

The Pennsylvania State University
The Graduate School

**IMPROVING ESTIMATION FOR EXPONENTIAL-FAMILY RANDOM
GRAPH MODELS**

A Dissertation in
Statistics
by
Ruth M. Hummel

© 2011 Ruth M. Hummel

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2011

The dissertation of Ruth M. Hummel was reviewed and approved* by the following:

David R. Hunter
Associate Professor of Statistics
Dissertation Advisor, Chair of Committee

James L. Rosenberger
Professor of Statistics

Murali Haran
Associate Professor of Statistics

Scott D. Gest
Associate Professor of Human Development

Bruce Lindsay
Department Head and Professor of Statistics

*Signatures are on file in the Graduate School.

Abstract

A statistical model for observed network data allows us to both summarize quantitatively the effects that give rise to the network and simulate new networks from a probability distribution that resembles the generating distribution of the original network. This thesis discusses a flexible class of statistical models for network data, called ERGMs (exponential family random graph models), and describes some of the challenges in parameter estimation for these models. We present a novel computational method that results in more usability of Markov chain Monte Carlo methods for estimation in ERGMs, enabling an MCMC MLE to be more easily found in many cases. This so-called *stepping* algorithm is illustrated on a transcriptional regulation network for *E. coli*, an example where previous attempts to approximate an MLE had failed.

In light of this new development in approximating the MLE, additional alternative estimators are introduced, including the frequently-used MPLE (*maximum pseudolikelihood estimate*), as well as bias-adjusting versions of both the MPLE and the MCMC MLE, a bias-adjustment for the latter being applied here for the first time in this context. Comparisons are made between these estimators and the MCMC MLE on a network of corporate law partnerships, showing improved accuracy and efficiency of the estimation for the bias-adjusted approximate MLE.

Additional estimators are introduced in the context of *contrastive divergence* methods, frequently used in computer science and statistical mechanics. The connection between contrastive divergence and likelihood maximization is explored, and the wide class of *composite likelihood* functions is applied to ERGMs, with estimation by contrastive divergence.

This work advances the current practice of parameter estimation and improves the accuracy of prediction and inference for network models. Two of the computational methods introduced in this thesis, the stepping algorithm and the penalized MLE, are now implemented in the publicly available `ergm` package for R.

Table of Contents

List of Figures	vii
List of Tables	x
Chapter 1	
Introduction to social network analysis	1
1.1 What is social network analysis?	1
1.1.1 Terminology	1
1.1.2 Examples of social networks	4
1.1.3 Branches of analysis	5
1.1.4 Model construction	8
1.2 A history of statistical social network analysis	10
1.2.1 Dyadic independence models: Erdős-Rényi-Gilbert	10
1.2.2 Transitivity models	12
1.2.3 More dyadic independence: Holland and Leinhardt's $p1$ model	12
1.2.4 Markovian dependence	13
1.2.5 Beyond Markovian dependence: $p2$ and general p^* models	16
1.2.6 Advanced model terms in ERGMs	17
1.3 Inference for ERGMs	19
1.3.1 The loglikelihood function	19
1.3.2 Maximum pseudolikelihood estimation	20
1.3.3 Statistical inference in Markov graphs	21
1.3.4 MCMC-based estimators	22
1.4 Recent related advances in ERG modeling	24
1.4.1 Missing data	24
1.4.2 Network sampling	24
1.5 Summary of this work	25

Chapter 2	
A steplength algorithm for fitting ERGMs	26
2.1 Introduction	26
2.2 Approximate maximum likelihood estimation methods	28
2.3 Example: The Gilbert–Erdős–Rényi model	31
2.4 Partial stepping	32
2.5 Estimating Standard Errors	35
2.6 Application to a Gilbert–Erdős–Rényi model	36
2.7 Application to network datasets	39
2.7.1 A biological network	39
2.7.2 A Sociological Dataset	42
2.8 Discussion	46
Chapter 3	
Bias-adjusted maximum likelihood estimation for exponential-family random graph models	48
3.1 Introduction	48
3.2 Bias-reduction theory	51
3.2.1 Asymptotic bias	51
3.2.2 Bootstrap and jackknife estimates of bias	52
3.2.3 Inapplicability of these methods to ERGMs	54
3.2.4 Firth penalty	54
3.2.5 Choice of parameterization	57
3.3 Bias-adjusted MLE	60
3.4 Study Design	62
3.5 Description of the data and model	63
3.6 Results	64
3.7 Discussion	70
Chapter 4	
Contrastive divergence likelihood estimation in ERGMs	72
4.1 Introduction	72
4.2 Contrastive divergence	73
4.3 Pseudolikelihood example	77
4.4 Gibbs sampling in contrastive divergence	83
4.4.1 Formulation of two sampling schemes: random scan and full scan	83
4.4.2 Variation and efficiency	85
4.5 Composite likelihood and blocked contrastive divergence	87
4.6 Contrastive divergence using composite likelihood in ERGMs, with triads as blocks	89
4.7 Discussion	92

Chapter 5	
Summary and Conclusions	93
5.1 General summary	93
5.2 Summary of major contributions	98
5.3 Future work	99
Bibliography	101

List of Figures

1.1	Simple graph of a directed network.	2
1.2	The same network, enumerated in an adjacency matrix.	2
1.3	The same network, now enumerated in an edgelist.	2
1.4	All possible triads in a directed network, taken from Frank and Strauss (1986).	3
1.5	A number of commonly referenced network configurations. Where arrowheads are present, the subgraph is specific to directed networks; otherwise, the subgraph is specific to undirected networks.	4
1.6	Friendship network: Reciprocated friendships of sixth graders in a rural PA school in Fall 2002. Sex is indicated by the color of the node (pink for girls; blue for boys) and nodes are sized such that the larger the node, the higher the score of that node on the peer-nominated aggression rating.	5
1.7	Biological network: Salgado's <i>E. coli</i> transcriptional regulation network (self-loops in green).	5
1.8	A network generated from the simplest model: Dyadic independence. Also called an Erdős-Rényi-Gilbert model or a Bernoulli model, this model has only one parameter, the log-odds of the probability that, for any dyad, an edge exists.	11
2.1	Approximated values of $\ell(\boldsymbol{\eta}) - \ell(\boldsymbol{\eta}_0)$. The solid line is the true loglikelihood (2.5), maximized at $\hat{\eta} = -0.625$. The dashed lines are equation (2.6) with idealized sample sizes 10^3 , 10^5 , 10^{10} , and 10^{15} . The dotted line is equation (2.7).	31
2.2	On the left, the boundaries of the convex hull of the MCMC sample statistics (the dots) are represented by dotted lines. Since the true MLE $\hat{\boldsymbol{\xi}}_\infty = \hat{\boldsymbol{\xi}} = \mathbf{g}(\mathbf{y}^{\text{obs}})$ is not inside the boundaries, we apply step-halving two times to find a value $\hat{\boldsymbol{\xi}}_1$ that is inside. On the right, the true log-likelihood ratio $l_\infty(\boldsymbol{\eta}) - l_\infty(\boldsymbol{\eta}_0)$ is not well-approximated, particularly at $\hat{\boldsymbol{\eta}}_\infty$. However, substituting $\hat{\boldsymbol{\xi}}_1$ for $\mathbf{g}(\mathbf{y}^{\text{obs}})$ yields a new log-likelihood ratio, also shown here, that is much better approximated in the neighborhood of its maximum.	34

2.3	The solid line is the true log-likelihood ratio function, the dotted line is Approximation (2.6), and the dashed line is Approximation (2.7). The values of η_0 are 1.099 in (a) and -0.571 in (b). Since the closed-form map from mean-value parameters ξ to canonical parameters η is known, the range of η values corresponding to the range of sampled statistics is shown as a short segment with its endpoints marked in each plot.	37
2.4	Distribution of the approximate MLE for the final step of the example, for the naive approximation (top) and the lognormal approximation (bottom), for $\eta_0 = -.53$ (on left) and $\eta_0 = -.55$ (on right). The thick dashed lines indicate the minimum and maximum η values found by each approximation, and the thick solid line marks the true MLE.	38
2.5	The <i>E. coli</i> transcriptional regulation network of Shen-Orr et al. (2002), with self-regulating nodes depicted as triangles.	40
2.6	The model including a 2-star term and fit using MPLE produces unrepresentative random networks.	40
2.7	Plots of the sampled statistics from selected iterations, showing that ξ (the small black point) is moving from the mean of the sample (the star) towards the observed statistics (the large black point) without leaving the convex hull of the sample. In the final plot, ξ is the same as the observed statistics.	43
2.8	Sociational network of Kapferer (1972). An edge between nodes i and j indicates that Kapferer observed person i and person j interacting in their work environment on more than one occasion during time period 1. The vertices have size proportional to their sociality (number of workers with whom they interacted during time 1).	44
3.1	Modification of the unbiased score function.	55
3.2	The Lazega collaboration network for the 36 partners, with the size of the nodes proportional to the partner's seniority at the firm, and with the nodes colored according to the primary office location of the partner. The Boston partners are shown in red and include the two isolates. The Hartford partners are in green; and the lone partner in Providence is shown in blue.	64
3.3	Boxplots of the <i>Edge</i> and <i>Gwesp</i> parameters estimated by each of the four methods for the same 1,000 random networks, each network generated from a known underlying parameter (denoted by dashed line). Bootstrapped 95% confidence limits for the means of each method are also shown.	65

3.4	Boxplots of the nodal covariate parameters for <i>Seniority</i> and <i>Practice</i> estimated by each of the four methods for the same 1,000 random networks, each network generated from a known underlying parameter (denoted by dashed line). Bootstrapped 95% confidence limits for the means of each method are also shown.	66
3.5	Boxplots of the homophily parameters for <i>Practice</i> , <i>Gender</i> , and <i>Office</i> estimated by each of the four methods for the same 1,000 random networks, each network generated from a known underlying parameter (denoted by dashed line). Bootstrapped 95% confidence limits for the means of each method are also shown.	67
4.1	The Lazega collaboration network for the 36 partners, with the size of the nodes proportional to the partner's seniority at the firm, and with the nodes colored according to the primary office location of the partner. The Boston partners are shown in red and include the two isolates. The Hartford partners are in green; and the lone partner in Providence is shown in blue.	80

List of Tables

1.1	Interpretation of some cases of the $p1$ model.	13
1.2	Some statistics for directed p^* models.	16
1.3	New terms for ERG models.	18
2.1	Compare means to the true MLE, -0.6246769 . In some of the 10,000 samples drawn for each η_0 , $\hat{\xi} = g(y^{obs})$ is not in the convex hull of the sampled statistics, so, when necessary, we show the results with these cases removed for both approximations.	39
2.2	Model terms and MPLEs for the E. coli dataset of Shen-Orr et al. (2002). The first model is the model specified and fit in Saul and Filkov (2007). The second model is our adaptation to avoid issues of degeneracy. . . .	41
2.3	Approximate MLEs (with standard errors and MCMC standard errors in parentheses) obtained using the stepping algorithm and the lognormal approximation (2.7). Model 2 is based on the analysis of Saul and Filkov (2007) but omits the problematic 2-star term.	42
3.1	Coverage rates of perceived (95%) t-confidence intervals for the approximate MLE, the bias-adjusted approximate MLE (pMLE), the MPLE, and the bias-adjusted MPLE (MBLE).	68
3.2	Relative efficiency (as the ratio of mean squared errors) of the bias-adjusted approximate MLE (pMLE), the MPLE, and the bias-adjusted MPLE (MBLE), compared to the MLE.	68
3.3	Asymptotic standard errors for the approximate MLE, the bias-adjusted approximate MLE (pMLE), the MPLE, and the bias-adjusted MPLE (MBLE).	69
4.1	Simulation arguments used for CD- n estimation using the <code>ergm</code> function in the <code>ergm</code> package (version 2.3) in R.	80
4.2	Simulation results for parameter estimation using MPLE, CD-1, CD-2, CD-5, and MCMC-MLE (CD- ∞), with MCMC standard errors. (Usual standard errors are reported for the MPLE in square brackets.)	82

Dedication

With thanks to Abe Lincoln for saying it first, *Everything I am or ever hope to be, I owe to my mother.*

Thank you, Mum. And congratulations on *our* Ph.D.

