.2 their length is of 10.1 million steps (100,000 + 100,000100). While analyzing outputs of control diagnostic we notice that both, the chain in the model 1.1 and the chain in the model 1.2, are sufficiently long to obtain estimates, but since the second one is longer and should ensure more precision we choose this one for further analysis. For models 2.1 and 2.2 the second one demonstrates the more adequate length so this one is chosen for further analysis.

It is worth noting that for with the FLOMARRIAGE network for different runs the estimates obtained from the Newton-Raphson procedure presented values very different from one another (more than two standard errors). Indeed, for the results for FAUX.MAGNOLIA.HIGH we notice better stability.

Table 4.4 and Table 4.5 present same principal statistics such as Minimum and Maximum values, 1st and 3rd Quartiles, Median and Mean for distributions of estimates and degeneracy.value obtained for 500 runs of estimating procedures for the model 1.2 and the model 2.2, respectively.

	Min.	**1st Qu.**	**Median**	**Mean**	**3rd Qu.**	**Max**
edges	-6.482	-4.134	-3.768	-3.788	-3.440	-1.690
gwdsp	-0.2806	0.2889	0.3588	0.3664	0.4410	1.0990
gwdegree	1.062	1.378	1.473	1.470	1.556	1.778
deg.val	0.9297	2.6490	4.0640	5.3150	6.6760	Inf

Table 4.4: Principal statistics for distributions of estimates and *degeneracy.value*. Model 1.2.

	Min.	**1st Qu.**	**Median**	**Mean**	**3rd Qu.**	**Max**
edges	-8.791	-8.428	-8.339	-8.335	-8.247	-7.950
nodematch.Grade	2.476	2.752	2.831	2.827	2.897	3.153
nodematch.Race	0.7113	0.9312	0.9938	0.9928	1.0590	1.2530
nodematch.Sex”	0.5378	0.7399	0.7995	0.7986	0.8577	1.0480
gwesp	1.523	1.647	1.694	1.693	1.734	1.903
gwdsp	-0.3473	-0.3367	-0.3343	-0.3343	-0.3320	-0.3232
gwdegree	-0.9821	-0.9362	-0.9280	-0.9284	-0.9200	-0.8964
deg.val	0.9323	4.5910	6.9400	7.5680	9.8310	Inf

Table 4.5: Principal statistics for distributions of estimates and *degeneracy.value*. Model 2.2.

Below we present the detailed analysis for seven runs chosen from 500 runs of the

model 1.2 and the model 2.2. We try to answer the question whether the diagnostic tools provide diverse results for different runs. We try to examine if the use of diagnostic tools may help to choose the run with estimates that fit the data well in the case when estimates differ from each other for different runs of the model.

	Seed 274	Seed 324	Seed 76	Seed 140	Seed 433	Seed 143	Seed 402
deg.val	Inf	17	6.3	3.2	2.3	1.4	0.92
edges	-2.007 (0.49)	-4.88 (0.46)	-4.7954 (0.75)	-3.477 (0.52)	-3.698 (0.45)	-3.93 (0.51)	-3.87 (0.5)
gwdsp	0.279 (0.07)	0.794 (0.1)	0.6596 (0.19)	0.3288 (0.12)	0.3562 (0.09)	0.3489 (0.012)	0.3922 (0.11)
gwdegree	1.393 (0.15)	1.472 (0.13)	1.369 (0.15)	1.4456 (0.13)	1.4145 (0.18)	1.3442 (0.12)	1.4303 (0.13)
mle.lik	-34.45	-40.5	-49.93	-51.35	-50.53	-52.01	-51.68
AIC	74.9	87	105.86	108.7	107.06	110.03	109.36

Table 4.6: Results for fits chosen from 500 runs of model 1.2

The first function which we use is the *check.degeneracy* function. For model 1.2 11.4% of runs present *degeneracy.value* equal Inf which means the model is very unstable and degenerate. Only about 1% of *degeneracy.values* are smaller than 1. For model 2.2 2% of runs presented *degeneracy.value* equal to Inf which means the model is very unstable and degenerate. Only 0.4% of runs presented *degeneracy.value* smaller than1. Figure 4.2 illustrates the distributions of *degeneracy.value* without values equal Inf which resulted for fits of model 1.2 and model 2.2. In the case of the model 1.2 the distribution dimonstrates a pronounced right skew, the mean is equal to 5.31 and 50% of runs result with *degeneracy.value* higher than 2.65 and smaller than 6.68. For the model 2.2 we notice the mean equal to 7.56 and 50% of runs with *degeneracy.value* higher than 4.59 and smaller than 9.83. For the FLOMARRIAGE network, analyzing diagnostic plots obtained from the *mcmc.diagnostic* function and presented in Figure 4.3, we notice that chains of runs with small *degeneracy.value* (seed=143, seed=402) move stochastically around 0 which represents the value of the statistic in the observed network. Runs with higher *degeneracy.value* (seed=274, seed=324, seed=65) illustrate

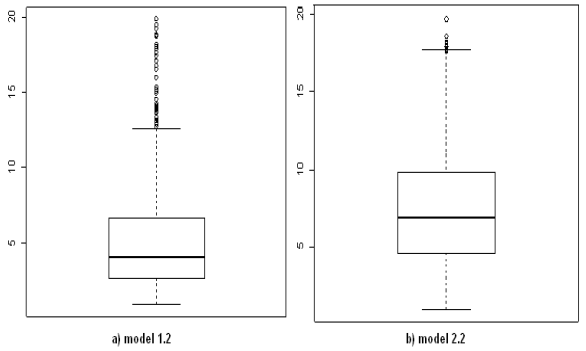

Figure 4.2: Boxplot of the degeneracy.value distribution for runs of model 1.2 and model 2.2.

that the process is very unstable and networks generated during the last iteration of the MCMC estimation procedure overestimate or underestimate statistics included in the model 1.2 in comparison to the observed network. Figure 4.4 and Figure 4.5 illustrate the number of edges and the number of triangles, respectively, across 100 simulations from model 1.2 compared to the number of edges and the number of triangles of the observed network (red arrow and green arrow, respectively). It is worth noting that for the examples chosen for analysis the runs with *degeneracy.value* equal to 3.2 or smaller are able to generate large number of networks that reproduce the number of triangles of the observed network. With regard to the number of edges we observe that the runs with *degeneracy.value* 1.4 and 0.92, in comparison to the other fits, generate more networks which resemble the number of edges of the observed network. The run with *degeneracy.value* equal to Inf is not able to reproduce either number of edges or the number of triangles of the observed network.

Figure 4.6 and Figure 4.7 illustrate the goodness of fit for distribution of edgewise shared partners and minimum geodesic distance distribution, respectively. We note that fits with Inf or high values of *degeneracy.value* are not able to reproduce the edgewise shared partner distribution and the geodesic distance distribution in an appropriate way. Runs with *degeneracy.value* equal to 3.2 or smaller are able to generate networks which

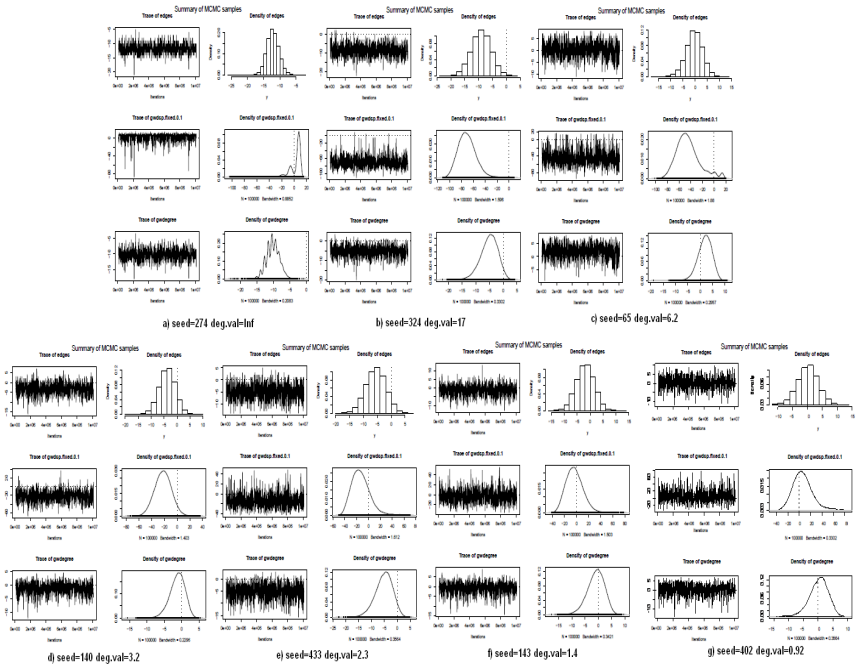

Figure 4.3: Diagnostic plots for the MCMC sampled statistics produced from fits with different seed and different degeneracy.value. Model 1.2

present the edgewise shared partner distribution and the geodesic distance distribution similar to the observed networks. In the case of FLOMARRIAGE dataset the runs with high *degeneracy.value* present larger values of approximate log-likelihood (or smaller values of AIC) in comparison to the runs with small values of *degeneracy.value*. We notice that run with *degeneracy.value* equal to Inf presents the most improved value of loglikelihood equal to -34.46 which, in comparison to -52.01 and -51.68 from the fit with *degeneracy.value* equal 1.4 and 0.92, respectively, shows an important difference. It might be a sign of the situation that estimates of fit with *degeneracy.value* equal Inf maximize loglikelihood function but the observed network may not be particularly likely relative to other networks.

In the case of the large network FAUX.MAGNOLIA.HIGH, we notice that for fits

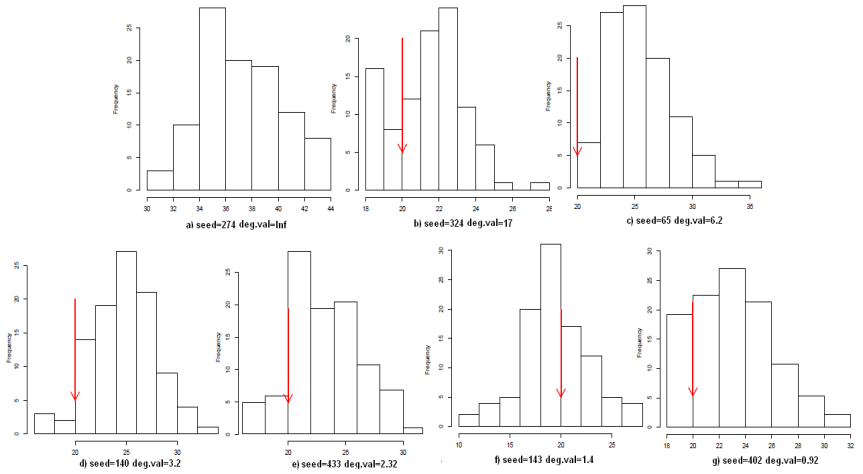

Figure 4.4: The number of edges across 100 simulations from different runs of model 1.2 compared to the number of edges in the observed network (red arrow).

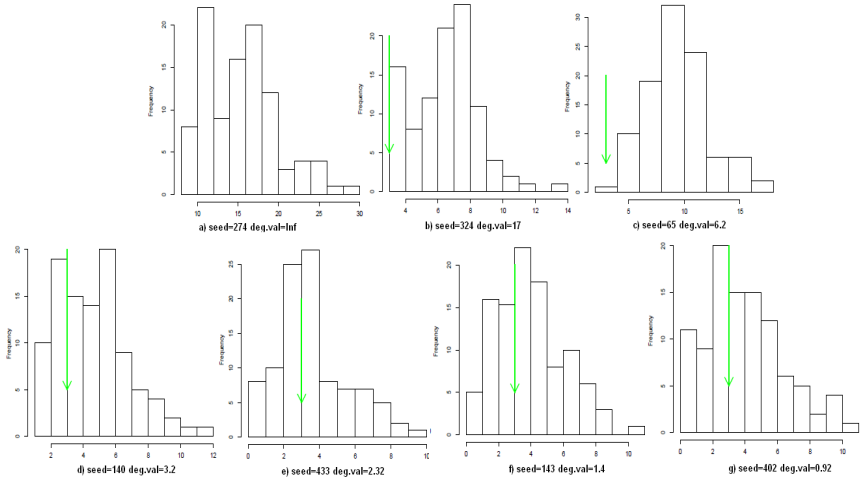

Figure 4.5: The number of triangles across 100 simulations from different runs of model 1.2 compared to the number of triangles in the observed network (green arrow).

with different seeds for two models (model 2.1 and model 2.2) estimates do not differ so strongly as in the case of the FLOMARRIAGE network. (Table 4.5 and Table 4.7)

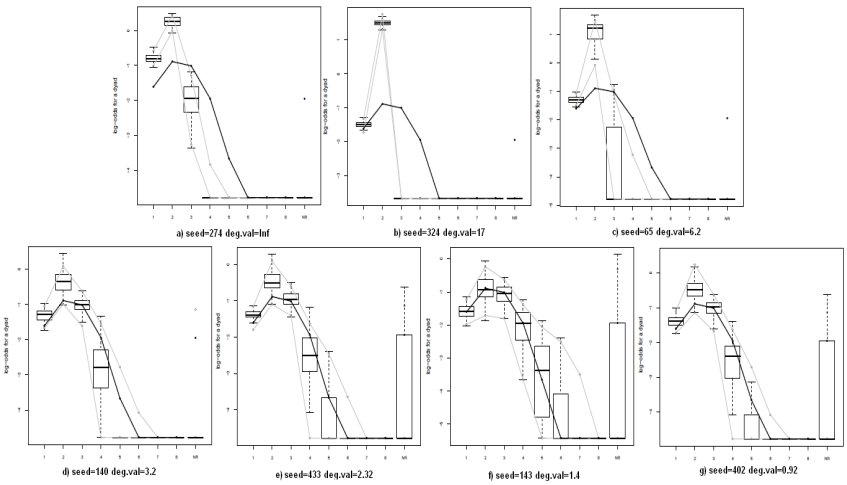

Figure 4.6: Graphical goodness of fit for distribution of edgewise shared partners for fits with different degeneracy.value (model 1.2)