Introduction to social network analysis

1.1 What is social network analysis?

1.1.1 Terminology

A *network* is a set of *nodes*, V , (also called *vertices*), on which exists a set of *edges*, E , (sometimes called *arcs*), describing the presence of a specified (directed or undirected) relationship within pairs of nodes. For example, the nodes could be individual humanitarian aid organizations working in the aftermath of Hurricane Katrina, and an edge could connect two nodes that collaborated in providing aid.

We refer to two nodes as a *dyad*, and that dyad has an *edge* if the specific relationship of interest exists between the two nodes in the dyad. A network is easily visualized graphically, as in Figure 1.1.

Here the nodes are represented by points and the edges represented by arcs connecting nodes. A network can also be visualized as an *adjacency matrix*, as in Figure 1.2, or an *edgelist*, as in Figure 1.3. An *adjacency matrix* is a v by v matrix, where v is the number of nodes in the network, and the row names and column names are the names (or correspond directly to the names) of the nodes (for example, the aid organizations of the previous example). The entries in the adjacency matrix are, for an undirected and

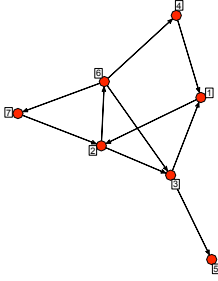


Figure 1.1. Simple graph of a directed network.

	1	2	3	4	5	6	7	Y_{i+}
1	0	1	0	0	0	0	0	1
2	0	0	1	0	0	1	0	2
3	1	0	0	0	1	0	0	2
4	1	0	0	0	0	0	0	1
5	0	0	0	0	0	0	0	0
6	0	0	1	1	0	0	1	3
7	0	1	0	0	0	0	0	1
Y_{+j}	2	2	2	1	1	1	1	Y_{++}

Figure 1.2. The same network, enumerated in an adjacency matrix.

Edgelist	
(3, 1)	(6, 3)
(4, 1)	(6, 4)
(1, 2)	(3, 5)
(7, 2)	(2, 6)
(2, 3)	(6, 7)

Figure 1.3. The same network, now enumerated in an edgelist.

unweighted network, ones and zeros, depending if the corresponding row and column nodes share the relationship of interest (e.g., collaboration). So the ij th element of the adjacency matrix is zero if there is no relationship between nodes i and j and is one if there is and hence only the upper triangle of the adjacency matrix is necessary for an undirected network. When the relationship of interest has an implicit direction in its application (node i sends this relationship to node j , or node k reports this relationship with node l), this information can be incorporated into the network by adding a direction to the edges. For a directed network, the ij th element, Y_{ij} , of the adjacency matrix is zero if there is no relationship from node i to node j and is one if there is a relationship from node i to node j . In a directed network Y_{ij} does not necessarily equal Y_{ji} (perhaps the New Orleans Police Department (NOPD) claims to work with The Red Cross, but The Red Cross does not report collaborating with NOPD). The adjacency matrix for a directed network has the additional meaning that row i corresponds to the *out*-edges of node i ; that is, row i will have “1”s in the entries corresponding to the nodes which receive an edge coming from node i , and will have zeros in all other entries. Likewise, column i of an adjacency matrix for a directed network will correspond to the *in*-edges of node i .

The network can also be characterized by an edgelist. An *edgelist* is simply a list of the edges that are present in the network (see Figure 1.3). For a directed network, the direction is indicated by the order of the nodes in the pairing. For example, the entry (NOPD, Red Cross) would indicate that NOPD reported collaborating with the Red Cross and (Red Cross, NOPD) would indicate that the Red Cross reported collaborating

with NOPD. NB: The term *edge* is often used to indicate the existence of a relationship between two nodes in an undirected network, while *arc* is usually used to indicate a directed edge.

The *degree distribution*, frequently the focus of network analysis in physics and applied mathematics, is the summary of the degrees in a network (in- and out- degrees in a directed network.) In Figure 1.2 the in- and out-degree distributions are the marginal totals (column sums, Y_{+j} , for out-degree and row sums, Y_{i+} , for in-degree) shown in the adjacency matrix.

Several specific edge configurations in networks are frequently referenced in model-building and throughout the literature. One such useful configuration, a *clique*, is, in an undirected network, a completely full subgraph. That is, each member of a subset of the network's node set is connected to every other member in the subset. *Triangles* are cliques in sets of three nodes. A *triad* is any configuration on three nodes. All possible

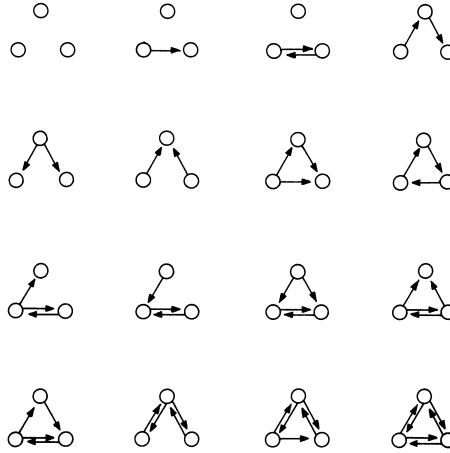


Figure 1.4. All possible triads in a directed network, taken from Frank and Strauss (1986).

triad configurations for a directed network are shown in Figure 1.4. Dyadic and triadic configurations are illustrated in Figure 1.5.

Another frequently-referenced network configuration is the *k-star*. This configuration consists of a set of exactly k edges that share a node in common. In “*More triadic terms*” in Figure 1.5 the three types of 2-stars for directed graphs are shown. In each case the node i is the node in common to the two edges.

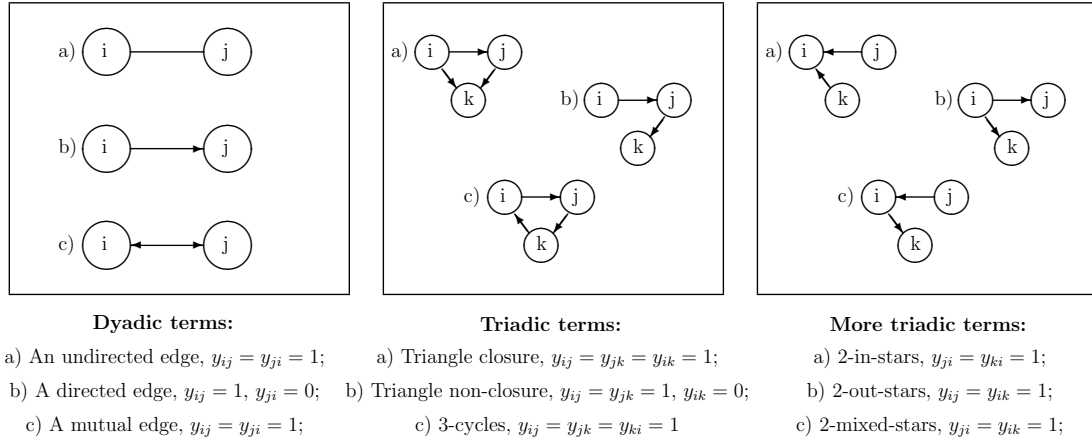


Figure 1.5. A number of commonly referenced network configurations. Where arrowheads are present, the subgraph is specific to directed networks; otherwise, the subgraph is specific to undirected networks.

1.1.2 Examples of social networks

Networks arise naturally in myriad settings. In social science, a number of large-scale longitudinal studies have been made of friendship networks in schools. Figure 1.6 is a graphical representation of a friendship network of 135 sixth-grade students in a small rural school in central Pennsylvania (Gest et al., 2007). This representation also illustrates some of the additional features of the collected network data, beyond the list of nodes and edges, that are of interest. (In this case, sex and peer-nominated aggression rating are shown.) These features are called *nodal attributes*, or *nodal covariates*. In this example, sex, grade, teacher-rated academic skills, number of times each student is named as a most-liked or most-disliked peer by other peers, and peer-nominated aggression rating are the nodal attributes of interest. We could illustrate any of these nodal attributes by changing the features of the nodes or edges in the graphical representation as in Figure 1.6.

Social network analysis allows us to test theories about how certain characteristics of a pair of nodes (such as being of the same sex, or having at least one aggressive student in the pair, or having at least one other friend in common) increases or decreases the odds of such a pair of nodes forming a reciprocated friendship.

In Chapter 2 of this thesis, a biological network example (Salgado et al., 2001; Alon,

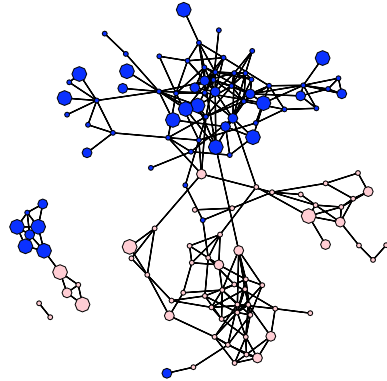


Figure 1.6. Friendship network: Reciprocated friendships of sixth graders in a rural PA school in Fall 2002. Sex is indicated by the color of the node (pink for girls; blue for boys) and nodes are sized such that the larger the node, the higher the score of that node on the peer-nominated aggression rating.

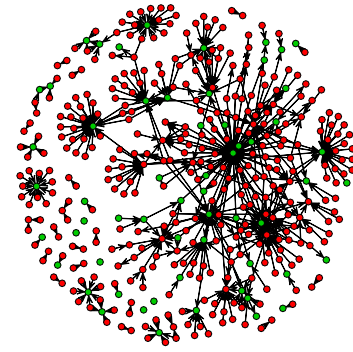


Figure 1.7. Biological network: Salgado's *E. coli* transcriptional regulation network (self-loops in green).