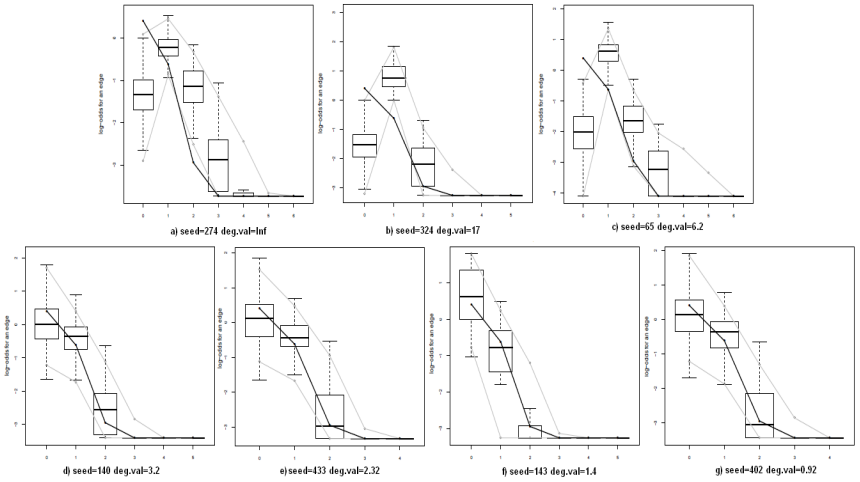

Figure 4.7: Graphical goodness of fit for minimum geodesic distance distribution for fits with different degeneracy.value (model 1.2)

With regard to the *mcmc.diagnostics* plots for FAUX.MAGNOLIA.HIGH network (Figure 4.8) we notice that only chains of fits with very small *degeneracy.value* (seed=403,

	Seed 429	Seed 21	Seed 249	Seed 53	Seed 49	Seed 403	Seed 7
deg.val	Inf	18	5.9	3.5	2.8	1.8	0.7
edges	-8.137 (0.1)	-8.288 (0.103)	-8.124 (0.096)	-8.595 (0.108)	-8.256 (0.102)	-8.315 (0.11)	-8.35 (0.107)
nodemG	2.782 (0.086)	2.69 (0.086)	2.739 (0.076)	2.904 (0.089)	2.748 (0.085)	2.748 (0.089)	2.849 (0.092)
nodemR	0.745 (0.072)	1.078 (0.072)	0.903 (0.07)	1.178 (0.078)	0.883 (0.069)	1.049 (0.076)	1.04 (0.066)
nodemS	0.826 (0.06)	0.725 (0.061)	0.723 (0.061)	0.789 (0.065)	0.854 (0.064)	0.829 (0.0.69)	0.799 (0.072)
gwesp	1.66 (0.051)	1.744 (0.045)	1.743 (0.0047)	1.75 (0.046)	1.779 (0.043)	1.739 (0.054)	1.675 (0.044)
gwdsp	-0.337 (0.004)	-0.338 (0.003)	-0.333 (0.004)	-0.337 (0.004)	-0.336 (0.004)	-0.336 (0.005)	-0.332 (0.005)
gwdegree	-0.916 (0.019)	-0.93 (0.019)	-0.928 (0.02)	-0.93 (0.018)	-0.917 (0.019)	-0.916 (0.02)	-0.933 (0.018)
mle.lik	-5438.5	-5442.7	-5456.1	-5457.1	-5458.2	-5459.7	-5460.4
AIC	10891	10899	10926	10928	10930	10933	10935

Table 4.7: Results for fits chosen from 500 runs of model 2.2

seed=7) move stochastically around the mean (observed network). For the remaining fits we observe that networks generated during the last iteration of the MCMC estimation procedure in comparison to the observed network (represented by 0) overestimate statistics included in the model 2.2. With regard to the results of simulations and graphical goodness of fit we notice that although fits have different *degeneracy.value* they manage to reproduce distributions of respective statistics equally well. For the FAUX.MAGNOLIA.HIGH network we notice that all runs are able to produce well networks with the number of edges similar to the observed network (Figure 4.9). None of the chosen runs is able to generate networks with the number of triangles of the observed network, generally the generated networks underestimated this network feature. With regard to the graphical goodness of fit we notice that the runs chosen for this analysis reproduced distributions of respective statistics in the same way. As an example, we present Figure 4.10 which illustrates the degree distribution. We can observe that the chosen runs with different *degeneracy.value* produce networks which equally overesti-

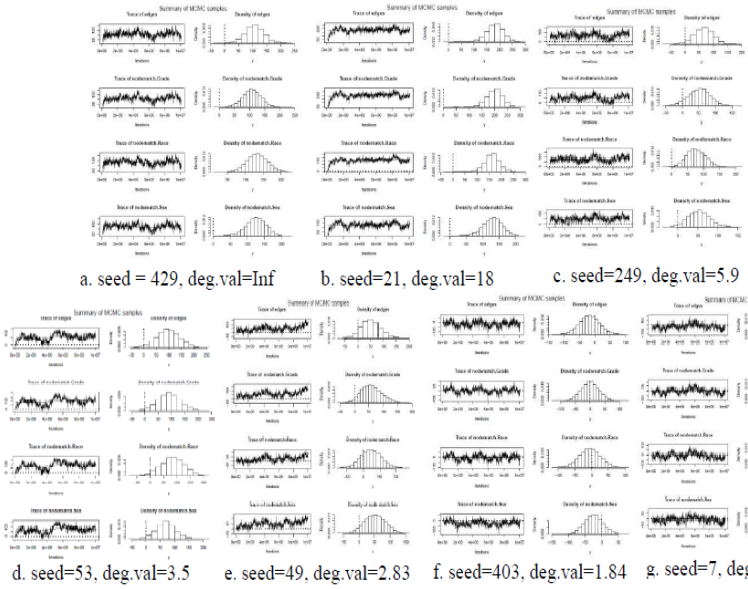

Figure 4.8: Diagnostic plots for the MCMC sampled statistics produced from fits with different seed and different degeneracy.value. Model 2.2

mate low degrees and underestimate high degrees of the original network.

Generally, while analyzing results for the small network we observe that runs with small *degeneracy.value*, with regard to all diagnostic tools, show ability to reproduce the observed networks. Runs with large values of *degeneracy.value* are not able to reproduce the observed network. Since in this case the fits with different seed provide estimates which are quite different one from each other, the use of all diagnostic tools may be very helpful to choose good parameter estimates and the *check.degeneracy* function may be recommended as a first selection criterion.

In the case of the FAUX.MAGNOLIA.HIGH network, we notice that estimates do not differ for runs with different seeds so strongly as in the case of the FLOMARRIAGE network. In this case we observe that the importance of the *check.degeneracy* in choosing good parameter estimates is not so big as in the case of the small network. Outputs of the *mcmc.diagnostics* show that fits with higher *degeneracy.value* present higher

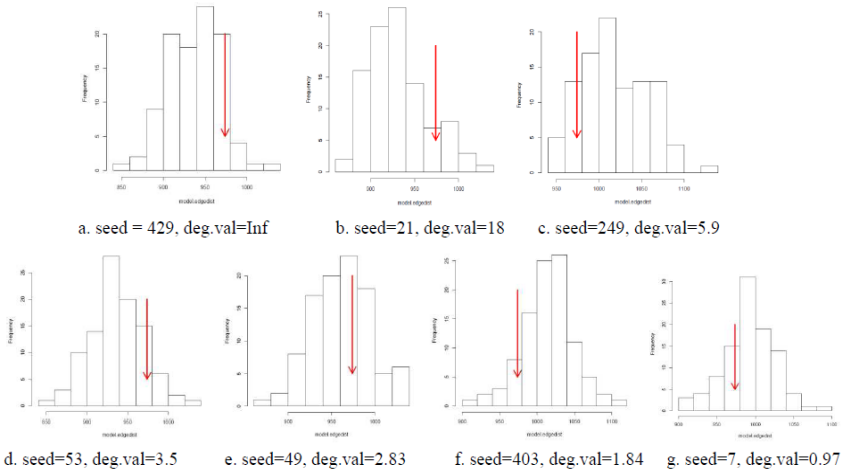

Figure 4.9: The number of edges across 100 simulations from different runs of model 2.2 compared to the number of edges in the observed network (red arrow).

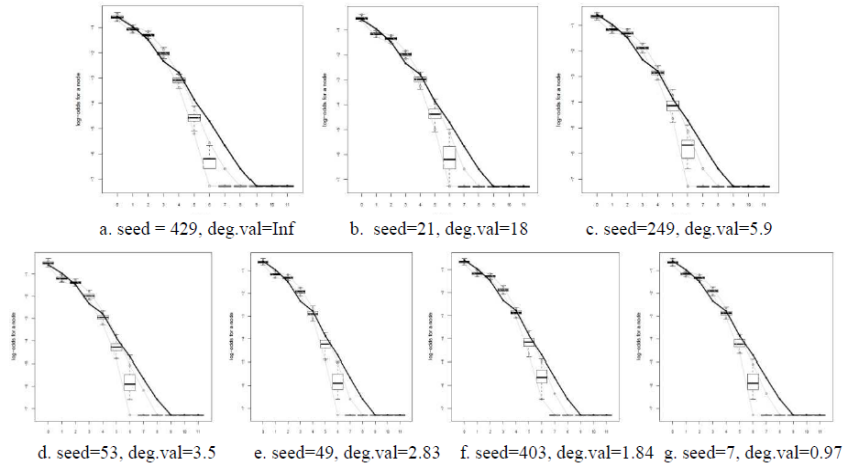

Figure 4.10: Graphical goodness of fit for degree distribution for fits with different degeneracy.value (model 2.2)

instability of chains. With regard to the results of simulations and graphical goodness of fit we noticed that although fits have different *degeneracy.value* they manage to

reproduce distributions of respective statistics equally well. We attribute this to the fact that for different runs estimates do not differ so much one from the other.

Results of this analysis show that although we use new specification statistics which help to avoid problem of degeneracy it is important to evaluate information from different degeneracy diagnostic tools and to control degeneracy and instability not only of different models but also of different runs of the same model. The tools analyzed above give the possibility to understand what is happening in the chains during estimation procedure and provide good information about which structural features are fit well and which are not.

4.3 Comparison of Newton-Raphson, Robbins-Monro, Stepping estimation procedures

The main interest of this part of the analysis is the comparison of the three different algorithms for approximation of the maximum likelihood estimator available in ergm package for R, such as the Newton-Raphson, the Robbins-Monro, the Stepping procedures. A simple Newton-Raphson algorithm is used as default by "ergm" function which fits exponential-family random graph models. Alternative methods of maximum likelihood estimation is a form of stochastic approximation called Robbins-Monro, chosen with the *control.ergm (style = "Robbins-Monro")* option and a partial stepping algorithm, chosen with the *control.ergm (style = "Stepping")* option. In this analysis we will focus on the version of the ergm package (the version 2.2-5 from 24 July 2010). Networks chosen for analysis are four networks generated from the Markov random graph distribution based on the vector of parameter values (-4.0, 0.1, -0.05, 1.0) and three real networks such as: the FLOMARRIAGE network, the ECOLI2 network and the FAUX.MAGNOLIA.HIGH network (see Section 4.1).

Firstly we consider a simple independent dyadic model with one term, the number of edges (denoted "edges") for four simulated networks. Model fittings for all the

networks are based on default MCMC argument values proposed in "ergm" function such as `burnin` equal to 1e+4, `MCMCsamplesize` equal to 1e+4 and `interval` equal to 100.

Since in the case of the Bernoulli model the value of maximum likelihood estimator is equal to the MPLE and can be found using the standard logistic regression, to obtain approximations we force all the three methods to use MCMC maximum likelihood. Because as stated before we use θ_0 equal the value of MPLE as initial parameter value, in this situation we are sure that θ_0 is "really close" to the true value of the maximum likelihood estimator and we are able to evaluate which algorithm provide more precise approximations.

Figure 4.11 depicts boxplots of distributions of the maximum likelihood estimates of the "edges" terms for 1000 runs with different seeds. Each boxplot is labeled according to the estimation procedure: Newton-Raphson, Robbins-Monro, Stepping. Each subfigure corresponds to the results of the network of different sizes: a) 100 nodes, b) 300 nodes, c) 500 nodes and d) 1000 nodes. The horizontal line through each section represents the true parameter value. The boxplots give the quartiles and tails of each distribution, the dots correspond to the means of those distributions. The bias of each expected value of the estimates from the true value of parameter and the mean squared error of the estimates is presented in Table 4.8. It is worth noting that for all three algorithms the absolute values of bias are smaller than 0.008. We notice that the bias for the Newton-Raphson for all four networks exhibits a negative sign and for the Stepping procedure it has a positive sign. Since all the procedures demonstrate the bias very close to 0 so MSE is equal to the variance of estimates.

	Network 1	**Network 2**	**Network 3**	**Network 4**
Newton-Raphson	(1) -0.00114 (2) 0.00065	(1) -0.00056 (2) 0.00015	(1) -0.00017 (2) 8.28e-05	(1) -0.00009 (2) 4.89e-05
Robbins-Monro	(1) 0.00008 (2) 2.704e-05	(1) -0.00182 (2) 0.00039	(1) -0.0077 (2) 0.00179	(1) -0.00027 (2) 0.00028
Stepping	(1) 0.00025 (2) 3.24e-06	(1) 0.00035 (2) 4.41e-06	(1) 0.00044 (2) 7.84e-06	(1) 0.0005 (2) 1.156e-05

Table 4.8: The bias of the expected value of the estimates from the true value of parameter (1) and the mean squared errors of the estimates (2). The Bernoulli model for simulated networks.

For the networks with 300, 500 and 1000 nodes the approximations of MLE obtained from the Robbins-Monro algorithm present the biggest values of MSE so also variability. The Stepping algorithm presents the smallest variability from all the three procedures. For the Robbins-Monro procedure we notice that the distributions of estimates present the increase of the number of outlayers with increasing number of nodes. After this preliminary examination we consider the Stepping procedure the most accurate. The

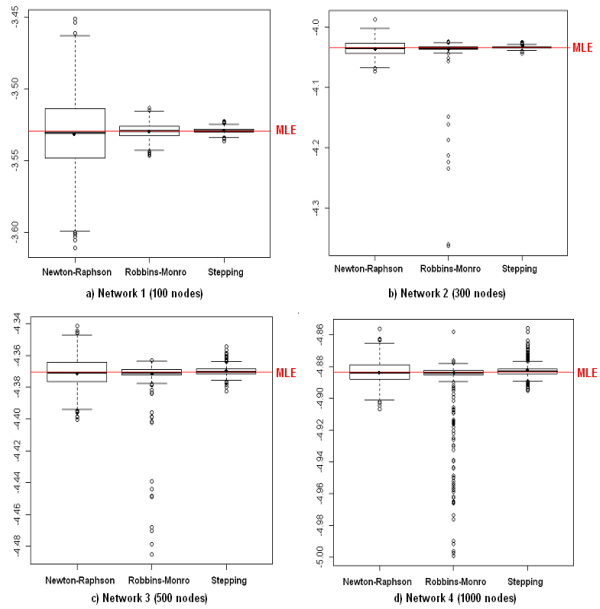

Figure 4.11: Boxplot of distribution of the approximations of MLE of the "edges" term for 1000 runs (with different seeds) of three different estimation procedures for networks simulated from Markov graph distribution with parameters (-4,0.1,-0.05,1). Line illustrates the true value of the parameter. The dots correspond to the means of the distributions.

next step is to examine results provided by the estimation procedures for a more complex model for simulated networks. In this case we fit the Markov model with the terms used for network generation so the true value of parameters are also known. As stated before (Section 2.2.3), the Markov model is considered not to be very useful in describing real social phenomenon because of the problem of degeneracy. However since the simulated networks have been generated from this model we can expect that it

will be able to describe the structure of them and the problem of degeneracy will not occur. The Markov model is more complex and requires different values of MCMC argument for networks of different sizes (Table 4.9).

	MCMCsamplesize	Burnin	Interval
Network 1	2e+4	1e+4	1000
Network 2	2e+4	1e+4	1000
Network 3	2e+4	3e+4	1000
Network 4	2e+4	1e+5	1000

Table 4.9: Values of MCMC arguments for fits from Markov model for simulated networks

Figures 4.12, 4.13, 4.14 and 4.15 depict boxplots of distributions of the approximations of the MLE for parameters of the number of edges, the number of 2-stars, the number of 3-stars, and the number of triangles terms, respectively.

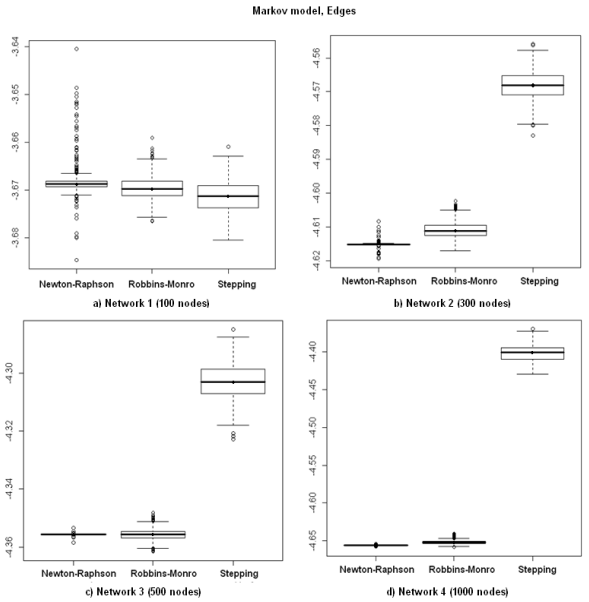

Figure 4.12: Boxplot of distribution of the approximations of the MLE of the "edges" term of Markov model for 1000 runs (with different seeds) of three different estimation procedures. The dots correspond to the means of the distributions. The true value of the parameter is equal to -4.

	Network 1	Network 2	Network 3	Network 4
Newton-Raphson	(1) 0.33147 (2) 6.4e-06 (3) 0.10988	(1) -0.61511 (2) 1.85e-07 (3) 0.37836	(1) -0.356 (2) 2.25e-08 (3) 0.12644	(1) -0.656 (2) 9.76e-09 (3) 0.43014
Robbins-Monro	(1) 0.33037 (2) 5.71e-06 (3) 0.10915	(1) -0.611 (2) 5.48e-06 (3) 0.37333	(1) -0.356 (2) 3.39e-06 (3) 0.12651	(1) -0.652 (2) 6.1e-06 (3) 0.42547
Stepping	(1) 0.32855 (2) 1.79e-05 (3) 0.10799	(1) -0.568 (2) 1.78e-05 (3) 0.32326	(1) -0.303 (2) 3.86e-05 (3) 0.09183	(1) -0.401 (2) 0.00011 (3) 0.16118

Table 4.10: The bias of the expected value of the estimates from the true value of parameter (1), the sample variance of the estimates (2) and the mean squared errors of the estimates (3). "Edges" term of the Markov model for simulated networks.

The approximations of MLE for "edges" term obtained from the Stepping algorithm exhibit the highest variability but it is worth noting that in comparison to the estimates provided by other two methods, these results demonstrate the smallest bias from the true value of parameter. It exhibits also the smallest values of MSE. The results of Newton-Raphson present the smallest variability for networks with 300, 500 and 1000 nodes and with the increasing number of nodes they present less outlayers and become less variation. The Newton-Raphson and the Robbins-Monro demonstrate very similar values of mean squared errors.

	Network 1	Network 2	Network 3	Network 4
Newton-Raphson	(1) -0.08798 (2) 7.83e-05 (3) 0.00782	(1) 0.11576 (2) 3.8e-06 (3) 0.0134	(1) 0.0427 (2) 9.98e-07 (3) 0.00182	(1) 0.1034 (2) 5.63e-07 (3) 0.0107
Robbins-Monro	(1) -0.09167 (2) 4e-08 (3) 0.0084	(1) 0.11561 (2) 1.44e-08 (3) 0.01337	(1) 0.0426 (2) 3.2e-09 (3) 0.00181	(1) 0.1034 (2) 5.76e-09 (3) 0.0107
Stepping	(1) -0.090648 (2) 1.64e-06 (3) 0.00823	(1) 0.1087 (2) 7.23e-07 (3) 0.01184	(1) 0.0331 (2) 1.04e-06 (3) 0.0011	(1) 0.069 (2) 2.04e-06 (3) 0.00476

Table 4.11: The bias of the expected value of the estimates from the true value of parameter (1), the sample variance of the estimates (2) and the mean squared errors of the estimates (3). "2-star" term of the Markov model for simulated networks.

In the case of "2-star" and "3-star" terms, it is necessary to underline that for network generation the values of this terms have been chosen very small, 0.1 and -0.05, respec-

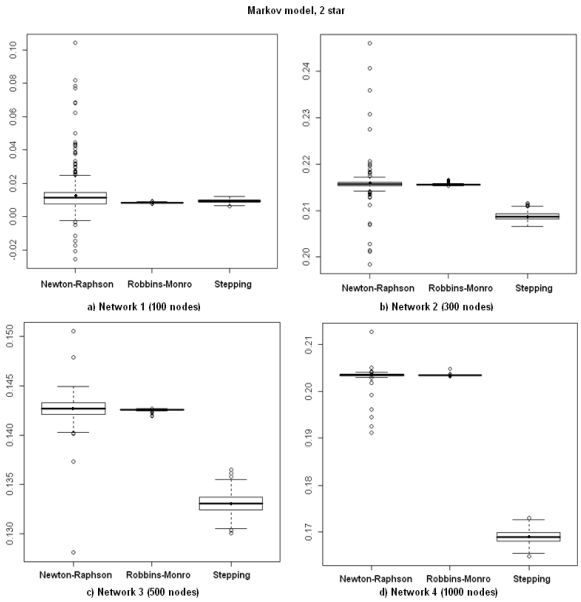

Figure 4.13: Boxplot of distribution of the approximations of the MLE of the "2-star" term of Markov model for 1000 runs (with different seeds) of three different estimation procedures. The dots correspond to the means of the distributions. The true value of the parameter is equal to 0.1.

tively.

	Network 1	**Network 2**	**Network 3**	**Network 4**
Newton-Raphson	(1) 0.04748 (2) 0.00015 (3) 0.00241	(1) -0.02082 (2) 2.07e-06 (3) 0.00044	(1) -0.0047 (2) 4.796e-06 (3) 2.69e-05	(1) -0.01888 (2) 5.041e-07 (3) 0.00036
Robbins-Monro	(1) 0.04269 (2) 1.296e-07 (3) 0.00182	(1) -0.02093 (2) 9.24e-09 (3) 0.00044	(1) -0.00491 (2) 1.21e-08 (3) 2.41e-05	(1) -0.01896 (2) 3.058e-09 (3) 0.00036
Stepping	(1) 0.04193 (2) 1.68e-07 (3) 0.00176	(1) -0.01985 (2) 4e-08 (3) 0.00039	(1) -0.00307 (2) 3.61e-08 (3) 9.42e-06	(1) -0.01378 (2) 4.84e-08 (3) 0.00019

Table 4.12: The bias of the expected value of the estimates from the true value of parameter (1), the sample variance of the estimates (2) and the mean squared errors of the estimates (3). "3-star" term of the Markov model for simulated networks.

Parameters indicate the strength of the effects in the data. If we want the particular

configuration to occur more frequently than by chance (the parameter equal zero) we have to define the corresponding parameter to be large and positive. In this case the values are very close to zero so we can expect that the effect of ”2-star” and ”3-star” in simulated networks is very weak. For the network with 100 nodes parameter estimates are insignificant for almost all runs.

	Network 1	**Network 2**	**Network 3**	**Network 4**
Newton-Raphson	(1) 0.05074 (2) 8.46e-07 (3) 0.00258	(1) -0.28895 (2) 2.89e-08 (3) 0.08349	(1) -0.0878 (2) 2.56e-08 (3) 0.00771	(1) 0.102 (2) 6.76e-08 (3) 0.01039
Robbins-Monro	(1) 0.08152 (2) 8.724e-05 (3) 0.00673	(1) -0.2805 (2) 5.155e-05 (3) 0.07873	(1) -0.0915 (2) 4.858e-05 (3) 0.00842	(1) 0.099 (2) 4.57e-05 (3) 0.00984
Stepping	(1) 0.09485 (2) 8.468e-06 (3) 0.00903	(1) -0.2775 (2) 2.69e-06 (3) 0.07707	(1) -0.0915 (2) 3.063e-06 (3) 0.00838	(1) 0.099 (2) 4.752e-06 (3) 0.0098

Table 4.13: The bias of the expected value of the estimates from the true value of parameter (1), the sample variance of the estimates (2) and the mean squared errors of the estimates (3). “Triangle” term of the Markov model for simulated networks.

For the networks with 300, 500 and 1000 nodes we notice that demonstrate that the Stepping procedure provides smaller bias from the true value of the parameter than the estimates obtained from the other two procedures (Table 4.11, Figure 4.13 (b), (c), (d)). In this case we notice also the smallest values of MSE. For “3 star” estimates we observe the same situation but for all four networks (Table 4.12). In the case of “2-star” and “3-star” statistics the Robbins-Monro exhibits the smallest variation. With regard to the “3-star” statistic the Newton-Raphson exhibits the biggest variability from all three methods. The Newton-Raphson and the Robbins-Monro demonstrate similar values of bias and MSE.

Figure 4.15 demonstrates results obtained for “triangle” statistic. For all four networks the Newton-Raphson procedure exhibits the smallest variation and the Robbins-Monro the highest one. For the networks of 100 and 500 nodes the Newton-Raphson demonstrates estimates which provide the smallest bias from the true value of the parameter and the smallest values of MSE (Table 4.13, Figure 4.15 (a), (c)). For networks with 500 and 1000 nodes the Robbins-Monro and the Stepping demonstrate equal bias but

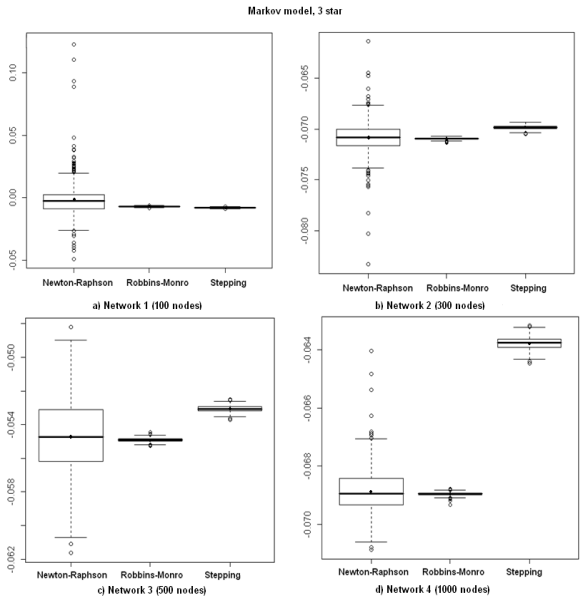

Figure 4.14: Boxplot of distributions of the approximations of MLE of the "3-star" term of Markov model for 1000 runs (with different seeds) of three different estimation procedures. The dots correspond to the means of the distributions. The true value of the parameter is equal to -0.05.

the Stepping exhibits smaller variability.