2007; Saul and Filkov, 2007) is described. This network (shown in Figure 1.7) is a transcriptional regulation network for *E. coli*, such that edges between nodes indicate that the operons represented by the connected nodes are in a regulating relationship. The analysis of this network tests the strength of certain local features, specifically the propensity for nodes to have degree 2, 3, 4, or 5, and for nodes to form 2-stars.

It should be clear from these examples that networks arise from a variety of contexts, with a variety of associated research questions. What they all have in common is that there is a set of elements of interest (nodes) and a relationship observed between them (edges). The nodes can be individual people or animals, parts of a cell, communities, businesses, concepts, etc. The edges can be based on friendship relationships, germ-spreading contact relationships, gene regulation relationships, writing or working collaboration relationships, email relationships, physical proximity relationships, or many other possibilities.

1.1.3 Branches of analysis

Analysis of networks can be divided into at least three schools of thought. The first school can be called the *descriptive methods*. Practitioners in this group employ techniques

to *describe* an observed network, specifically focusing on features such as centrality, betweenness, patterns of in- and out-degree (degree distribution), mixing matrices for counts of ties between partners with matching or non-matching personal attributes, and so on, as opposed to applying inferential techniques to estimate parameters. Moxley and Moxley (1974), Freeman (1977), Freeman (1979), for example, focus on measures of centrality. Nosanchuk (1963), Alba (1973), Roistacher (1974) discuss some clique-finding algorithms. Software packages such as UCINET (Borgatti et al., 1999) and Pajek: A Package for Network Visualization (Batagelj and Mrvar, 2009) are specific to these descriptive methods.

Network structure analysis has frequently focused on describing the degree distribution and measures of centrality and connectedness. As far as descriptives of the observed network, these measures are very informative. For example, if we want to know which nodes are highly influential in tying the network together, we can identify hub nodes. We can see how the network structure changes when we remove highly connected individuals. We can experiment with how the spread of a disease across this network changes when we remove the most highly connected nodes.

Occasionally this sort of descriptive work takes on an inferential component by including tests of randomness for certain observed structures based on the distribution of these structures in a random graph. Examples include the distribution of triads in a random graph (Holland and Leinhardt, 1978) and the distribution of the number of isolates (Katz, 1952). Often the observed distribution of the graph structure of interest is compared to a model of pure randomness, which may be a bit too simplistic (Frank and Strauss, 1986). In other words, we can test whether an observed network has significantly more triads than we would expect by chance if all of the edges in the network were being formed independently of each other, and with some constant probability of edge-occurrence. These assumptions under which we can calculate the number of triads expected by chance comprise the baseline distribution, or baseline model. Much of the literature uses simplistic baseline models that do not describe real-life networks very accurately. Section 1.2 introduces a larger class of baseline models, as well as some specifications that seem to lend themselves to much more realistic models.

Another framework of analysis is *actor-oriented* or *agent-based modeling*. This type of analysis views the development of a social network as the result of the actions of individual actors, where the actors' behavior is modeled through a utility function describing the propensity for an actor to form or break a certain type of tie (Snijders et al., 2009). The modeling of *panel* data via agent-based modeling is discussed in van de Bunt et al. (1999) and Schweinberger (2005). Panel data — that is, repeated observations over time of the relationships between nodes in a fixed group of nodes — is a term frequently used to describe *longitudinal* data for networks.

The approach taken in this dissertation is called *network-oriented modeling*. This type of analysis uses the entire network as the unit of analysis and can be used on *cross-sectional* network data or *panel* data. This thesis is concerned with cross-sectional data. The application of network models to panel data is called *dynamic modeling*. Such modeling, for networks with a fixed set of actors that are changing in structure over time, has a history in loglinear models (Wasserman and Iacobucci, 1988; Wasserman and Faust, 1994) and continuous time Markov chain models (Wasserman, 1980; Snijders and van Duijn, 1997; Leenders, 1995). In these fixed-actor continuous-time models, the edges are considered to be either present or absent, and only one dyad from the set is typically permitted to toggle its status (switch to the opposite state) per unit of time (Goldenberg et al., 2009). Multiple (and possibly unobserved) changes are permitted in the models used in Snijders et al. (2007a) and Koskinen and Snijders (2007). Notably, Holland and Leinhardt were the first to propose a continuous time Markov framework, discussed in Holland and Leinhardt (1981) and in Holland and Leinhardt (1977b) and Holland and Leinhardt (1977a). Snijders (2001) and Snijders (2007) summarize much of the work in dynamic modeling for statistical network models.

It may be desirable to generalize the effects we find in one social network to a population of similar social networks. One goal might be to collect data on multiple networks and treat these networks as a sample from the larger population. This framework would be an extension of *actor-oriented* or *network-oriented modeling*. Snijders and Baerveldt (2003) presents a meta-analysis of groupwise network analyses, and others have also suggested that random coefficient models could form a stronger basis for so-called *multilevel*

network analysis (Snijders, 2007; Schweinberger and Snijders, 2007).

1.1.4 Model construction

Robins et al. (2007a) describe an approach to model construction, which is summarized here.

1. *Choose a modeling framework.* As discussed in the previous section, there are a variety of perspectives from which one can attempt to view the social processes that give rise to networks. In this thesis, the modeling framework will always be the the cross-sectional exponential-family random graph model class, Equation 1.1, which will be defined briefly at the beginning of Section 1.2 and with a more complete discussion in Section 1.3.1.
2. *Determine a dependence hypothesis and specify the resulting model.* The following section will discuss the development of these models and their dependence hypotheses. Holland and Leinhardt classified social network analysis into five categories: cross-sectional data, time series data (longitudinal, panel data), covariates, valued relationships, and multiple relationships (Holland and Leinhardt, 1981). Time series network data, comprised of multiple cross-sectional networks, require estimation methods to assess changes in the cross-sectional estimates over time. Covariate information - that is, information about the nodes or dyads in the network - can be applied to both cross-sectional and time series data. With time series data it is often, but not always, assumed that this covariate information cannot change over time. Adding valued relationships (by, for example, weighting the edges according to some criteria, or allowing categories of “existence” of edges, e.g. 0, 1, or 2) further complicates the analyses in either static (cross-sectional) or time-series contexts. Multiple relationships can also be included in the same model. For example, a network involving the 37 workers in a certain company and signifying their email contacts with one type of edge, their friendship relationships with another type of edge, and their corporate hierarchy with a third type of edge would be a network with multiple simultaneous relationships. As noted before,

this dissertation will deal with cross-sectional data rather than time series data, and the extension to multiple relationships will not be considered. Covariate information will be incorporated through terms specified in the model (as described in section 1.2). Valued relationships (even directed networks) will not be considered in detail, but the extension of the methods developed in this work are immediately applicable to such networks.

3. *Estimate model parameters, determine the goodness of fit of the fitted model, and interpret the parameters.* For many models, estimation of the parameters can be a difficult task, as will be discussed in Section 1.3. Interpretation will be given for a few examples in Sections 1.3.

Model fit is an important issue in any model-based analysis. We want to know: Does a graph generated from our fitted model have a similar structure to our observed graph? One way to assess adequacy of models is to generate many simulated graphs from the fitted model in question and calculate higher-order statistics from these simulated graphs, as in Goodreau et al. (2008). We can then use these samples of statistics to calculate empirical distributions for the statistics and check if the observed network has very different values for the higher-order statistics than do the graphs simulated from the fitted model. If there is no noticeable difference, then we have reason to believe that the model adequately captured those higher-order characteristics of the network. However, as we discuss in Section 1.2.6, infinite values for parameter estimates are possible for both well-informed and poorly-informed models of the forms described in Section 1.2, and for “typical” data as well as for “extreme” data.

In fact, most of the models presented in Section 1.2, when fitted to the data, are frequently inadequate representations of real social network data. The fact that the models are inadequate might be a simple lack of fit as discussed in Hunter et al. (2008). In more serious cases, the inadequacy of the model might be due to the inclusion of certain terms that essentially preclude any meaningful estimation. This condition is termed *model degeneracy*; it is discussed in broad terms by Handcock (2003), then defined rigorously and analyzed in depth by Schweinberger (2010). For network models, some of the possible appropriate predictors are network-structure features, such as the overall

density of edges in the network, or the propensity for an edge to exist when its existence would close a triangle. Section 1.2.6 presents some very promising model specifications, which, when included in the hypothesized model, seem to improve the ability of the model to adequately describe the data for many naturally-arising social networks (see Handcock, 2003).

1.2 A history of statistical social network analysis

Although not proposed in this form until Holland and Leinhardt (1981), with the $p1$ model, and in Wasserman and Pattison (1996) with the full class of p^* models (both described later in this section), it serves the discussion of historical models to first describe the general class of exponential family random graph models (ERGMs), a broad class of statistical models for the description and analysis of network data. We define the ERGM class for simple loopless, unweighted networks as in Hunter et al. (2007) and many of the contained references:

$$P_{\boldsymbol{\eta}}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y})\}}{\kappa(\boldsymbol{\eta})}, \quad \mathbf{y} \in \mathcal{Y}, \quad (1.1)$$

where $\mathbf{Y}$ is a random network written as an adjacency matrix so that Y_{ij} is the indicator of an edge from i to j ; $\mathbf{g}(\mathbf{y})$ is a vector of the network statistics of interest; $\boldsymbol{\eta}$ is the vector of parameters measuring, in some sense, the strength of the effect of each corresponding entry in the vector $\mathbf{g}(\mathbf{y})$; $\kappa(\boldsymbol{\eta})$ is the constant of proportionality that makes the probabilities sum to one; and $\mathcal{Y}$ is the space of allowable networks.

We now discuss some important historical social network models as they relate to this ERGM class.

1.2.1 Dyadic independence models: Erdős-Rényi-Gilbert

Statistical exponential family models were used in mathematical social network analysis as early as 1953. At this time, Anatol Rapoport, a mathematical psychologist who had written about popularity differentiation and reciprocation in choice structures among people (Rapoport, 1949, 1950, 1957), first introduced the dyadic independence model as

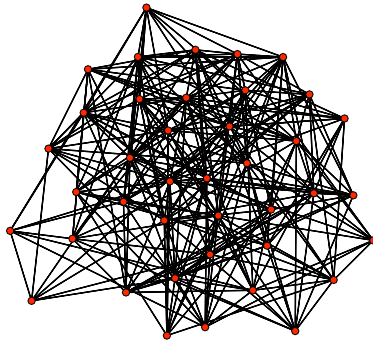


Figure 1.8. A network generated from the simplest model: Dyadic independence. Also called an Erdős-Rényi-Gilbert model or a Bernoulli model, this model has only one parameter, the log-odds of the probability that, for any dyad, an edge exists.

a “pure randomness model” against which to compare hypothesized network behavior as systematic deviations from that pure randomness model (Rapoport, 1953). Erdős and Rényi (1959) and Gilbert (1959) are generally credited with developing the formal properties of this model.

This Erdős-Rényi-Gilbert model specifically describes the probability of observing a certain configuration in a network of fixed size, V , and with exactly E edges. This corresponds to a model proposing that edges in a network of fixed size, V , form independently of all other edges, with some fixed probability for formation.