After analysis based on the simulated networks we consider the Stepping procedure to be more precise in approximation of MLE. It demonstrates the smallest variability for approximation of the true value MLE in the dyadic independence model. For more complex models even though it exhibits higher variation than other techniques it demonstrates results which are the closest to the values of parameters.

The next step of the analysis is to evaluate results obtained for real network data. As real data we consider three networks such as the FLOMARRIAGE, the ECOLI2, and the FAUX.MAGNOLIA.HIGH network data. As in the case of simulated networks, firstly we evaluate the results of fitting the dyadic independence model. Table 4.14 illustrates the bias of the expected value of estimates from the true value of the parameter. In the case of real network data the procedures exhibit negligible bias, there is no bias

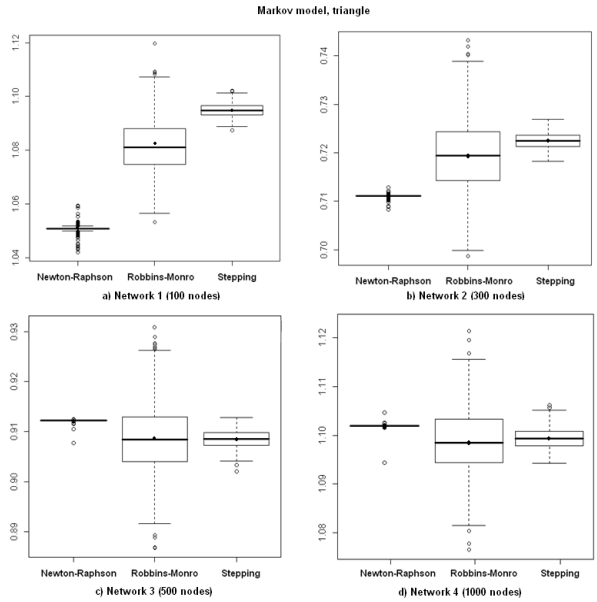

Figure 4.15: Boxplot of distribution of the approximations of MLE of the "triangle" term of Markov model for 1000 runs (with different seeds) of three different estimation procedures. The dots correspond to the means of the distributions. The true value of the parameter is equal to 1.

with an absolute value bigger than 0.00074. For all the three networks the highest bias was demonstrated by the Newton-Raphson procedure, indeed the smallest one by the Robbins-Monro.

	FLOMARRIAGE	ECOLI2	FAUX.MAGNOLIA.HIGH
Newton-Raphson	(1) 0.00072 (2) 0.00368	(1) -0.00057 (2) 0.0002	(1) -0.00074 (2) 0.00013
Robbins-Monro	(1) 0.00004 (2) 5.5431e-05	(1) 0.00002 (2) 1.32456e-05	(1) -0.00033 (2) 1.110617e-05
Stepping	(1) 0.00016 (2) 4.93719e-06	(1) 0.00044 (2) 4.61455e-06	(1) 0.00048 (2) 5.405132e-06

Table 4.14: The bias of the expected value of the estimates from the true value of parameter (1), the mean squared errors of the estimates (2). The Bernoulli model for real network data.

While analyzing results obtained for estimation of simple dyadic independence model for real networks, we can observe that all the three procedures show similar behaviors as

in the case of the simulated networks. The Newton-Raphson demonstrates the highest variability while the Stepping the smallest one. To analyze the accuracy of the proce-

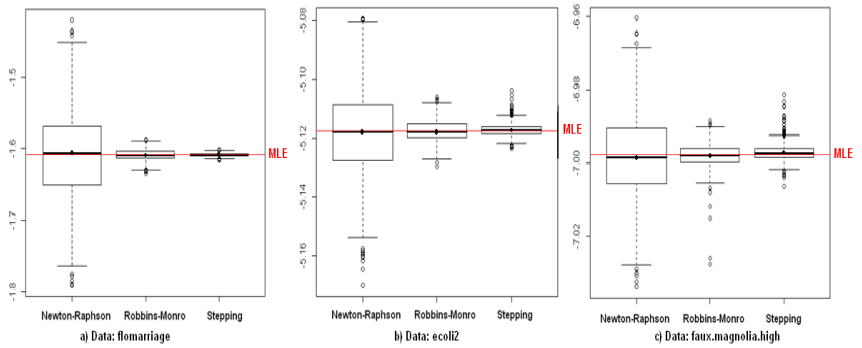

Figure 4.16: Boxplot of distribution of the approximations of MLE of the "edges" statistic for 1000 runs (with different seeds) of three different estimation procedures for real network data (a) FLOMARRIAGE, (b) ECOLI2, (c) FAUX.MAGNOLIA.HIGH. Line illustrates the true value of the parameter and the dots correspond to the means of the distributions.

dures for more complex models we consider models based on the recent developed statistics. In the case of real network data we do not consider the Markov graph model because of its inadequacy in describing real social phenomena. For each network we explore different models because as stated in Section 2.1.3 the effective choice of terms for an ERGM depends on the context and on the structure of the observed network. For each model we also use different values of MCMC arguments to obtain chains sufficiently long.

For the FLOMARRIAGE network we use the model with terms such as the number of edges (denoted "edges"), the geometrically weighted dyadwise shared partner with value of ϕ equal to 0.5 (denoted "gwdsp(0.5,fixed=TRUE)") and the geometrically weighted degree distribution with value of ϕ equal to 1 (denoted "gwdegree (1,fixed=TRUE)"). The value of arguments of MCMC are as follows: `burnin` equal to 1e+4, `MCMCsamplesize` equal to 1e+4 and `interval` equal to 100. Figure 4.17 demonstrates boxplot of the distribution of estimates obtained from different algorithms; each subfigure relates to different term used in the model. We notice that the Robbins-Monro exhibits the smallest variations (Table 4.14). For "edges" and "gwde-

gree" parameters the Stepping algorithm demonstrates the highest variation. There is a pronounced left skew to the edges parameter estimates for the Stepping procedure and the right skew to the gwdegree parameter estimates for the Newton-Raphson and the Robbins-Monro procedures.

It is worth noting that in this version of "ergm" package, the estimates obtained from

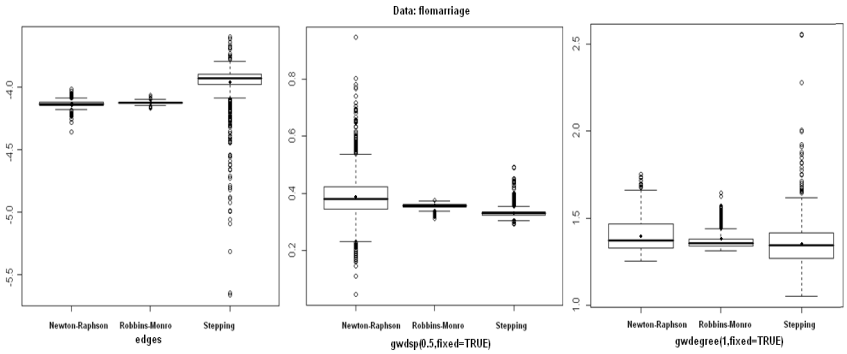

Figure 4.17: Boxplot of distribution of the approximations of MLE for model parameters for real network data FLOMARRIAGE, terms: (a) edges, (b) gwdsp(0.5,fixed=TRUE) and (c) gwdegree(1,fixed=TRUE) of three different estimation procedures. The dots correspond to means of the distributions.

the algorithm Newton-Raphson for small networks do not present such great instability as those obtained from the version 2.2-4 (22 April 2010) and previous versions. Unfortunately there is no documentation of the changes to ergm from version 2.2-4 to 2.2-5 and we are not able to highlight the source of the improvements.

After analysis of the variability of the estimates we decide to evaluate the capacity of estimates provided by the procedures under comparison in simulating networks which resemble the original network. For further analysis we use configurations of parameters based on the expected values of estimates obtained from each estimation procedure (Table 4.15).

We randomly generate a series of 100 networks from and compare how well simulated networks resemble the observed network. Figure 4.18 illustrates histograms of the number of edges and the number of triangles across 100 networks simulated from the expected values of estimates obtained from three estimation procedures, compared to

	edges	gwdsp	gwdegree
Newton-Raphson	(1) -4.132	(1) 0.3867	(1) 1.402
	(2) 0.0007	(2) 0.0083	(2) 0.0079
Robbins-Monro	(1) -4.121	(1) 0.3547	(1) 1.382
	(2) 0.0001	(2) 7.69e-05	(2) 0.0039
Stepping	(1) -3.973	(1) 0.3329	(1) 1.355
	(2) 0.0353	(2) 0.0004	(2) 0.0163

Table 4.15: Expected values (1) and sample variances (2) of the estimates. Model for the FLOMARRIAGE network.

the number of edges (red arrow) and number of triangles (green arrow) of the original network. It is evident that all three parameter sets generate networks which capture well the edges and the triangles counts of the observed network. Figure 4.19 demonstrates

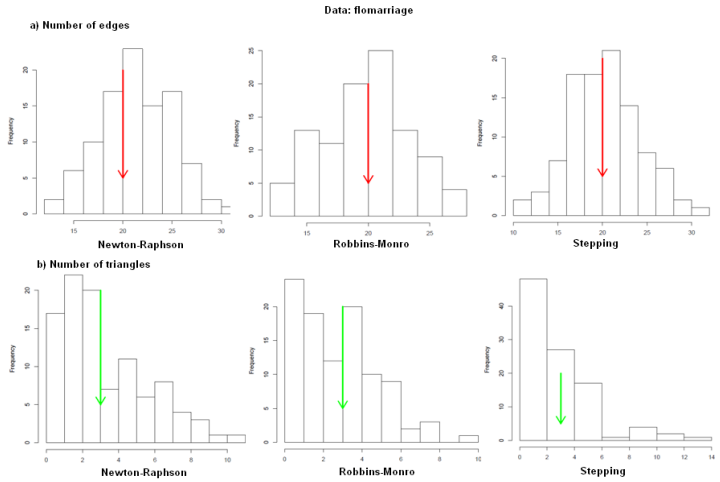

Figure 4.18: The number of edges (a) and the number of triangles (b) across 100 networks simulated using the expected values of estimates obtained from three estimation procedures compared to the number of edges and the number of triangles in the observed network (red arrow and green arrow). Model for the FLOMARRIAGE network.

plots provided by the geometrically goodness of fit and gives the possibility to evaluate how simulated networks resemble the degree distribution, the minimum geodesic distance distribution and the edgewise shared partner distribution of the FLOMARRIAGE network. We notice that all the three parameter sets generate networks which overestimate the number of nodes with degree 2 and underestimate the number of nodes

with degree 3 (Figure 4.19 a). The value from the observed network is smaller than all the 1st quartiles of distributions of the number of nodes with degree 2 and higher that all the 3rd quartiles of distributions of the number of nodes with degree 3. The best capturing of the minimum geodesic distance distributions of the observed network is provided by networks simulated from the parameter set based on the Stepping algorithm estimates. These distributions exhibit the smallest variability in comparison to those of networks generated from the parameters obtained from the Newton-Raphson and the Robbins-Monro procedures (Figure 4.19 b). The edgewise shared partner distribution of the observed network is captured well by all the networks simulated from the parameter sets obtained from the estimation procedures under comparison.

Comparing these three estimation procedures, we are also interested in evaluating their execution times. To examine the computational times of fits provided from the methods under comparison, we sample 100 runs for each procedure and we calculate the mean time of a single run. In our simulations using a 64-bit 2.26GHz Intel Core 2 Duo, the mean time of a single run for the Newton-Raphson take 21.67 seconds (s) versus 12.89 s for the Robbins-Monro and 10.82 s for the Stepping. For the small network the Stepping procedure is the fastest one, its single run is two times faster than a run of the Newton-Raphson method.

For the network ECOLI2 we focus on the model with the six structural parameters such as the number of edges (denoted "edges"), the number of nodes of degree 2 (denoted "degree(2)"), the number of nodes of degree 3 (denoted "degree(3)"), the number of nodes of degree 4 (denoted "degree(4)"), the number of nodes of degree 5 (denoted "degree(5)") and the geometrically weighted degree distribution with value of ϕ equal to 0.25 (denoted "gwdegree (0.25,fixed=TRUE)") [53]. The value of `burnin` is equal to 1e+5, `MCMcsamplesize` equal to 2e+4 and `interval` equal to 1000. Figure 4.20 demonstrates boxplots for distributions of parameter estimates for terms used in the model for the ECOLI2 network. All the estimates distributions demonstrate very low variability with the exception of the estimates obtained from the Newton-Raphson procedure for "edges" term and from the Robbins-Monro for the "degree(5)" and "gwdegree" terms. The smallest dispersions are exhibited by the results of the

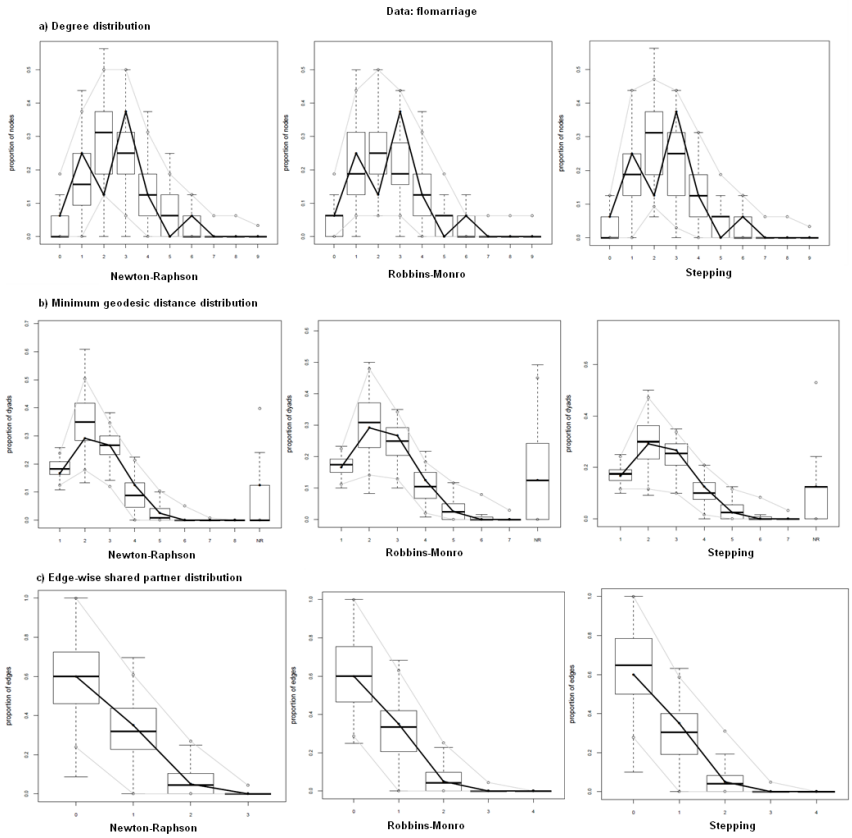

Figure 4.19: Graphical goodness of fit for (a) the degree distribution , (b) the minimum geodesic distance distribution and (c) the the edgewise shared partner distribution. Model for the FLOMARRIAGE network

Stepping procedure. For distributions provided by the Newton-Raphson method we notice the most outlayers. In the case of this model it is worth noting that all the three procedures demonstrate results which are very different to each other. The biggest difference is noted for “gwdegree” term. Because of such different estimates among the method under comparison it is very important to evaluate how simulated networks manage to resemble the observed graph. Configurations of parameters used for further analysis are formed by the expected values of estimates and are presented in Table 4.16.

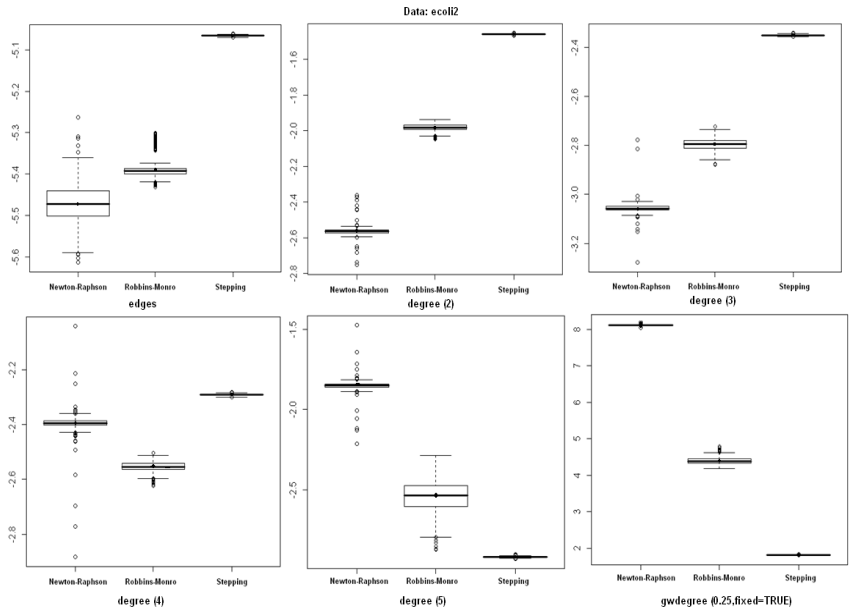

Figure 4.20: Boxplot of distribution of the approximations of MLE for model parameters for real network data ECOLI2, terms: (a) edges, (b) degree(2), (c) degree(3), (d) degree(4), (e) degree(5) and
(f) gwdegree(0.25, fixed = TRUE) of three different estimation procedures. The dots correspond to means of the distributions.

We observe that only the parameter sets obtained from the expected values of estimates produced by the Robbins-Monro and the Stepping generate graphs with the number of nodes which is equal to or very close to the number of nodes of the observed network. Estimates produced by the Newton-Raphson method simulate networks with disproportionate number of nodes (Figure 4.21 a). We observed that none of simulated networks manages to resemble the number of triangles in the network under analysis (Figure 4.21 b). While observing Figure 4.22 b, we can deduce that the Robbins-Monro and the Stepping provide similar graph distributions which underestimate the number of dyads with the minimum geodesic distance equal to 2 and overestimate the number of dyads with the minimum geodesic distance equal 7, even though the latter procedure exhibits slightly better results than the Robbins-Monro. The Newton-Raphson produces networks which are not able to resemble the minimum geodesic distance of the observed

	edges	**degree2**	**degree3**
Newton-Raphson	(1) -5.471 (2) 0.0022	(1) -2.564 (2) 0.0004	(1) -3.056 (2) 0.0003
Robbins-Monro	(1) -5.389 (2) 0.0005	(1) -1.982 (2) 0.0003	(1) -2.795 (2) 0.0005
Stepping	(1) -5.066 (2) 2.2e-06	(1) -1.459 (2) 1.9e-06	(1) -2.35 (2) 4.63e-06
	degree4	**degree5**	**gwdegree**
Newton-Raphson	(1) -2.395 (2) 0.0009	(1) -1.85 (2) 0.0008	(1) 8.122 (2) 3.38e-05
Robbins-Monro	(1) -2.554 (2) 0.0003	(1) -2.54 (2) 0.0092	(1) 4.397 (2) 0.0079
Stepping	(1) -2.292 (2) 7.53e-06	(1) -2.92 (2) 1.27e-05	(1) 1.812 (2) 1.06e-05

Table 4.16: Expected values (1) and sample variances (2) of the estimates. Model for the ECOLI2 network.

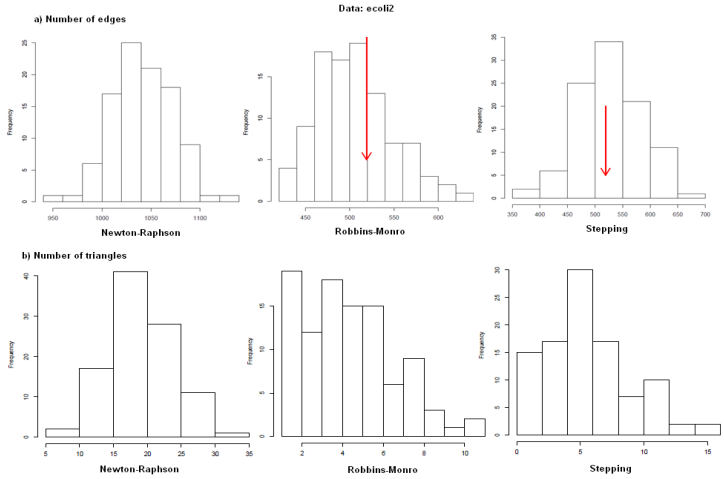

Figure 4.21: The number of edges (a) and the number of triangles (b) across 100 networks simulated using the expected values of estimates obtained from three estimation procedures compared to the number of edges and the number of triangles in the observed network (red arrow and green arrow). Model for the ECOLI2 network.

graph. It manages to resemble only the number of dyads with the minimum geodesic distance equal to 2 (Figure 4.22 b). The Stepping and the Robbins-Monro produce networks which well reconstruct the number of nodes with degrees from 0 to 5. However,

for the number of nodes with degrees higher than 8 we notice that generated networks overestimate these statistics. The networks generated from the parameters obtained form the Newton-Raphson are not able to reconstruct well the number of nodes with degree higher tha 3 (Figure 4.22 a). Since the degree terms has been included in the model we suppose that these parameter values may produce degenerate networks. With regard to the edgewise shared partner distribution (Figure 4.22 c), none of the simulated graph distributions is able to resemble the observed data and because this statistic has not been included in the model we can say that the models are poor fitting for this network feature. It is worth noting that both, configuration of parameters produced by the Robbins-Monro and that produced by the Stepping method generate networks which reconstruct the observed network in the same way even if parameters exhibit values diffrent from one another. Comparing the execution times of three estimation procedures we notice that for the ECOLI2 network the Robbins-Monro method turned out the fastest one with a mean time for a run equal to 114.25 s which is two times faster than the mean time of a single run produced by the Stepping 230.48 s. A mean time for a single run of the Newton-Raphson is 193.83 s.

For the network FAUX.MAGNOLIA.HIGH we focus on the model with the seven structural parameters, related to network statistics such as the number of edges (denoted "edges"), the geometricaly weighted edgwise shared partner with value of ϕ equal to 0.1 (denoted "gwesp(0.1,fixed=TRUE)"), the geometrically weighted dyadwise shared partner with value of ϕ equal to 0.1 (denoted "gwdsp(0.1,fixed=TRUE)"), the geometrically weighted degree distribution with value of ϕ equal to 0.1 (denoted "gwdegree (0.1,fixed=TRUE)") and with uniform homophily nodal attribute effect such as "nodematch("Grade", diff=FALSE)", "nodematch("Race", diff=FALSE)", "nodematch("Sex", diff=FALSE)". The value of `burnin` is equal to 1e+5, `MCMcsamplesize` equal to 1e+5, the number of iterations (`maxit`) equal to 5 and `interval` equal to 100. For this network we notice that the estimates provided by the Newton-Raphson show the smallest variability. The highest one is demonstrated by the results of the Robbins-Monro procedure. For all distributions we observe noticeable skews, in the case of parameter for "edges" term it is a right skew and for other parameters left

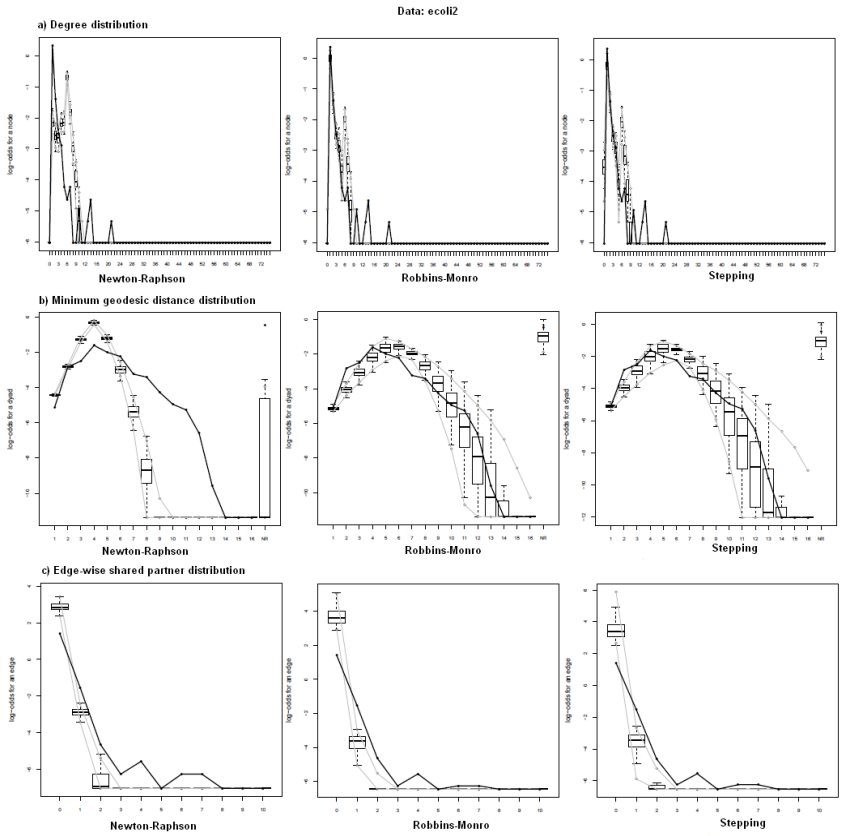

Figure 4.22: Graphical goodness of fit for (a) the degree distribution , (b) the minimum geodesic distance distribution and (c) the the edgewise shared partner distribution. Model for the ECOLI2 network.

skews are dimonstrated. We notice that only in the case of "nodemtach.Race" term the expected values are very close to each other for all the three methods. In the case of the estimates for the geometricaly weighted edgewise shared partner the procedures give different results. The distribution of estimates provided by the Netwon-Raphson has the smallest variance and its expected value is 1.663. The estimates resulted for the Robbins-Monro exhibit the highest variability, the pronounced left skew and its expected value is equal to 2.04. For the Stepping procedure we notice the slight right