An immediate extension to the Erdős-Rényi-Gilbert is a model with a fixed node set, but with different edge-probabilities for some dyad pairs. This specification would require separating the set of dyads into classes (perhaps based on node-matching for some nodal attribute), with the largest number of distinct edge probability parameters occurring when each dyad has its own probability. Such a model is still a dyadic independence model, but with (at least one) covariate effect and more (≥ 2) edge-formation parameters.

The simple Erdős-Rényi-Gilbert dyad-independence model (as well as the extension to multiple edge-formation parameters) is contained in the class of exponential family random graph models, specifically as

$$P_{\eta}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\{\eta^t e(\mathbf{y})\}}{\kappa(\eta)}, \quad \mathbf{y} \in \mathcal{Y}, \quad (1.2)$$

where the statistic $e(\mathbf{y})$ is the number of edges in the graph and η is the parameter

corresponding to $e(\mathbf{y})$, $\eta = \log(p/(1 - p))$, the log-odds of an edge.

Although the Erdős-Rényi-Gilbert model is a special case of the ergm, the model class was not identified as a broader exponential family until Holland and Leinhardt (1981).

1.2.2 Transitivity models

As in Holland and Leinhardt (1978), Katz (1952), and Rapoport (1953) (mentioned above), Davis (1970) and Holland and Leinhardt (1976) compared network-structural characteristics, such as transitivity, to simple baseline models. Davis compared the observed transitivity in a network to the expected transitivity in that network had it been generated from a model controlling only the one uniform probability for the existence of an edge. Holland and Leinhardt compared to a model allowing only a tendency toward reciprocity of ties (but not transitivity of ties). In this way it was possible to test the frequency of transitivity in the observed data compared to the baseline models, but it was not possible to estimate the strength of the transitivity (or other network-structural characteristic) effect in the network, as in typical statistical parameter estimation, e.g. for the strength of an effect in regression.

1.2.3 More dyadic independence: Holland and Leinhardt's *p1* model

Wasserman (1980) and Holland and Leinhardt (1981) first proposed log-linear models for social network analysis, which allowed dyad-independent network terms such as node- and dyad-level effects to be modeled, and thereby broadened the set of possible null models. Holland and Leinhardt were also the first to formulate their model explicitly as a case of the canonical exponential-family model (Holland and Leinhardt, 1981). They state: “Exponential families of distributions are natural choices to consider for this purpose, since they explicitly tie sufficient statistics to parameters.” They called this model class *p1*. Earlier models, such as the Erdős-Rényi and the transitivity models, were, like *p1*, special cases of an exponential family class of models, but it was Holland and Leinhardt (1981) who first used this form explicitly.

The $p1$ model is of the form

$$P_{\boldsymbol{\eta}}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y})\}}{\kappa(\boldsymbol{\eta})} = \frac{\exp\{\rho m + \theta y_{++} + \sum_i \alpha_i y_{i+} + \sum_j \beta_j y_{+j}\}}{\kappa(\boldsymbol{\eta})} \quad (1.3)$$

where $(\rho, \theta, \boldsymbol{\alpha}, \boldsymbol{\beta})$ make up the parameter vector $\boldsymbol{\eta}$, and the corresponding statistics, $\mathbf{g}(\mathbf{y})$, are the number of reciprocated edges in the observed graph (m), the total number of edges (y_{++}), the out-degree of node i (y_{i+}), and the in-degree of node j (y_{+j}). As such, the parameters have the following interpretations: ρ is a measure of mutuality, θ of the overall density, α_i the productivity of node i , and β_i the attractiveness of node i .

Table 1.1, taken from Holland and Leinhardt (1981), shows some special cases of the $p1$ model.

Parameter Values	Interpretation
$\rho = \theta = \alpha_i = \beta_j = 0$	The uniform distribution in which all digraphs are equally likely.
$\rho = \alpha_i = \beta_j = 0$	Y_{ij} are iid; $\theta = \log(p/(1-p))$.
$\alpha_i = \beta_j = 0$	pairs of dyads are iid; $m_{ij} = m$, $a_{ij} = a_{ji} = a$, $n_{ij} = n$ and $m + 2a + n = 1$.
$\rho = \beta_j = 0$	Y_{ij} are iid in each row of $\mathbf{Y}$; $\theta + \alpha_i = \log_e(p_i/(1-p_i))$.
$\rho = \alpha_i = 0$	Y_{ij} are iid in each column of $\mathbf{Y}$; $\theta + \beta_j = \log_e(p_j/(1-p_j))$.
$\rho = 0$	Y_{ij} are independent; logit of p_{ij} is additive.
$\rho = \infty$	$Y_{ij} = Y_{ji}$ and the graph is symmetric.
$\rho = -\infty$	$Y_{ij}Y_{ji} = 0$ and the digraph is asymmetric.

Table 1.1. Interpretation of some cases of the $p1$ model.

1.2.4 Markovian dependence

Based on developments in spatial statistics in Besag (1974), Frank and Strauss (1986) introduced Markovian dependence to social network analysis. While a dyad-independence assumption was made in all log-linear models studied prior to this seminal work, Frank and Strauss show how other assumptions about the dependence structure (particularly Markov dependence) lead to various families of log-linear models for graphs.

Markovian dependence is defined as follows: *A graph G is a Markov Graph, or has*

Markov dependence, if nonincident dyads in G are conditionally independent.

The condition for Markovian dependence is that the dependent dyads must be “nearest neighbors,” where the nearest neighbors to a dyad (u, v) are those that share a vertex in common ((u, w) and (v, w) for any other node w). In other words, the probability of an edge existing at dyad (j, k) depends on (and only on) the status of the additional dyads to which nodes j and k belong (its *incident* dyads).

Using the Hammersley-Clifford Theorem to link dependence structures to the probability function, Frank and Strauss articulated the probability functions for several new dependence structures in terms of observable network features (Frank and Strauss, 1986; Besag, 1974). Since triangles and stars (see Figure 1.5 for an illustration of these) are shown to be the sufficient subgraphs for general Markov graphs, Frank and Strauss showed that any undirected Markov graph has probability:

$$P_{\tau, \sigma_k}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\left(\tau t(\mathbf{Y}) + \sum_{k=1}^{n-1} \sigma_k s_k(\mathbf{Y})\right)}{\kappa(\tau, \sigma)}$$

where $t(\mathbf{y})$ is the number of triangles in the graph $\mathbf{y}$, $s_k(\mathbf{y})$ is the number of k -stars in $\mathbf{y}$, τ is the parameter corresponding to the number of triangles, σ_k , $1 \leq k \leq n - 1$, are the parameters corresponding to the k -stars, and $\kappa(\tau, \sigma)$ is the constant of proportionality that makes the probabilities sum to one.

This model is called the *homogeneity model* in Frank and Strauss (1986) because it defines the parameters to be homogenous across all vertices instead of allowing each set of 3 vertices to have its own probability of forming a triangle and each set of k vertices to have its own probability of forming a k -star.

In this Markov Graph model, the parameter τ measures an aspect of *transitivity* in the data, and σ measures *clustering*. *Transitivity* here is triad closure (the making of triangles). This can occur as a result of a number of social processes, and it might be beneficial to be able to distinguish among these. The transitivity effect of the Markov Graph model is not parceled into the triad closure that results from, say, the nodes sharing the same covariates as opposed to the general propensity for any two nodes to close a triad.

The *clustering* modeled here is that which is not already explained by the triad closure. This could be driven by self-organization into groups, homophily for some nodal attribute (that is, that the two nodes in the edge share an attribute in common), or any number of other explanations.

A special case of the homogeneity model is called $\rho\sigma\tau$. We can capture both the transitivity and clustering in the model

$$P_{\rho,\sigma,\tau}(\mathbf{Y} = \mathbf{y}) = \kappa^{-1} \exp\{\rho r(\mathbf{Y}) + \sigma s(\mathbf{Y}) + \tau t(\mathbf{Y})\}, \quad (1.4)$$

where $\rho = \sigma_1$, $r = s_1$, $\sigma = \sigma_2$, and $s = s_2$. This is called the *triad model* or the $\rho\sigma\tau$ model. If σ and τ are 0, we have a Bernoulli graph with edge probability $p = (\exp \rho)/(1 + \exp \rho)$. In some cases ρ is a nuisance parameter to be eliminated by conditioning on r . Then, for graphs of r edges, we have

$$P_{\sigma,\tau}(\mathbf{Y} = \mathbf{y}) = \kappa^{-1} \exp\{\sigma s + \tau t\}$$

In the general homogenous case the various star parameters are hard to interpret jointly in view of the multiple counting. For example, every k -star contains $\binom{k}{j}$ j -stars for all $j < k$. Using d_k , the number of vertices of degree k , we have

$$P_{\tau,\delta}(\mathbf{Y} = \mathbf{y}) = \kappa^{-1} \exp \left(\tau t + \sum_{j=1}^{n-1} \delta_j d_j \right)$$

where $\tau > 0$ gives a bias toward transitivity, $\tau < 0$ against, and the δ_j control the bias for or against vertices of degree j .

Despite the increased applicability (compared to simpler models) of Markovian dependence to describe real-world phenomena, the Markovian model still suffers from model degeneracy (discussed in section 1.1.4) in numerous situations. For this reason, the Markov model is not generally of much use in practical modeling problems. For a further discussion of model degeneracy, and suggestions to avoid it, see section 1.2.6.

1.2.5 Beyond Markovian dependence: $p2$ and general p^* models

The *Markov Graphs* paper was followed by additional work in Frank (1991), which generalized the application of the canonical ERG model to social networks. A further specific model, the $p2$ model, was introduced in van Duijn et al. (2004). The $p2$ model extended the dependence structure to model the dependence between the relations to and from the same actor or node. Zijlstra et al. (2006) discusses the $p2$ model further and introduces a multi-level model for multiple observations of networks.

Wasserman and Pattison (1996) introduced the term p^* to describe any social network model of the Exponential-family Random Graph Model form (1.1). All of the models described thus far in this section can be described as p^* models. Typical model terms for $\mathbf{g}(\mathbf{Y})$ in recent use (late 1990s) are shown in Table 1.2 (Anderson et al., 1999).

Table 1.2. Some statistics for directed p^* models.