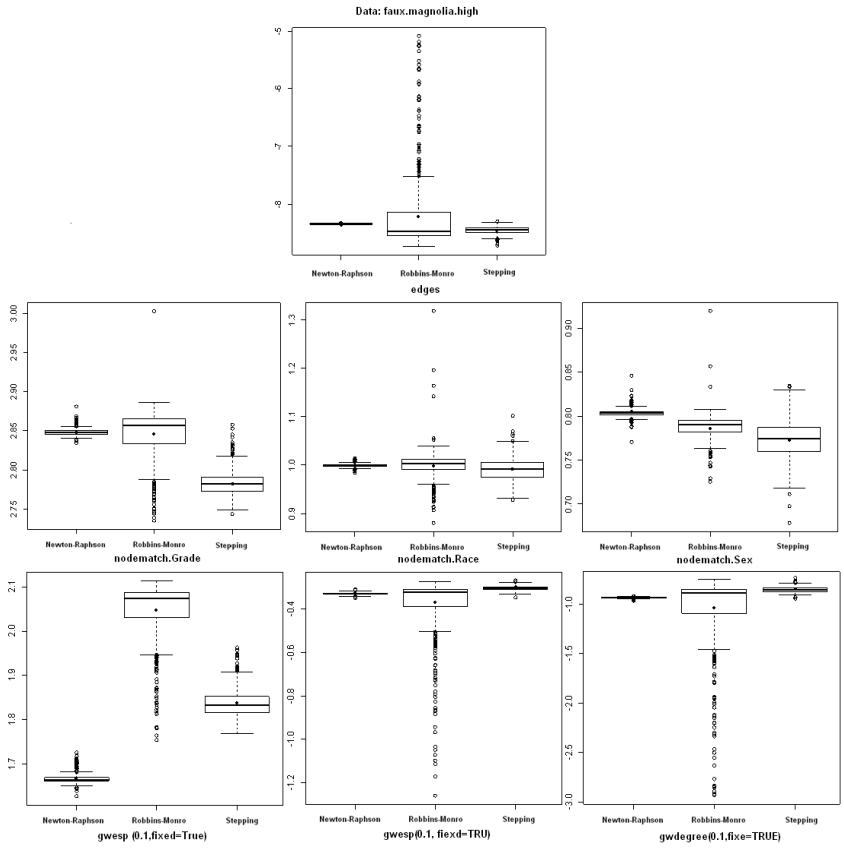

Figure 4.23: Boxplot of distribution of the approximations of MLE for model parameters for real network data FAUX.MAGNOLIA.HIGH, terms: (a) edges, (b) nodematch("Grade", diff=FALSE), (c) nodematch("Race", diff=FALSE), (d) nodematch("Sex", diff=FALSE), (e) gwesp(0.1,fixed=TRUE), (f) gwdsp(0.1,fixed=TRUE),(g) gwdegree (0.1, fixed = TRUE) of three different estimation procedures. The dots correspond to means of the distributions.

skew and the expected value equal to 1.837 (Figure 4.23). The expected values of estimates form configurations of arguments used for further analysis and are illustrated in Table 4.17.

Figure 4.24 illustrates how well the simulated networks resemble the number of nodes of the observed graph. All three distributions contain networks with the number of edges equal to the count of edges in the observed network. However, in the distribu-

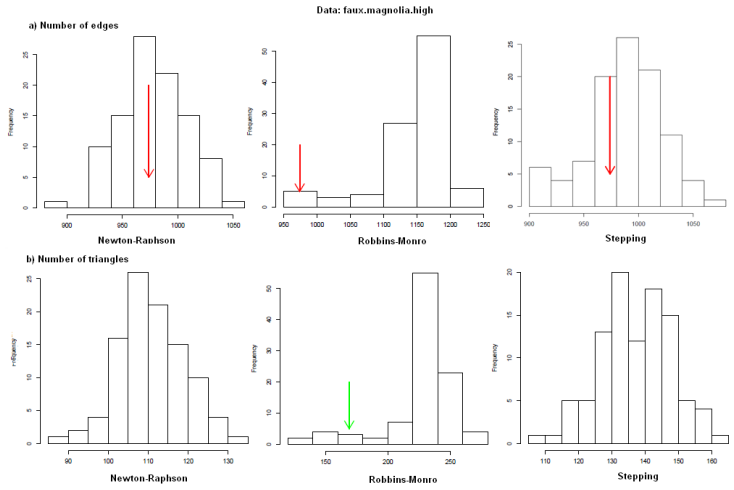

Figure 4.24: The number of edges (a) and the number of triangles (b) across 100 networks simulated using the expected values of estimates obtained from three estimation procedures compared to the number of edges and the number of triangles in the observed network (red arrow and green arrow). Model for the FAUX.MAGNOLIA.HIGH network.

	edges	nodematch.G	nodematch.R	nodematch.S
Newton-Raphson	(1) -8.342 (2) 2.66e-05	(1) 2.850 (2) 3.87e-05	(1) 1 (2) 3.74e-05	(1) 0.8059 (2) 3.8e-05
Robbins-Monro	(1) -8.192 (2) 0.0347	(1) 2.845 (2) 0.0006	(1) 0.9988 (2) 0.0004	(1) 0.7879 (2) 0.0003
Stepping	(1) -8.458 (2) 0.0038	(1) 2.783 (2) 0.0003	(1) 0.9908 (2) 0.0006	(1) 0.7738 (2) 0.0005

	gwesp	gwdsp	gwdegree
Newton-Raphson	(1) 1.667 (2) 0.0001	(1) -0.3286 (2) 4.49e-05	(1) -0.9369 (2) 4.84e-05
Robbins-Monro	(1) 2.05 (2) 0.0065	(1) -0.3851 (2) 0.0009	(1) -1.067 (2) 0.0095
Stepping	(1) 1.837 (2) 0.001	(1) -0.3037 (2) 0.0001	(1) -0.8562 (2) 0.0008

Table 4.17: Expected values (1) and sample variances (2) of the estimates. Model for the FAUX.MAGNOLIA.HIGH network.

tion of networks based on the parameters obtained from the Robbins-Monro, there are only a few of networks with this value of the number of edges and the most frequent are networks with number of edges higher than 1100 and smaller than 1200 (Figure4.24 a).

With regard to the number of triangles we notice that only the configuration of parameters obtained from the Robbins-Monro produce some graphs with the number of triangles similar to that of the observed graph, however the most frequent are networks with higher number of triangles (Figure 4.24 b). Configurations of parameters provided by the other two procedures generate networks with the number of triangles smaller than the presented by the observed graph.
The plots provided by the goodness of fit function (Figure 4.25) demonstrate that networks based on the results obtained from all three methods gives simular results. With regard to the degree distributions we notice that simulated networks slightly overestimate the number of nodes with degrees 2, 3 and underestimate the number of nodes with degrees from 4 to 8; however the networks generated from the Robbins-Monro parameters set give slightly worse results (Figure 4.25 a). With regard to the minimum geodesic distance distribution we notice that the original network exibits the numbers of dyads with the geodesic distances from 7 to 23 lower than in the simulated networks (Figure 4.25 b). However, we observe that the generated networks reflect well the number of dyads with higher values of the minimum geodesi distance. Analyzing the edgewise shared partner distributions, we observe that none of the generated network distributions is able to reflect well the edgewise shared partner distribution of the observed network.

Comparison of the execution times of three estimation procedure reveals that for the FAUX.MAGNOLIA.HIGH network the Robbins-Monro method turned out to be the fastest one with a mean time for a run equal to 446.49 s. The Newton-Raphson demonstrates the longest mean time for a single run equal to 602.18 s. The mean time of a single run produced by the Stepping is 500.39 s.

4.4 Summary of results

Summarizing the main findings of the analysis, we can make some important observations.

Firstly, in the Bernoulli model for both, simulated and real networks, all the three

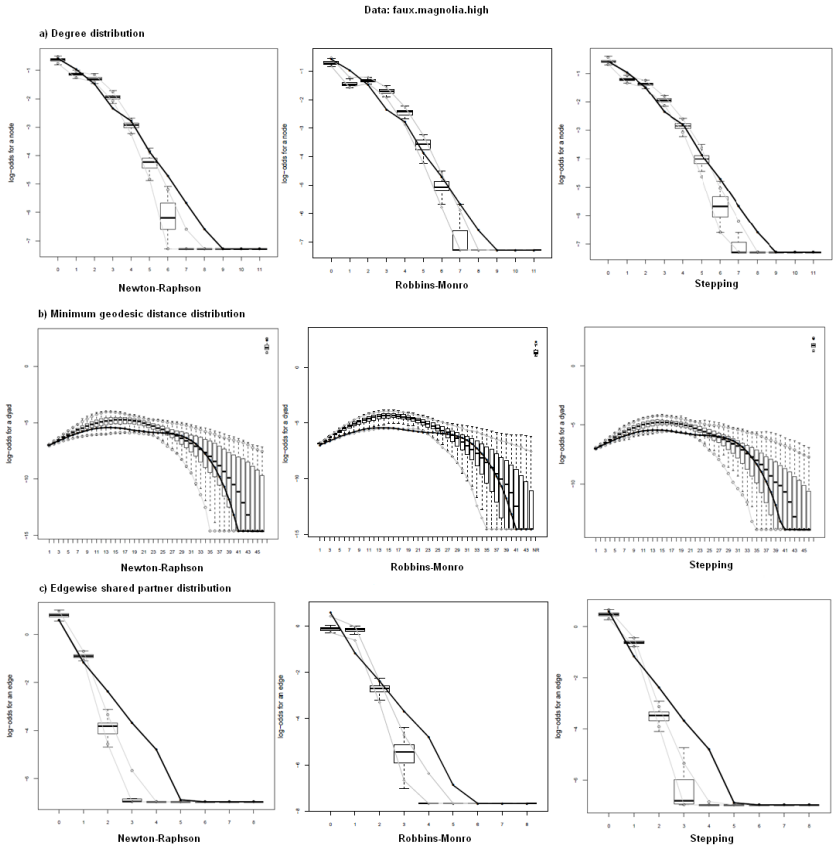

Figure 4.25: Graphical goodness of fit for (a) the degree distribution,(b) the minimum geodesic distance distribution and (c) the the edgewise shared
partner distribution. Model for the FAUX.MAGNOLIA.HIGH network.

methods demonstrated a very small bias from the true value of parameters but the Stepping procedure exhibited the smallest variability of the estimates. For the Markov model for simulated networks, the Stepping procedure provided estimates which exhibited the smallest bias from the true value of parameters. With regard to the ECOLI2 network it produced the combination of parameters which provided a very good model fit whereas the Newton-Raphson method failed.

Secondly, in the version 2.2-4 of the "ergm" package the Newton-Raphson pro-

cedure dimonstrated high instability for estimation of model parameters for a small network which has been got over in the version 2.2-5. This method exhibited the highest variability of the estimates for the "edges" term in the Bernoulli model for both simulated and real networks. It is worth noting that for the ECOLI2 network this algorithm failed in approximation of the MLE and provided estimates which were not able to reproduce networks similar to the original one. For this model the Robbins-Monro and the Stepping procedure gave better results. The Newton-Raphson method demonstrated some shortcomings in the estimation process, especially in the presence of the value of θ_0 which is far from the true value of MLE. In estimating the parameters of the model for FAUX.MAGNOLIA.HIGH network it dimonstrated the smallest variability of estimates and a "good model" for the original network.

Thirdly, the Robbins-Monro procedure demonstrates the smallest variability of estimates for the model for FLOMARRIAGE network. In the case of the Bernoulli model for simulated networks, it exhibited an increase of the number of outlayers with an increase of the number of nodes. With regard to the Markov model for simulated networks, it provided results similar to those of the Netwon-Raphson but more variable, especially for the "triangle" term. It is worth noting that for the ECOLI2 although the Robbins-Monro provided estimates different than those produced by the Stepping procedure, both parameter configurations managed to generate networks which resembled the observed network equally well.

Considering the results produced by the Stepping procedure and its ability for estimation in the case of the value of θ_0, potentially quite far from the true MLE, we reccomend this algorithm as that which provides the best results from the procedures under comparison.

With regard to the value of the approximated maximum likelihood and the value of AIC we noticed that the Robbins-Monro procedure provides values different from those produced by the other two methods. For instance, in the case of the model for the FLOMARRIAGE network, the value of AIC runs of the Robbins-Monro algorithm was about 172, and for the other two methods it was about 110. Differences were noticeable also for models for the other two networks. It is worth noting that the value

of approximated log likelihood and AIC cannot be used to compare models fitted by different procedures.

Analyzing the outputs of the *check.degeneracy* function and the *mcmc.diagnostic* function, we noticed that a high *degeneracy.value* indicates a high instability in the last iteration in the chain, and vice versa, the small value of *degeneracy.value* means a good stability in the last iteration. The *check.degeneracy* function, which can be used only with the Newton-Raphson and the Stepping procedure, may be recommended as a first selection criterion, especially in the situation when results for different runs of the same model are quite different. All diagnostic tools analyzed in this work are very useful in identifying model degeneracy and in assessing goodness of fit and provide a rich picture of the various asspects of estimation process in ERGMs.

Conclusions

In this work we have paid attention to the statistical models for social networks, especially to the Exponential Random Graph Models (ERGMs), the statistical models which provide a general framework for modeling overall network structural properties by the effects of localized interaction rules. This class of models forms a statistical exponential family, it attempts to represent the stochastic mechanisms that produce ties, and the complex dependencies that this induces. Using ERGMs the principal goal is to estimate model parameters from an observed network and then evaluate how adequately the model reproduces the observed data. The development of Monte Carlo estimation methods for ERGMs has given the possibility to understand better model behavior, especially the model degeneracy - the property of random graph models that have an important consequence on the use of ERGMs.

Apart from the review of developments of the ERGMs from the beginning until nowadays, the present manuscript offers two contributions. The first one deals with an evaluation of model diagnostic tools and check for degeneracy of ERGMs using different methods and functions proposed in the literature and available in "statnet" package for R. Taking into account the stochastic nature of the Newton-Raphson estimation algorithm, we examined information provided by degeneracy diagnostics tools for different runs of the same model. The first contribution illustrates degeneracy and instability which appear in different runs of the same model. Results of this analysis show that it may be worth controling degeneracy and instability not only of different models but also of different runs of the same model.

The second contribution provides a comparison of three different procedures such as the

Newton-Raphson, the Robbins-Monro and the Stepping. We analyzed the accuracy and precision of these three methods in approximating the maximum likelihood estimator for networks of various size. Until recently, estimation for ERGMs has been almost exclusively based on the two first methods. Considering the results produced by the Stepping procedure in this analysis and the costruction of this algorithm, we reccomend this method as that which provides the best results and from the procedures under comparison we indicate this one for estimation in this class of models.

As the estimation and the assessment in ERGMs are very important asspects of research applications we are interested in obtaining further developments of the analysis. Further research could be applied to evaluate the stability of the three methods with regard to changes in the values of such MCMC arguments as burnin and interval. We hope that, together with the analysis presented in this work, it would be possible to form a complex picture of estimation procedures which could help to classify these methods in terms of their usefulness for estimation of networks of different sizes and topologies. Since ERGMs are used extensively in statistical modeling of social, economic and biological networks, the analysis presented in this manuscript and its further developments could provide, from the practical points of view, very useful information for everybody who is interested in estimating parameters in this class of models.

APPENDIX A

Appendix

Below we present R code of functions used to obtain results of simulation study. The R packages necessary to be loaded are: ”statnet”, ”slam” and ”coda”. To know all the networks available in ergm package, use the code: data(package = ”ergm”). In the function presented below we use three important function proposed in the package ”erg”, such as *ergm* which fits exponential-family random graph models; *simulate* which is used to simulate random networks using an ERGM and *gof* which evaluates the goodness of fit of an ERGM to the data. To obtain more information and details about features of the package ”ergm”, see [36].

The function ”evaluationERGM” give possibility to make multiple runs of the MLE model fit for given networks, in this case, one of three real network data or networks simulated from a particular distribution. It produces a txt file where information about network, summary for each run of the model fit, numeric information of goodness of fit and mcmc.diagnostics are printed. It provides also a pdf file with depicts of boxplots of distributions of estimates for model parameters, boxplots of distributions of their standard errors and graphical outcomes of goodness of fit and mcmc.diagnostics function.

```
#Description of parameters:
#name_net: Name of network for analysis,
#It is one of the vectors: c("florentine"),
#c("ecoli"),c("faux.magnolia.high"),c("simulated")
```

```
#To analyze a simulated network it is necessary
#to indicate c("simulated") and the parameters:
#param_sim: Vector of statistics for network simulation
#for network generation [i.e. c("edges","kstar(2)")]
#num_nodes: Number of nodes in a simulated network
#theta_sim: Vector of parameters for statistics used
#for network simulation
#burnin_sim: Value of burnin used for network simulation
#inter_sim: Value of interval used for network simulation
#seed_sim: Value of seed for graph simulation
#param: Vector of terms for the model
[i.e. c("edges","kstar(2)")]
#alg: Vector which indicates the name of procedures
#used for estimation: c("Newton-Raphson")
#or c("Robbins-Monro") or ("Stepping")
#n_sim: Number of times the model is run
#burnin_est: Value of burnin used for estimation
#inter_est: Value of interval used for estimation
#MCMCsample_est: Value of MCMCsamplesize used for
#parameter estimation
#it_est: Value of maxit for parameter estimation
#burnin_gof: Value of burnin used for goodness of fit
#inter_gof: Value of interval used for goodness of fit
#seed_gof: Value of seed used for goodness of fit

evaluationERGM<-function(name_net,param_sim,num_nodes,
theta_sim,burnin_sim,inter_sim,seed_sim,param_est,alg,
n_sim,burnin_est,inter_est,MCMCsample_est,it_est,
burnin_gof,inter_gof,seed_gof)
{sink("RESULTS.txt")
```

```
 ########################################
 ### Selection of network for analysis ###
 ########################################
    if (name_net=="simulated")
{form_sim<-as.formula(paste("network(num_nodes,
directed=FALSE)~",paste(param_sim,collapse="+")))
net<-simulate(form_sim,burnin=burnin_sim,
interval=inter_sim,nsim=1,theta0=theta_sim,
verbose=FALSE,seed_sim)
  }
  else
  {
  if (name_net=="florentine")
    { data("florentine")
      net <- flomarriage }
  if (name_net=="ecoli")
    { data("ecoli")
      net <- ecoli2 }
  if (name_net=="faux.magnolia.high")
    { data("faux.magnolia.high")
      net <- faux.magnolia.high }
  }
  print(net)
  print("Network density")
  print(network.density(net))
form_est<-as.formula(paste("net~",
paste(param_est,collapse="+")))
est<-matrix(0,ncol=length(param),nrow=n_sim)
se<-matrix(0,ncol=length(param),nrow=n_sim)
edges<-NULL
```

```
star2<-NULL
star3<-NULL
triang<-NULL
maxlik<-NULL
degen<-NULL
mcmc_se<-NULL
  for(i in 1:n_sim)
  {
  cat("\n")
  cat(paste("RESULTS FROM SIMULATION NUMBER "))
  cat(i,"\n")
  cat("\n")
 ############################
 ###Parameter estimation###
 ############################
  model<-ergm(form_est,burnin=burnin_est,interval=inter_est,
  maxit=it_est,MCMCsample_est,verbose=FALSE,seed=i,
  control=control.ergm(check.degeneracy=TRUE,style=alg))
  if(network.edgecount(model$newnetwork)>50000)
  {model$degeneracy.value=100}
  print("SUMMARY MODEL")
  print(summary(model))
  edges[i]<-network.edgecount(model$newnetwork)
triang[i]<-as.numeric(summary(model$newnetwork~triangle))
star2[i]<-as.numeric(summary(model$newnetwork~kstar(2)))
star3[i]<-as.numeric(summary(model$newnetwork~kstar(3)))
  degen[i]<-model$degeneracy.value
  est[i,]<-model$coef
  maxlik[i]<-model$mle.lik
  mcmc_se[i]<-model$mc.se
```

```
  mcmc.diagnostics(model)
    for(j in 1:length(param))
      {
        se[i,j]<-sqrt(model$covar[j,j])
      }
  }
cat("\n")
cat(paste("RESULTS"))
cat("\n")
################################################
### Evaluation of results and goodness of fit ###
################################################
x<-list(estimates=est,standard_errors=se,edges=edges,
star2=star2,star3=star3,triangles=triang,mle.lik=maxlik,
degeneracy=degen,MCMC_se=mcmc_se)
theta_mean<-apply(est,2,mean)
sample_var<-apply(est,2,var)
print("vectors of results")
print(x)
("sample mean of estimates")
print(theta_mean)
sim<-simulate(form_est,theta0=theta_mean,burnin=burnin_gof,
interval=inter_gof,nsim=100,verbose=FALSE,seed=seed_gof)
gof.model<-gof(form_est,theta0=theta_mean,
GOF=~degree+distance+espartners,verbose=FALSE,
burnin=burnin_gof,interval=inter_gof,seed=seed_gof)
model.edgedist<-sapply(sim$networks,
function(x)summary(x~edges))
net.edge<-summary(net~edges)
model.tridist<-sapply(sim$networks,
```

```
function(x)summary(x~triangle))
net.tri<-summary(net~triangle)
print("GOODNESS OF FIT")
print(gof.model)
    pdf(file="RESULTS.pdf")
      for(j in 1:length(param))
      {
        boxplot(est[,j])
        title("Boxplot of parameter estimates")
        boxplot(se[,j])
        title("Boxplot of standard error")
      }
    boxplot(edges,names="edges")
    title("Boxplot edges")
    boxplot(star2,names="star2")
    title("Boxplot star2")
    boxplot(triang,names="triangle")
    title("Boxplot triangle")
    boxplot(degen,names="degeneracy.value")
    title("Boxplot degeneracy.value")
    hist(model.edgedist)
    arrows(net.edge,20,net.edge,5,col="red",lwd=3)
    hist(model.tridist)
    arrows(net.tri,20,net.tri,5,col="green",lwd=3)
    plot(gof.model,logodds=TRUE)
    plot(sim$networks[[1]])
    dev.off()
  sink()
}
```

Bibliography

[1] Airoldi E. M., Blei D. M., Fienberg S. E.,Xing E. P. (2008). Mixed Membership Stochastic Blockmodels. *Journal of Machine Learning Research* 9, 1981-2014.

[2] Albert R., Barabasi A.L.(2002). Statistical Mechanics of Complex Networks. *Reviews of Modern Physics*. 74(1), 47-97.

[3] Anderson C.J., Wasserman S., Crouch B. (1999). A p* primer: logit models for social networks. *Social Networks*, 21, 37-66.

[4] Barndorff-Nielsen O. E. (1978). *Information and Exponential Families in Statistical Theory* New York: Wiley.

[5] Besag J. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society*, Series B, 36, 96-127.

[6] Besag J. (1975). Statistical analysis of non-lattice data. *The Statistician*, 24, 179-195.

[7] Besag J. (2001). Markov Chain Monte Carlo for Statistical Inference, Working Paper No. 9 University of Washington, USA.

[8] Broder A. et al. (2000). Graph structure in the web. *Comput. Netw.*, 33, 309-320.

[9] Caimo A., Friel N. (2010). Bayesian inference for exponential random graph models. URL:arXiv:1007.5192v2

[10] Carter N., Hadlock C., Haughton D.(2008). Generating random networks from a given distribution. *Computational Statistics and Data Analysis*, 52, 3928-3938.

[11] Cartwright D., Harary F. (1956). Structural balance: A generalization of Heiders theory. *Psychological Review*, 63, 277-292.

[12] Corander J., Dahmström K., Dahmström P.(1998). Maximum Likelihood Estimation for Markov Graphs. Research Report 8, Department of Statistics, University of Stockholm.

[13] Corander J., Dahmström, K., Dahmström, P. (2002). Maximum likelihood estimation for exponential random graph models. In: J. Hagberg (Ed.), Contributions to social network analysis, information theory and other topics in statistics: A festschrift in honour of Ove Frank. Stockholm: Department of Statistics, University of Stockholm.

[14] Crouch B., Wasserman S. (1998). Fitting p*: Monte Carlo maximum likelihood estimation. Paper presented at International Conference on Social Networks, Sitges, Spain, May 28-31.

[15] Davis J.A. (1970). Clustering and hierarchy in interpersonal relations: Testing two theoretical models on 742 sociograms. *American Sociological Review*, 35, 843-852.

[16] Davis J. A., Leinhardt S. (1968). The structure of positive interpersonal relations in small groups. Annual Meeting of the American Sociological Association. Boston, Massachusetts.

[17] Davis J.A., Leinhardt S. (1971). The Structure of Positive Interpersonal Relations in Small Groups. In *Sociological Theories in Progress*, Vol. 2, edited by J. Berger, M. Zelditch, and B. Anderson. Boston: Houghton-Mifflin.

[18] Erdös P., Renyi A. (1959). On random graphs. *I. Publicationes Mathematicae (Debrecen)*, 6, 290-297.

[19] Fienberg S., Wasserman S. (1979). Methods for the Analysis of Data From Multivariate Directed Graphs. Technical Report 351, University of Minnesota, School of Statistics.

[20] Fienberg S., Wasserman S. (1981). Categorical Data Analysis of Single Sociometric Relations. In: Sociological Methodology 1981, Leinhardt S., San Francisco: Jossey-Bass, 156-192.

[21] Firth D. (1993). Bias Reduction in Maximum Likelihood Estimates. *Biometrika*, 80, 27-38.

[22] Frank, O. (1981). A Survey of Statistical Methods for Graph Analysis. In: Sociological Methodology, edited by S. Leinhardt. San Francisco: Jossey-Bass, 110-55.

[23] Frank O., Strauss D. (1986). Markov graphs. *Journal of the American Statistical Association*, 81, 832-842.

[24] Frank O., Nowicki K. (1993). Exploratory Statistical Analysis of Networks. In *Quo Vadis, Graph Theory?* edited by J. Gimbel, J. W. Kennedy, and L. V. Quintas. *Annals of Discrete Mathematics*, 55, 34966.

[25] Galaskiewicz J., Wasserman S. (1979). A Dynamic Study of Change in a Regional Corporate Network. Technical Reports 347, University of Minnesota, School of Statistics.

[26] Geman D., S. Geman S. (1984). Stochastic relaxation, gibbs distributions, and bayesian restoration of images. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 6, 721741.

[27] Geyer C. J., Thompson E. A. (1992). Constrained Monte Carlo maximum likelihood for dependent data. *Journal of the Royal Statistical Society*, B54, 657-699.

[28] Gilks W., Richardson S., Spiegelhalter D. (1996). *Markov Chain Monte Carlo in Practice*. London: Chapman Hall.

[29] Goodreau S.M. (2007). Advances in exponential random graph (p*) models applied to a large social network. *Social Networks*, 29, 231-248.

[30] Goodreau S.M., Handcock M.S., Hunter D.R., Butts C.T., Morris M. (2008). A statnet Tutorial. *Journal of Statistical Software*, 24(9).

[31] Grenander U. (1993). *General Pattern Theory*. Oxford: Oxford University Press.

[32] Hggstrm O., Jonasson J. (1999). Phase transition in the random triangle model. *Journal of Applied Probability*, 30, 1101-1115.