Term	Description	Graph Statistic ($\mathbf{g}(\mathbf{Y})$)
<i>Dyadic</i>		
	Choice (total number of edges)	$\sum_{i,j} Y_{ij}$
	Mutuality (reciprocated ties)	$\sum_{i < j} Y_{ij} Y_{ji}$
<i>Triadic</i>		
	Transitivity (triangle closure)	$\sum_{i,j,k} Y_{ij} Y_{jk} Y_{ik}$
	Intransitivity (triangle non-closure)	$\sum_{i,j,k} Y_{ij} Y_{jk} (1 - Y_{ik})$
	Cyclicity (3-cycles)	$\sum_{i,j,k} Y_{ij} Y_{jk} Y_{ki}$
	2-in-stars	$\sum_{i,j,k} Y_{ji} Y_{ki}$
	2-out-stars	$\sum_{i,j,k} Y_{ij} Y_{ik}$
	2-mixed-stars	$\sum_{i,j,k} Y_{ji} Y_{ik}$
<i>Subgroup effects</i>		$B^{rs} = \sum_{i,j} Y_{ij} I[i \in \text{rth subgroup}, j \in \text{sth subgroup}]$
<i>Individual level</i>		
	Differential expansiveness	$Y_{i+} = \text{outdegree (degree centrality)}$
	Differential attractiveness	$Y_{+i} = \text{indegree (degree prestige)}$

Hence the proposed ERG model can be written

$$P_{\boldsymbol{\eta}}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y})\}}{\kappa(\boldsymbol{\eta})} = \frac{\exp\{\eta_1 g_1(\mathbf{Y}) + \eta_2 g_2(\mathbf{Y}) + \eta_3 g_3(\mathbf{Y}) + \dots\}}{\kappa(\boldsymbol{\eta})} \quad (1.5)$$

with $g_1, g_2, g_3, \dots$, any of the statistics listed in Table 1.2.

1.2.6 Advanced model terms in ERGMs

All of the model configurations mentioned thus far have been shown to suffer from *model degeneracy*. Model degeneracy is a sign that the proposed model configuration is a poor explanation of the underlying social process that generated the observed network. Recent work by Handcock (2003) and Snijders et al. (2006) has addressed the degeneracy issue in many cases. This model improvement is gained by adding new higher order network terms to the model, terms which seem both in concept and practice to better capture the true nature of social tie formation. Some of these terms are described in Morris et al. (2007) and Snijders et al. (2006) and are shown in Table 1.3. One example is the Edgewise Shared Partner term, and its weighted summary, the Geometrically Weighted Edgewise Shared Partner (GWESP) term. The Edgewise Shared Partner statistics are similar to the usual degree distribution: this is the distribution of the number of edges in the graph with exactly each possible number of shared partners. The GWESP statistic is a geometrically weighted average of the edgewise shared partner distribution, with the weights determined by an additional “decay” parameter (Snijders et al., 2006; Hunter et al., 2007; Morris et al., 2007). In addition to mitigating the degeneracy issue and resulting in better model fit, these new model terms give us a new sense of how to quantify network structure. GWDEGREE is a weighted average of the degree distribution with a “decay” parameter controlling the weights. The version of the GWDEGREE listed in Table 1.3 is the version given in Hunter et al. (2007) rather than the version in Snijders et al. (2006). These geometrically weighted statistics somewhat mitigate the dependence in the statistics of the degree (or edgewise or dyadwise shared partner) distributions. This idea originated in this context in Snijders et al. (2006), who proposed the alternating k -star statistic to geometrically weight the sequence of k -stars in one summary statistic specifically to counteract the dependence in the k -star statistics. Intuitively, if there are many $(j + 1)$ -stars in the network, there are certainly many j -stars, as well. For example, every 4-star is composed of exactly four 3-stars, six 2-stars, and four edges (1-stars) (Hunter et al., 2008). The model with edges and the alternating k -star statistic is mathematically equivalent to the model with edges and GWDEGREE. In practice, these terms seem to capture the effect of these distributions well without incurring the

dependence issues inherent in them.

Table 1.3. New terms for ERG models.

Model Term	Graph Statistic ($\mathbf{g}(\mathbf{Y})$)
ESP	$EP_i(\mathbf{y})$ = the number of edges that serve as the common base for exactly i distinct triangles
GWESP	$u_E(\mathbf{y}; \theta_s) = e^{\theta_s} \sum_{i=1}^{n-2} \left\{ 1 - (1 - e^{\theta_s})^i \right\} EP_i(\mathbf{y})$
DSP	$DP_i(\mathbf{y})$ = the number of pairs (i, j) that share exactly i neighbors in common, whether or not (i, j) is an edge
GWDSP	$u_{\text{DSP}}(\mathbf{y}; \theta_s) = e^{\theta_s} \sum_{i=1}^{n-2} \left\{ 1 - (1 - e^{\theta_s})^i \right\} DP_i(\mathbf{y})$
DEGREE	$D_i(\mathbf{y})$ = the number of nodes with exactly i undirected edges
GWDEGREE	$u_{\text{Deg}}(\mathbf{y}; \theta_s) = e^{2\theta_s} \sum_{i=1}^{n-1} \left\{ (1 - e^{\theta_s})^i - 1 + ie^{-\theta_s} \right\} D_i(\mathbf{y})$
in-DEGREE	$ID_i(\mathbf{y})$ = the number of nodes receiving exactly i edges
GWIDEGREE	$u_{\text{Deg}}(\mathbf{y}; \theta_s) = e^{2\theta_s} \sum_{i=1}^{n-1} \left\{ (1 - e^{\theta_s})^i - 1 + ie^{-\theta_s} \right\} ID_i(\mathbf{y})$
out-DEGREE	$OD_i(\mathbf{y})$ = the number of nodes sending exactly i edges
GWODEGREE	$u_{\text{Deg}}(\mathbf{y}; \theta_s) = e^{2\theta_s} \sum_{i=1}^{n-1} \left\{ (1 - e^{\theta_s})^i - 1 + ie^{-\theta_s} \right\} OD_i(\mathbf{y})$

These advances are further described and illustrated in a special issue of *Social Networks*, in Hunter et al. (2007), Goodreau et al. (2008), Morris et al. (2007), and Handcock et al. (2007). One article in this special issue enumerates the use and benefits of these new terms on the ubiquitous AddHealth data (Goodreau, 2007). The National Longitudinal Study of Adolescent Health (AddHealth) study collected friendship data, among other information, on seventh through twelfth graders in a stratified sample of 86 school communities in the US. The friendship networks resulting from this study have nodes determined by the students in the school community, and directed edges originating from a node correspond to the students (up to five boys and five girls, possibly outside of the school) selected by that student as “close friends” (Resnick et al., 1997; Udry and Bearman, 1998). Goodreau (2007) demonstrates, on School 42 from the AddHealth data, that traditional Markov dependence models often result in degeneracy, whereas models based on a combination of exogenous (independent of the network) attributes

and endogenous (network structural, including these new geometrically-weighted model terms) characteristics seem to capture the observed network structure when assessed in comparison to higher-order network statistics in the generated graphs (Pattison and Robins, 2002; Robins et al., 2007b).

1.3 Inference for ERGMs

1.3.1 The loglikelihood function

As above, we define the class of exponential-family random graph models (ERGMs) for simple loopless, unweighted networks as follows:

$$P_{\boldsymbol{\eta}}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y})\}}{\kappa(\boldsymbol{\eta})}, \quad \mathbf{y} \in \mathcal{Y}, \quad (1.6)$$

where $\mathbf{Y}$ is a random network written as an adjacency matrix so that Y_{ij} is the indicator of an edge from i to j ; $\mathbf{g}(\mathbf{y})$ is a vector of the network statistics of interest; $\boldsymbol{\eta}$ is the vector of parameters measuring, in some sense, the strength of the effect of each corresponding entry in the vector $\mathbf{g}(\mathbf{y})$; $\kappa(\boldsymbol{\eta})$ is the constant of proportionality that makes the probabilities sum to one; and $\mathcal{Y}$ is the space of allowable networks.

For a particular ordered or unordered pair (i, j) , we denote by $\mathbf{Y}_{ij}^c$ all entries in $\mathbf{Y}$ *except for* Y_{ij} ; furthermore, $\mathbf{Y}_{ij}^+$ and $\mathbf{Y}_{ij}^-$ are the networks obtained from $\mathbf{Y}$ by replacing Y_{ij} by 1 and 0, respectively. Calculating the odds from (1.6) of setting a particular Y_{ij} to 1 or 0 conditional on the rest of the network $\mathbf{Y}_{ij}^c$, we obtain

$$\frac{P(Y_{ij} = 1 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)}{P(Y_{ij} = 0 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)} = \frac{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y}_{ij}^+)\}}{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y}_{ij}^-)\}} = \exp\{\boldsymbol{\eta}^t \boldsymbol{\delta}_g(\mathbf{y})_{ij}\}, \quad (1.7)$$

where $\boldsymbol{\delta}_g(\mathbf{y})_{ij}$ is the change in $\mathbf{g}(\mathbf{y})$ when y_{ij} is changed from 0 to 1. The vector $\boldsymbol{\delta}_g(\mathbf{y})_{ij}$ is called the *vector of change statistics* (Wasserman and Pattison, 1996).

The log-likelihood for this class of models is

$$\ell(\boldsymbol{\eta}) = \boldsymbol{\eta}^t \mathbf{g}(\mathbf{y}^{\text{obs}}) - \log \kappa(\boldsymbol{\eta}) = \boldsymbol{\eta}^t \mathbf{g}(\mathbf{y}^{\text{obs}}) - \log \sum_{\substack{\text{all possible} \\ \text{networks } \mathbf{y}}} \exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y})\}. \quad (1.8)$$

Maximum likelihood estimation of the parameters $\boldsymbol{\eta}$ can be a very difficult problem in some cases. It would be ideal to directly maximize the loglikelihood, but this requires knowledge of the normalizing constant, $\kappa(\boldsymbol{\eta})$, which is itself a sum involving the unknown parameter $\boldsymbol{\eta}$ over *all possible networks* with the same number of nodes as the observed graph. This number of networks can be quite large. For an undirected network of n nodes, there are $2^{\binom{n}{2}}$ possible configurations. For a directed network of the same size, there are $2^{2\binom{n}{2}}$ possible configurations.

1.3.2 Maximum pseudolikelihood estimation

If a particular ERGM implies that each Y_{ij} is independent of every other, we can avoid the trouble of the normalizing constant, $\kappa(\boldsymbol{\eta})$. From (1.7), conditional on $\mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c$, $\mathbf{Y}$ has only two possible states, depending on whether $Y_{ij} = 0$ or $Y_{ij} = 1$. The ratio of the two respective probabilities is

$$\frac{P(Y_{ij} = 1 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)}{P(Y_{ij} = 0 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)} = \frac{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y}_{ij}^+)\}}{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y}_{ij}^-)\}} = \exp\{\boldsymbol{\eta}^t [\mathbf{g}(\mathbf{y}_{ij}^+) - \mathbf{g}(\mathbf{y}_{ij}^-)]\}$$

So

$$\log \frac{P(Y_{ij} = 1 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)}{P(Y_{ij} = 0 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)} = \boldsymbol{\eta}^t \boldsymbol{\delta}_g(\mathbf{y})_{ij}$$

What if we approximate the conditional probability $P(Y_{ij} = 1 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)$ by the marginal probability $P(Y_{ij} = 1)$? Then the Y_{ij} are independent with

$$\log \frac{P(Y_{ij} = 1)}{P(Y_{ij} = 0)} = \boldsymbol{\eta}^t \boldsymbol{\delta}_g(\mathbf{y}^{\text{obs}})_{ij}, \text{ for all } (i, j),$$

and we may therefore obtain $\hat{\boldsymbol{\eta}}$ using logistic regression in which the response values are the y_{ij}^{obs} and the predictor vectors are the $\boldsymbol{\delta}$. The result is called the *maximum pseudolikelihood estimate*.

So, if the Y_{ij} really are independent, then the ERGM is exactly logistic regression and maximum pseudolikelihood estimation (MPLE) finds the maximum likelihood estimate (MLE). So long as the independence assumption (and therefore the logistic regression framework) is reasonable, the MPLE is reasonable. If, however, the statistics $\mathbf{g}(\mathbf{y})$ used in

the model imply independence, then the likelihood function being maximized by MPLE is not the correct likelihood function.

This MPLE method is introduced in Besag (1974) and Strauss and Ikeda (1990), and is recommended as the method for parameter estimation for p^* models in Wasserman and Pattison (1996). The many well-known and well-studied properties of the MLE (consistency, asymptotic efficiency, etc.) do not apply to the MPLE, however, and this weakness is a cause for concern in using the MPLE. In fact, a simulation study in van Duijn et al. (2009) suggests that, in some cases, the MPLE can be a badly biased and inconsistent estimator.

1.3.3 Statistical inference in Markov graphs

The more complicated dependence introduced in Frank and Strauss (1986) required inference methodology beyond the logistic-regression-type analysis that is appropriate to dyad-independent data. To deal with this Markovian dependence, Frank and Strauss (1986) suggested several possible inference methodologies.

The clustering model is an excellent motivating example for the inference developed in Frank and Strauss (1986). The probability distribution function for the Markov clustering model is $P(\mathbf{Y} = \mathbf{y}) = \exp(\sigma s(\mathbf{Y}))/\kappa(\sigma)$, where $s(\mathbf{Y})$ is the number of 2-stars in the network, Y . The cumulant generating function of the statistic $s(\mathbf{Y})$ is $K_\sigma(t) = \log \kappa(t + \sigma) - \log \kappa(\sigma)$.

We want to find the maximum likelihood estimator $\hat{\sigma}$, which is a root of $s(\mathbf{Y}) = (d/d\sigma) \log \kappa(\sigma) = E_\sigma(s(\mathbf{Y}))$ for this model. Because we cannot evaluate κ and its derivatives by direct enumeration, and it is generally impossible to calculate $\hat{\sigma}$ exactly from the cumulants of $s(\mathbf{Y})$, as $\kappa(\sigma)$ involves all of them, an alternative is to generate many simulated graphs of various σ and find the average number of 2-stars for graphs with the same number of vertices and edges as the data.

We can also approximate $\kappa(\sigma)$ by dropping off terms of order σ^3 or higher so that $\tilde{\sigma} = (s(\mathbf{Y}) - \kappa_1)/\kappa_2$, where κ_1 is the first cumulant of $s(\mathbf{Y})$, its mean, and κ_2 is the second cumulant of $s(\mathbf{Y})$, its variance. Frank and Strauss (1986) finds that the bias here is serious for $|\sigma| > .2$.

Another approach to estimating $\hat{\sigma}$, the MLE, taken by Frank and Strauss is to use logistic regression of Y_{ij} on σ_{ij} , i.e. MPLE, despite the fact that MPLE relies on the assumption that the dyads are independent, an assumption not reflected in the model. The paper claims this is a promising method; however, we know from more recent studies (van Duijn et al., 2009) that the MPLE can perform very badly in general.

Frank and Strauss (1986) showed results based on the MPLE for the general triad model (discussed in section 1.2.4) in the Sampson monastery data (Sampson, 1968). This is a classic data set in social network analysis, and is included in, for example, the **ergm** package (Handcock et al., 2010) in R. The data are the friendships among novices in a New England monastery, studied in the doctoral thesis of F. Sampson. For model (1.4), the MPLE fit is $\hat{\rho} = -.10$ for the edge (or density) coefficient, $\hat{\sigma} = -.23$ for the clustering coefficient, and $\hat{\tau} = .97$ for the transitivity coefficient. The large value for $\hat{\tau}$ suggests a tendency toward transitivity of positive influence. To check the fit, the authors predicted the edges according to “Edges in $\mathbf{Y}$ ” = $\hat{\rho} + \hat{\sigma}s(\mathbf{Y}) + \hat{\tau}t(\mathbf{Y})$ and found that 119 of the 153 values are correctly predicted, compared to 92 of the 153 if $\sigma = \tau = 0$, i.e. if we have a Bernoulli graph. Thus, compared to a Bernoulli model, the odds of correct prediction, evaluated for the MPLE fit, increase from $92/61 = 1.5$ to $119/34 = 3.5$ by including $s(\mathbf{Y})$ and $t(\mathbf{Y})$ in the model.

1.3.4 MCMC-based estimators

In general, real-world social networks do not exhibit dyadic independence. For example, consider the triad closure from the *transitivity* models: if A and B have a connection, and A and C have a connection, then there might be an extra propensity for B and C to connect and “complete the triangle.” This triad-closure propensity is often described by the phrase “the friend of my friend is my friend.” This is an example of dyadic dependence (as the probability of a tie between B and C is not independent of the other ties involving B and C).

Because of the effect we believe network dependency will have on the occurrence of edges in the network, we are particularly interested in studying the network-dependent statistics. For this reason we need to include these terms in $\mathbf{g}(\mathbf{y})$, as discussed in the

previous section. Once we attempt to account for this dependence, we can no longer estimate the model using simple logistic regression. Instead, we can use Monte Carlo MLE to estimate the parameters in the model. This idea was first introduced in this context by Geyer and Thompson (1992), building on the work of Besag (1974), Holland and Leinhardt (1981), and Frank and Strauss (1986), and it is made explicit for ERGMs in Hunter and Handcock (2006).

Equation (1.7) may be used to define a Markov chain with stationary distribution (1.6), as explained in, for example, Snijders (2002), and to generate a sample from this distribution. By taking the sample mean of this Markov chain sample, an estimate of the log-likelihood-ratio $\ell(\boldsymbol{\eta}) - \ell(\boldsymbol{\eta}_0)$ can be made, and its maximizer, the Markov chain Monte Carlo (MCMC) MLE, can be found. This MCMC ML estimation is explained in much greater detail in Chapter 2, and the MCMC MLE is the default estimator for maximum likelihood in ERGMs in the `statnet` package (Handcock et al., 2003) in R (R Development Core Team, 2010). Corander et al. (2002) and Crouch and Wasserman (1998) also implement MCMC MLE approximations based on the Geyer and Thompson (1992) idea.

The software package SIENA (Snijders et al., 2007b) and its platform StOCNET (Boer et al., 2003) use a slightly different idea to obtain a method of moments estimator, though this method too is based on MCMC. The method of moments, using a Robbins-Monro algorithm (Robbins and Monro, 1951), is described in Snijders (2001) and Snijders et al. (2007c). This algorithm, which uses MCMC to find an approximate method of moments estimator (MME) differs from an MCMC MLE method in that the MCMC sample is used to estimate the moments of the distribution rather than to estimate the loglikelihood function itself. Nevertheless, for an exponential family, the MCMC MLE is also the MCMC MME, and both can be found in the `ergm` package (Handcock et al., 2010) for R.

A third method for estimating the parameter vector (which will not be considered in this thesis) can be found using Bayesian methods, as in Koskinen (2004), Koskinen and Snijders (2007), and Schweinberger and Snijders (2007).

1.4 Recent related advances in ERG modeling

1.4.1 Missing data

Another important area of social network analysis is handling missing data in this context. In social network analysis, we observe only one realization of a network process. We call this the *observed network*. The dyads (pairs of actors) that comprise this network are the population of interest. If we can observe the information of interest on each dyad pair, then we can observe the entire population. Far more often, however, we can only observe a subset of the network. For example, we might be able to survey only a subset of the actors. In this case, we would be missing the identities and attributes of the actors not surveyed, the ties among the un-surveyed actors, and the ties between the surveyed and un-surveyed actors. As mentioned in Handcock and Gile (2010), missing data in social network analysis is similar to missing data in time series analysis in that, rather than missing independent observations replicated from the process of interest, “we are missing parts of a single realization from a dependent process.” More information on the state of the art of missing data research can be found in Gile and Handcock (2006) and Handcock and Gile (2010).

1.4.2 Network sampling

Related to the general missing data problem is the idea that an observed network is only a sample from the larger network. This concept is intuitively basic, but has been slow-moving in its theoretical statistical development as methods for treating the observed network as a realization of a complete network (the easier question) have themselves been slow in appearing in the literature. Thompson (2006) and the references therein illustrate many of the recent developments and the current state of network sampling research.

1.5 Summary of this work

In Chapter 2, details on the computation of the approximate MLE are given, and an algorithm is introduced to improve the estimation of the MLE in current computational software. Chapter 3 explores the performance of several closely-related *alternatives to the MLE*, including the MPLE and a *bias-adjusted* version of the MLE, in a variety of settings. In Chapter 4, the method of *contrastive divergence*, from the field of machine learning, is compared to maximum likelihood for network data, and a new class of estimators, the maximizer of the *composite likelihood*, is introduced. Contrastive divergence for composite likelihoods is demonstrated in ERGMs.

A steplength algorithm for fitting ERGMs

2.1 Introduction

Networks are a form of relational data, often represented as a set of pairs of individuals, called nodes, where each pair may be directed or undirected and is called an arc or an edge, respectively. Examples of networks include disease contact networks, where the nodes are individuals and the edges indicate some type of physical contact, and friendship networks, where the nodes are school students and an arc indicates that one student considers another to be a friend. with a directed edge from student A to student B indicating that A considers B to be a friend. These two examples are far from exhaustive; networks can involve all kinds of individuals, groups, organizations, objects, or concepts, and the scientific literature on networks is massive and growing quickly.

This chapter focuses on maximum likelihood estimation for a class of statistical network models called exponential-family random graph models (ERGMs). A large literature on ERGMs exists; for an introduction, we refer the interested reader to the survey article of Robins et al. (2007a) and the many references therein. Here, we present only the basic mathematical groundwork necessary to make this discussion self-contained.

We define the class of ERGMs for simple (no repeated edges), loopless (no self-edges)

networks as follows:

$$P_{\boldsymbol{\eta}}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y})\}}{\kappa(\boldsymbol{\eta})}, \quad \mathbf{y} \in \mathcal{Y}, \quad (2.1)$$

where $\mathbf{Y}$ is a random network written as an adjacency matrix so that Y_{ij} is the indicator of an edge from node i to node j ; $\mathbf{g}(\mathbf{y})$ is a vector of the network statistics of interest, which may include covariates that are measured on the nodes; $\boldsymbol{\eta}$ is the vector of parameters measuring, in some sense, the strengths of the effects of the corresponding entries in the vector $\mathbf{g}(\mathbf{y})$; $\kappa(\boldsymbol{\eta})$ is the constant of proportionality that makes the probabilities sum to one; and $\mathcal{Y}$ is the space of allowable networks. Typically, $\mathcal{Y}$ includes only networks on a *specific* set of nodes; in particular, this means that each element of $\mathcal{Y}$ typically has the same number of nodes. Letting n denote this number, we may always refer to $\mathbf{Y}$ as an $n \times n$ matrix of zeros and ones. For simplicity, we assume throughout this chapter that $\mathbf{Y}$ is undirected; that is, $Y_{ij} = Y_{ji}$, and so we will use the convention that $i < j$ whenever we write Y_{ij} .