[33] Handcock M.S. (2003a). Assessing degeneracy in statistical models of social networks. Working Paper 39, Center for Statistics and the Social Sciences, University of Washington.

[34] Handcock M.S. (2003b). Statistical Models for Social Networks: Inference and Degeneracy, In: Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers, R. Breiger, K. Carley, P. Pattison, Editors, Comittee on Human Factors, National Research Council

[35] Handcock M.S., Hunter D.R., Butts C.T., Goodreau S.M., Morris M. (2003). Software Tools for the Statistical Modeling of Network Data. Version 2.1-1. http://CRAN.R-project.org/package=statnet.

[36] Handcock M.S., Hunter D. R., Butts C.T., Goodreau S.M., Morris M., Krivitsky P. (2010). Fit, Simulate and Diagnose Exponential-Family Models for Networks, URL http://statnet.org

[37] Handcock M.S., Hunter D.R., Butts C.T., Goodreau S.M., Morris M. (2008). statnet: Software Tools for the Representation, Visualization, Analysis and Simulation of Network Data. *Journal of Statistical Software*, 24(1).

[38] Handcock M. S., Raftery A. E., Tantrum J. M. (2007). Model-based clustering for social networks. *Journal Of The Royal Statistical Society*, Series A, 170, 301-354.

[39] Harary F. (1953) On the notion of balance of a signed graph. *Michigan Math. J.*, 2(2), 143-146.

[40] Heider F. (1946). Attitudes and cognitive organization. *Journal of Psychology*, 21, 107-112.

[41] Heinze G., Schemper M. (2002). A solution to the problem of separation in logistic regression. *Statistics in Medicine*, 21, 2409-2419.

[42] Hoff P. D., Raftery A. E., Handcock M. S. (2002). Latent Space Approaches to Social Network Analysis. *Journal of the American Statistical Association*, 97, 1090-1098.

[43] Holland P.W., Leinhardt S. (1970). A method for detecting structure in sociometric data. *American Journal of Sociology*, 70, 492-513.

[44] Holland P.W., Leinhardt S. (1971). Transitivity in structural models of small groups. *Comapartive Groups Studies*, 2, 107-124.

[45] Holland P.W., Leinhardt S. (1972). Some evidence on the transitive of postive interpersonal sentiment. *American Journal of Sociology*, 72, 1205-1209.

[46] Holland P.W., Leinhardt S. (1975). The statistical analysis of local structures in social networks. In: Heise D.R. (ed.) Sociological Methodology, 1976, pag: 1-45, San Francisco: Jossey-Bass

[47] Holland P.W., Leinhardt S. (1977a). Social Structure as a Network Process,*Zeitschrift fur Soziologie*, 6, 386-402.

[48] Holland P.W., Leinhardt S. (1977b). A Dynamic Model for Social Networks, *Journal of Mathematical Sociology*, 5, 5-20.

[49] Holland P.W., Leinhardt S. (1977c). Notes on the statistical analysis of social network data. Unpublished manuscript.

[50] Holland P.W., Leinhardt S. (1978). An omnibus test for social structure using triads. *Sociological Methods and Research*, 7, 227-256.

[51] Holland P.W., Leinhardt S. (1979). Structural sociometry. In: Holland, P.W., Leinhardt, S. (eds.) Perspectives on Social Research, 63-83. New York, Academic Press

[52] Holland P.W., Leinhardt S. (1981). An exponential family of probability distributions for directed graphs. *Journal of the American Statistical Association*, 76, 33-65.

[53] Hummel R. M., Hunter D. R., Handcock M. S. (2010). A Steplength Algorithm for Fitting ERGMs, Penn State Department of Statistics Technical Report.

[54] Hunter D.R. (2007). Curved exponential family models for socila networks, *Social Networks*, 29, 216-230.

[55] Hunter D.R., Handcock M.S., Butts C.T., Goodreau S.M., Morris M. (2008). ergm: A Package to Fit, Simulate and Diagnose Exponential-Family Models for Networks. *Journal of Statistical Software*, 24(3).

[56] Hunter D.R., Handcock M.S. (2006). Inference in curved exponential family models for networks. *Journal of Computational and Graphical Statistics*, 15(3), 565-583.

[57] Hunter D.R., Goodreau S.M., Handcock M.S. (2008). Goodness of Fit for Social Network Models. *Journal of the American Statistical Association*, 103, 248-258.

[58] Jeong H., Tombor B., Albert R., Oltavi Z. N., Barabási A.-L. (2000). The large-scale organization of metabolic networks. *Nature* 407, 651-654.

[59] Johnsen E.C. (1985). Network macrostructure models for the Davis-Leinhardt set of empirical sociometrices. *Social Networks*, 7, 203-224.

[60] Johnsen E.C. (1986). Structure and process: Agreement models for friendship formation. *Social Networks*, 8, 257-306.

[61] Katz L., Powell J.H. (1955). Measurement of the Tendency Toward Reciprocation of Choice. *Sociometry*, 18, 403-409.

[62] Katz L., Powell J.H. (1957). Probability distributions of random variables associated with a structure of the sample space of sociometric investigations. *Annals of Mathematical statistics*, 28, 442-448.

[63] Kenny D., Nasby W. (1980). Splitting in Reciprocity Correlation. *Journal of Personality and Social Psychology*, 38, 249-256.

[64] Koehly L. M., Pattison P. (2005). Random Graph Models for Social Networks: Multiple Relations or Multiple Raters. In: Carrington, P.J., Scott, J., Wasserman, S. (Eds.), Models and Methods in Social Network Analysis. Cambridge University Press, Cambridge, 162-191.

[65] Koskinen J.H., Snijders T.A.B. (2007). Bayesian inference for dynamic network data. *Journal of Statistical Planning and Inference*, 13, 3930-3938.

[66] Lazega E., van Duijn M.A.J. (1997). Position in formal structure, personal characteristics and choices of advisors in a law firm: a logistic regression model for dyadic network data. *Social Networks*, 19, 375-397.

[67] Leenders R.Th.A.J. (1995). Models for network dynamics: a Markovian framework. *Journal of Mathematical Sociology*, 20, 1-21.

[68] Moreno J., Jennings H. (1938). Statistics of Social Configurations, *Sociometry*, 1, 342-374.

[69] Morris M., Handcock M.S., Hunter D.R. (2008). Specification of exponential-family random graph models: terms and computational aspects. *Journal of Statistical Software*, 24(4).

[70] Newman M. E. J. (2002). Random graphs as models of networks. www.santafe.edu/media/workingpapers/02-02-005.pdf

[71] Newman M. E. J. (2001). The structure of scientific collaboration networks, Proc. Natl Acad. Sci. USA 98, 404-409.

[72] Newman M. E. J., Strogatz S. H., Watts D. J. (2001). Random Graphs with Arbitrary Degree Distributions and Their Applications. *Physical Review*, E 64, 026-118.

[73] Nowicki K., Snijders T. A. B. (2001). Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96, 1077-1087.

[74] Padgett J.F. (1994). Marriage and Elite Structure in Renaissance Florence, 1282-1500. Paper delivered to the Social Science History Association.

[75] Park J., Newman M. (2004). Solution of the 2-star model of a network. Condensed Matter abstracts, cond-mat/0405457.

[76] Pattison P.E., Wasseraman S. (1999). Logit models and logistic regressions for social networks. II. Multivariate relations. *British Journal of Mathematical and Statistical Psychology*, 52, 169-194.

[77] Pattison P.E., Robins G.L. (2002). Neighbourhood-based models for social networks. *Sociological Methodology*, 32, 301-337.

[78] Pattison,P., Robins G. (2004). Building models for social space: Neighbourhood-based models for social networks and affiliation structures. *Math. Sci. hum. Mathematics and Social Sciences*, 168(4), 11-29.

[79] Resnick M.D., Bearman P.S., Blum R.W. et al. (1997). Protecting adolescents from harm. Findings from the National Longitudinal Study on Adolescent Health. *Journal of the American Medical Association*, 278, 823-32.

[80] Robbins H., Monro S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, 22, 400-407.

[81] Robins G.L., Elliott P., Pattison P.E. (2001a). Network models for social selection processes. *Social Networks*, 23, 1-30.

[82] Robins G.L., Elliott P., Pattison P.E. (2001b). Network models for social influence processes. *Psychometrika*, 66, 161-190.

[83] Robins E., Pattison P.E. (2005). Interdependencies and Social Processes: Dependence Graphs and Generalized Dependence Structures, In: Carrington, P.J., Scott J., Wasserman, S. (Eds.), Models and Methods in Social Network Analysis. Cambridge University Press, Cambridge, pp. 192-214.

[84] Robins G.L., Pattison P.E., Kalish Y., Lusher D. (2007). An Introduction to Exponential Random (p*) models for social networks. *Social Networks*, 29, 173-191.

[85] Robins G.L., Pattison P.E., Woolcock J. (2005). Small and Other Worlds: Global Network Structures from Local Processes. *American Journal of Sociology*, 110, 894-936.

[86] Robins G.L., Pattison P.E., Wasserman S. (1999). Logit models and logistic regressions for social networks. III. Valued relations. *Psychometrika*, 64, 371-394.

[87] Robins G.L., Snijders T.A.B., Wang P., Handcock M.S., Pattison P.E. (2007). Recent developments in exponential random graph (p*) models for social networks, *Social Networks*, 29, 192-215.

[88] Salgado et al. (2001). Regulondb (version 3.2): Transcriptional Regulation and Operon Organization in Escherichia Coli K-12. *Nucleic Acids Research*, 29(1), 72-74.

[89] Sampson F. (1969). *Crisis in a Cloister*, PhD dissertation, Cornell University, Dept. of Sociology.

[90] Shen-Orr et al. (2002). Network Motifs in the Transcriptional Regulation Network of Escerichia Coli. *Nature Genetics*, 31(1), 64-68.

[91] Skvoretz J., Faust K. (1999). Logit Models for Affiliation Networks. *Sociological Methodology*, 29(1), 253-280.

[92] Snijders T.A.B. (1991). Enumeration and simulation methods for 0-1 matrices with given marginals. *Psychometrika*, 56, 397-417.

[93] Snijders T.A.B. et al. (2007). Manual of SIENA version 3. ICS, University of Groningen, Groningen.

[94] Snijders T.A.B. (2001). The statistical evaluation of social network dynamics. In: M.E. Sobel and M.P. Becker (eds.), Sociological Methodology-2001, 361-395. Boston and London: Basil Blackwell.

[95] Snijders T.A.B. (2002). Markov chain Monte Carlo estimation of exponential random graph models. *Journal of Social Structure*, 3, 2.

[96] Snijders T.A.B., van Duijn, M.A.J. (1997). Simulation for statistical inference in dynamic network models. In: Simulating Social Phenomena. Eds. Conte R. Hegselmann R. Terna P. Berlin, Springer.

[97] Snijders T.A.B., van Duijn M.A.J. (2002). Conditional maximum likelihood estimation under various specifications of exponential random graph models. In: Jan Hagberg (Ed.), Contributions to social network analysis, information theory, and other topics: A festschrift in honour of Ove Frank. University of Stockholm: Department of Statistics.

[98] Snijders T.A.B., Koskinen J., Schweinberger M. (2010). Maximum Likelihood Estimation for Social Network Dynamics, *Ann. Appl. Stat.*, 4(2), 567-588.

[99] Snijders T.A.B., Pattison P.E., Robins G.L., Handcock M.S. (2006). New specifications for exponential random graph models. *Sociological Methodology*, 36, 99-153.

[100] Stawinoga A. (2010). On the diagnostic tools for the analysis of degeneracy in ERGMs. XXX International Sunbelt Social Network Conference in Riva del Garda, Italy.

[101] Strauss D. (1986). On a General Class of Models for Interaction. *SIAM Review*, 28, 513-527.

[102] Strauss D., Ikeda M. (1990). Pseudo-likelihood estimation for social networks. *Journal of the American Statistical Association*, 85, 204-212.

[103] Van Duijn M.A.J., Snijders T.A.B., Zijlstra B.H. (2004). p2: a random effects model with covariates for directed graphs. *Statistica Neerlandica*, 58, 234-254.

[104] Van Duijn M. A. J, Gile K.J, Handcock M.S. (2009). A framework for the comparison of maximum pseudo-likelihood and maximum likelihood estimation of exponential family random graph models. *Social Networks*, 31, 52-62.

[105] Warner R., Kenny D., Stoto M. (1979). A New Round Rubin Analysis of Variance for Social Interaction Data. *Journal of Personality and Social Psychology*, 37, 1742-1757.

[106] Wasserman S. (1979). A Stochastic Model for Directed Graphs with Transition rates Determinde Reciprocity. In: Sociological Methodology 1980, ed. k. Schuessler, san Francisco: Jossey-Bass.

[107] Wasserman S. (1980). Analyzing Social Networks As Stochastic Processes. *Journal of the American Statitical Association*, 75, 280-294.

[108] Wasserman S., Faust K. (1994). *Social Network Analysis: Methods and Applications*, Cambridge University Press.

[109] Wasserman S., Pattison, P.E. (1996). Logit models and logistic regressions for social networks: I. An introduction to Markov random graphs and p*. *Psychometrika*, 60, 401-426.

[110] Wasserman S., Robins, G.L. (2005). An introduction to random graphs, dependence graphs, and p*. In: Carrington, P.J., Scott, J., Wasserman, S. (Eds.), Models and Methods in Social Network Analysis. Cambridge University Press, Cambridge, 148-161.

[111] Watts D. J. (1999). Small Worlds: The Dynamics of Networks between Order and Randomness. Princeton, N.J.: Princeton University Press.