Given an observed network $\mathbf{y}^{\text{obs}}$, the log-likelihood function

$$\ell(\boldsymbol{\eta}) = \boldsymbol{\eta}^\top \mathbf{g}(\mathbf{y}^{\text{obs}}) - \log \kappa(\boldsymbol{\eta}) \quad (2.2)$$

is, in general, very difficult to maximize due to the intractability of $\kappa(\boldsymbol{\eta})$, which is evidently equal to the sum of $\exp\{\boldsymbol{\eta}^t \mathbf{g}(\mathbf{y})\}$ for all possible networks $\mathbf{y}$. Since there may be $2^{\binom{n}{2}}$ such networks even in the undirected case, $\kappa(\boldsymbol{\eta})$ is not computable directly unless some mathematical simplification is possible. Nevertheless, approximate maximum likelihood may be accomplished using a simulation-based method as described by Geyer and Thompson (1992). Typically, such methods rely on Markov chain Monte Carlo (MCMC) to simulate random networks. We describe a standard MCMC algorithm and explain how approximate maximum likelihood estimation (MLE) may proceed in Section 2.2.

In this chapter we suggest two qualitatively distinct improvements to a standard MCMC MLE algorithm as described in Geyer and Thompson (1992). First, the standard approximation is replaced by a lognormal-based approximation. Second, we describe a “stepping” algorithm that moves step-by-step toward a maximizer of the loglikelihood.

Though our improvements are still MCMC MLE algorithms and therefore rely on randomly simulated networks, we stress that this chapter does not discuss improvements to the MCMC sampler used for this purpose. In fact, as we demonstrate in Section 2.3, even a perfectly efficient sampler applied to the simplest of all network models cannot overcome the flaws inherent in the standard MCMC MLE method.

After describing the basic MCMC MLE idea and introducing a lognormal approximation to the loglikelihood (Section 2.2), we illustrate a major downfall of the basic method using a simple network model in Section 2.3. We then introduce a new “stepping” algorithm in Section 2.4 and discuss standard error estimation in Section 2.5. Finally, we demonstrate our methods on both the simplistic model used earlier (Section 2.6) and two network datasets from the literature (Section 2.7).

2.2 Approximate maximum likelihood estimation methods

Any approximate MLE method based on simulation relies on some form of random sampling of networks. Typically, this sampling is accomplished via MCMC based on the following derivation. For a particular ordered pair (i, j) , we denote by $\mathbf{Y}_{ij}^c$ all entries in $\mathbf{Y}$ *except for* Y_{ij} . Furthermore, $\mathbf{Y}_{ij}^+$ and $\mathbf{Y}_{ij}^-$ are the networks obtained from $\mathbf{Y}$ by replacing Y_{ij} by 1 and 0, respectively. Calculating the odds from (2.1) of setting a particular Y_{ij} to 1 or 0 conditional on the rest of the network $\mathbf{Y}_{ij}^c$, we obtain

$$\frac{P_{\boldsymbol{\eta}}(Y_{ij} = 1 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)}{P_{\boldsymbol{\eta}}(Y_{ij} = 0 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)} = \frac{\exp\{\boldsymbol{\eta}^\top \mathbf{g}(\mathbf{y}_{ij}^+)\}}{\exp\{\boldsymbol{\eta}^\top \mathbf{g}(\mathbf{y}_{ij}^-)\}} = \exp\{\boldsymbol{\eta}^\top \boldsymbol{\delta}_g(\mathbf{y})_{ij}\}, \quad (2.3)$$

where $\boldsymbol{\delta}_g(\mathbf{y})_{ij} = \mathbf{g}(\mathbf{y}_{ij}^+) - \mathbf{g}(\mathbf{y}_{ij}^-)$ denotes the change in $\mathbf{g}(\mathbf{y})$ when y_{ij} is changed from 0 to 1 and is sometimes called the *vector of change statistics* (Wasserman and Pattison, 1996). Equation (2.3) may be exploited to define a Markov chain with stationary distribution (2.1); the particular MCMC sampler we use is the one described by Hunter et al. (2007) and implemented in the `ergm` package (Handcock et al., 2010) for R (R Development Core Team, 2010).

For certain choices of the $\mathbf{g}(\mathbf{y})$ vector, the indicators Y_{ij} are independent; that is,

$P(Y_{ij} = 1) = P(Y_{ij} = 1 | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c)$ for all $i < j$. In such cases, we can estimate $\boldsymbol{\eta}$ simply by logistic regression with odds as defined in (2.3); that is, the vector of predictors for response Y_{ij} is $\boldsymbol{\delta}_g(\mathbf{y}^{\text{obs}})_{ij}$ for each $i < j$. The resulting log-likelihood in this special case can be written as

$$\ell(\boldsymbol{\eta}) = \sum_{i < j} \sum \left[y_{ij}^{\text{obs}} \boldsymbol{\eta}^\top \boldsymbol{\delta}_g(\mathbf{y}^{\text{obs}})_{ij} - \log(1 + \exp\{\boldsymbol{\eta}^\top \boldsymbol{\delta}_g(\mathbf{y}^{\text{obs}})_{ij}\}) \right]. \quad (2.4)$$

When the Y_{ij} are not independent, formula (2.4) is no longer the log-likelihood, but is called the (logarithm of the) *pseudolikelihood*, and its maximizer is the *Maximum Pseudolikelihood Estimator*, or MPLE. As discussed in Geyer and Thompson (1992), Robins et al. (2007a), and van Duijn et al. (2009), the MPLE can perform badly in some cases.

Since the behavior of the MPLE is unpredictable, it is preferable to maximize the true log-likelihood function (2.2) rather than the pseudolikelihood function. This is rarely an option in practice; for a network of more than a few nodes, the mere evaluation of the normalizing constant $\kappa(\boldsymbol{\eta})$ is computationally intractable, and direct maximization over $\boldsymbol{\eta}$ is therefore impossible. An indirect alternative arises if we fix an arbitrary parameter value $\boldsymbol{\eta}_0$ and note that for all $\boldsymbol{\eta}$,

$$\ell(\boldsymbol{\eta}) - \ell(\boldsymbol{\eta}_0) = (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{y}^{\text{obs}}) - \log E_{\boldsymbol{\eta}_0} \left[\exp \left\{ (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{Y}) \right\} \right], \quad (2.5)$$

where $E_{\boldsymbol{\eta}_0}$ denotes expectation with respect to the mass function (2.1) with $\boldsymbol{\eta} = \boldsymbol{\eta}_0$. A technical detail that we do not discuss here is the fact that when $\mathbf{g}(\mathbf{y}^{\text{obs}})$ happens to lie on the boundary of the convex hull of the set $\{\mathbf{g}(\mathbf{y}) : \mathbf{y} \in \mathcal{Y}\}$ of all possible statistics, the MLE does not exist. This phenomenon is quite well-understood and is described in detail by Rinaldo et al. (2009) and Geyer (2009). It is qualitatively different from the vexing issue of model degeneracy, which we address below.

As first suggested in Geyer and Thompson (1992) and later developed for ERGMs by Snijders (2002), Robins and Pattison (2005), and Hunter and Handcock (2006), we can exploit equation (2.5) by randomly sampling networks $Y_1, \dots, Y_m$ from $P_{\boldsymbol{\eta}_0}$ and

approximating

$$\ell(\boldsymbol{\eta}) - \ell(\boldsymbol{\eta}_0) \approx (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{y}^{\text{obs}}) - \log \left[\frac{1}{m} \sum_{i=1}^m \exp \left\{ (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{Y}_i) \right\} \right]. \quad (2.6)$$

Iterating an approximate Fisher scoring method (Hunter and Handcock, 2006) until convergence gives a maximizer of the approximate log-likelihood ratio (2.6). We refer to Equation (2.6) is the “naive” approximation since it simply approximates a population mean by the sample mean. As Geyer and Thompson (1992) points out, this approximation is ineffective except for $\boldsymbol{\eta}$ close to $\boldsymbol{\eta}_0$. We demonstrate in Section 2.3 how ineffective it can be even for a simplistic network.

One possibility for improving the approximation in Equation (2.6) arises from a loose distributional assumption that is reasonable in many cases. Suppose $Z = (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{Y})$ is approximately normally distributed with mean $\mu = (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{m}_0$ and variance $\sigma^2 = (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \boldsymbol{\Sigma}_0 (\boldsymbol{\eta} - \boldsymbol{\eta}_0)$, where $\mathbf{m}_0$ and $\boldsymbol{\Sigma}_0$ are the mean vector and covariance matrix of $\mathbf{g}(\mathbf{Y})$ under $P_{\boldsymbol{\eta}_0}$. Then $\exp(Z)$ is lognormally distributed and $\log E_{\boldsymbol{\eta}}(\exp(Z)) = \mu + \sigma^2/2$, which suggests

$$\ell(\boldsymbol{\eta}) - \ell(\boldsymbol{\eta}_0) \approx (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \left[\mathbf{g}(\mathbf{y}^{\text{obs}}) - \hat{\mathbf{m}}_0 \right] - \frac{1}{2} (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \hat{\boldsymbol{\Sigma}}_0 (\boldsymbol{\eta} - \boldsymbol{\eta}_0), \quad (2.7)$$

which has maximizer $\boldsymbol{\eta}_0 + \hat{\boldsymbol{\Sigma}}_0^{-1} [\mathbf{g}(\mathbf{y}^{\text{obs}}) - \hat{\mathbf{m}}_0]$. In Equation (2.7), we take $\hat{\mathbf{m}}_0$ and $\hat{\boldsymbol{\Sigma}}_0$ to be the usual sample estimators of $\mathbf{m}_0$ and $\boldsymbol{\Sigma}_0$. Alternatively, we could use robust estimators to approximate $\log E_{\boldsymbol{\eta}}(\exp(Z))$ based on $\hat{\mu} = \text{median}_{1 \leq i \leq m} (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{Y}_i)$ and

$$\hat{\sigma}^2 = c \times \text{median}_{1 \leq i \leq m} \left| (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{Y}_i) - \hat{\mu} \right|,$$

where c is some appropriately chosen constant such as 1.483, the reciprocal median absolute value of a standard normal variable.

Regardless of the method of approximation used, the MCMC MLE approach gives values of $\ell(\boldsymbol{\eta}) - \ell(\boldsymbol{\eta}_0)$ that are inaccurate unless $\boldsymbol{\eta}$ is near $\boldsymbol{\eta}_0$, so using the maximizer of the approximation as an approximant of the true MLE $\hat{\boldsymbol{\eta}}$ can be perilous if $\boldsymbol{\eta}_0$ lies far from $\hat{\boldsymbol{\eta}}$. In Section 2.4 we introduce a systematic method for moving closer to $\hat{\boldsymbol{\eta}}$ step by

step.

2.3 Example: The Gilbert–Erdős–Rényi model

Proposed as early as Rapoport (1953), then later described independently by Gilbert (1959) and Erdős and Rényi (1959), this model assumes that all Y_{ij} are independent Bernoulli random variables with common parameter b , so that the total number of edges in the network is binomial with parameters $N = \binom{n}{2}$ and b .

The Gilbert–Erdős–Rényi model can be written as an ERGM by defining the scalar $g(\mathbf{y})$ to be equal the number of edges of $\mathbf{y}$. Since each edge exists independently with probability b ,

$$P(\mathbf{Y} = \mathbf{y}) = (b)^{g(\mathbf{y})} (1 - b)^{N - g(\mathbf{y})} = \left(\frac{b}{1 - b} \right)^{g(\mathbf{y})} (1 - b)^N. \quad (2.8)$$

Setting $e^\eta = b/(1 - b)$, Equations (2.8) and (2.1) match with $\kappa(\eta) = (1 + e^\eta)^{-N}$. To apply the simulation-based likelihood approximation methods of Section 2.2, we simply simulate $g(\mathbf{y}_1), \dots, g(\mathbf{y}_m)$ as independent binomial random variables with parameters N and $e^{\eta_0}/(1 + e^{\eta_0})$.

As an example, we consider a simulated 40-node network that has 272 edges out of a possible $N = 780$ dyads. The MLE for η in this case is $\hat{\eta} = \log(272/508) = -0.625$. In

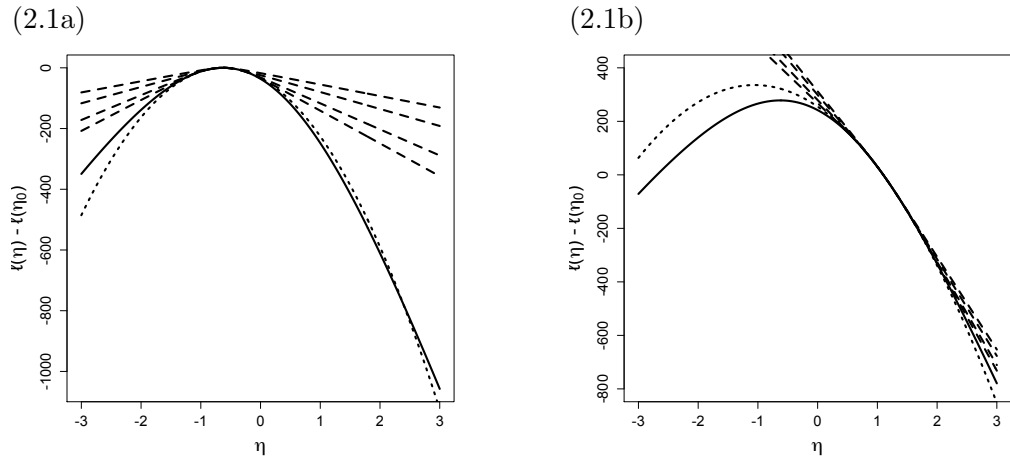


Figure 2.1. Approximated values of $\ell(\boldsymbol{\eta}) - \ell(\boldsymbol{\eta}_0)$. The solid line is the true loglikelihood (2.5), maximized at $\hat{\eta} = -0.625$. The dashed lines are equation (2.6) with idealized sample sizes 10^3 , 10^5 , 10^{10} , and 10^{15} . The dotted line is equation (2.7).

our example, we may rewrite Equation (2.5) as

$$E_{\eta_0} [\exp \{(\eta - \eta_0)g(\mathbf{Y})\}] = \sum_{j=0}^{780} p_j \exp \{j(\eta - \eta_0)\},$$

where the p_j are simply the usual binomial probabilities. We may approximate (2.5) using the idea of Equation (2.6) but with an idealized sample of size m in which p_j is replaced by the closest multiple of $1/m$ to p_j . Figure 2.1 depicts the resulting approximations to (2.5) using various values of η_0 and m . Of particular note is that even for incredibly large m , such as $m = 10^{15}$, the approximation to (2.5) is not good for η far from η_0 . Also of note is that in Figure 2.1b, even the enormous $m = 10^{15}$ yields a sample consisting only of values larger than $g(\mathbf{y}^{\text{obs}}) = 272$, so Equation (2.6) has no maximizer. Indeed, with $\eta_0 = 1.099$, a value like 272 is more than 25 standard deviations below the mean, so such an observation is exceptionally rare.

2.4 Partial stepping

The original MCMC MLE article by Geyer and Thompson (1992) recognized that “The theory of importance sampling leads to the conclusion that the best results are obtained when $\boldsymbol{\eta}_0$ is near $\hat{\boldsymbol{\eta}}$.” (We have substituted our notation for theirs in the quotation.) Indeed, as demonstrated in Section 2.3 for even the simplest of exponential family models, the “arbitrary” value $\boldsymbol{\eta}_0$ of equation (2.5) is not quite so arbitrary: The quality of approximations (2.6) and (2.7) degrades quickly as $\boldsymbol{\eta}$ moves away from $\boldsymbol{\eta}_0$. As a remedy for this problem, Geyer and Thompson (1992) recommended maximizing the approximate loglikelihood subject to a restriction on the maximum allowable step, e.g., subject to $\|\boldsymbol{\eta} - \boldsymbol{\eta}_0\| \leq \delta$. However, they offered no guidance on how to choose an appropriate δ . In one numerical example, they state that all steps were constrained to a maximum step size of $\delta = \sqrt{2}$; however, they state that this value was “chosen arbitrarily, though several other runs showed it to be about the right size.” We do not use constrained optimization using a value of δ as suggested by Geyer and Thompson (1992). The alternative method that we introduce here is also a systematic method for moving $\boldsymbol{\eta}_0$ closer to $\hat{\boldsymbol{\eta}}$ step by

step, but it avoids the difficult problem of how to choose the δ parameter.

To explain our method, we first review a bit of exponential family theory. In common statistical parlance, the original ERGM parameterization (2.1), with $\boldsymbol{\eta}$ as parameter, is called the *canonical parameterization*. Standard theory (Barndorff-Nielsen, 1978; Brown, 1986) states that the MLE, if it exists, is the unique parameter vector $\hat{\boldsymbol{\eta}}$ such that $E_{\hat{\boldsymbol{\eta}}} \mathbf{g}(\mathbf{Y}) = \mathbf{g}(\mathbf{y}^{\text{obs}})$. The so-called *mean value parameterization* is defined by $\boldsymbol{\xi}(\boldsymbol{\eta}) = E_{\boldsymbol{\eta}}(\mathbf{g}(\mathbf{Y}))$. Using the $\boldsymbol{\xi}$ parameters, calculating the maximum likelihood estimate is trivially easy: $\hat{\boldsymbol{\xi}} = \mathbf{g}(\mathbf{y}^{\text{obs}})$. However, this fact is of little practical value, since we cannot estimate standard errors or even simulate from the fitted model without knowing the inverse image $\hat{\boldsymbol{\eta}}$ of $\hat{\boldsymbol{\xi}}$.

Our idea is to take partial steps toward $\hat{\boldsymbol{\xi}}$ in mean value parameter space by pretending that the MLE $\hat{\boldsymbol{\xi}}$ is not $\mathbf{g}(\mathbf{y}^{\text{obs}})$ but rather some point in between $\mathbf{g}(\mathbf{y}^{\text{obs}})$ and our estimate of $\boldsymbol{\xi}(\boldsymbol{\eta}_0)$. This allows us to restrict our search for a maximizer of the approximate log-likelihood ratio to a region where this approximation is reasonably accurate. Conceptually, we will iteratively 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 we obtain a value of $\boldsymbol{\eta}_0$ close enough to $\hat{\boldsymbol{\eta}}$ to allow one final MCMC-based maximization step.

The procedure begins with an initial $\boldsymbol{\eta}_0$, such as $\boldsymbol{\eta}_0 = \mathbf{0}$ or $\boldsymbol{\eta}_0 = \text{MPLE}$. We then let $\bar{\boldsymbol{\xi}}_0$ denote the sample mean of the vectors $\mathbf{g}(\mathbf{Y}_1), \dots, \mathbf{g}(\mathbf{Y}_m)$, where $\mathbf{Y}_1, \dots, \mathbf{Y}_m$ is an MCMC sample from the model defined by $\boldsymbol{\eta}_0$, obtained using a procedure as described in Snijders (2002) or Hunter and Handcock (2006). Next, define the vector

$$\hat{\boldsymbol{\xi}}_1 = \gamma \hat{\boldsymbol{\xi}} + (1 - \gamma) \bar{\boldsymbol{\xi}}_0 \quad \text{for some } \gamma \in [0, 1]. \quad (2.9)$$

The idea is to treat $\hat{\boldsymbol{\xi}}_1$, rather than $\hat{\boldsymbol{\xi}}$, as the MLE in our maximization step because the former is closer to $\bar{\boldsymbol{\xi}}_0$ than the latter, which means that the approximation to the log-likelihood ratio should be better. At the very least, γ should be chosen so that $\hat{\boldsymbol{\xi}}_1$ is contained in the interior of the convex hull of the sampled points $\mathbf{g}(\mathbf{Y}_1), \dots, \mathbf{g}(\mathbf{Y}_m)$, particularly if we plan to use equation (2.6) to approximate the loglikelihood ratio. The

maximizer of the resulting approximation will be called $\boldsymbol{\eta}_1$. Since $\boldsymbol{\eta}_1$ is thus “closer” to the MLE than $\boldsymbol{\eta}_0$, in the sense that its corresponding mean-value parameter is closer, we then repeat the process, with $\boldsymbol{\eta}_1$ taking the place of $\boldsymbol{\eta}_0$. Eventually, when $\hat{\boldsymbol{\xi}}$ is in the convex hull, we may take the maximizer to be the final MLE or, alternatively, simulate a much larger sample in one final iteration.

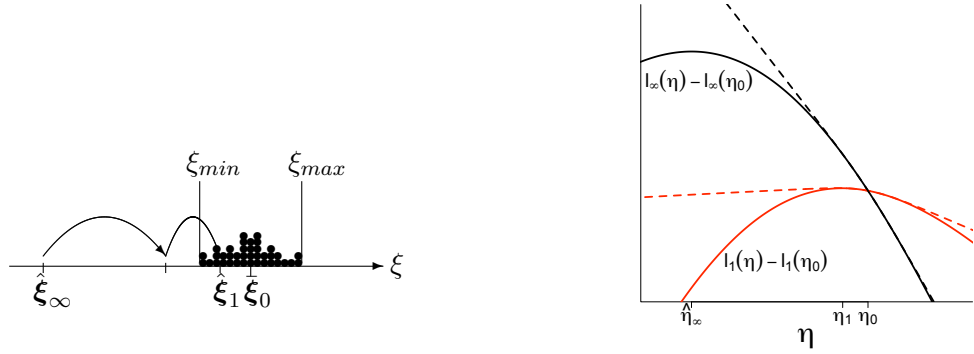


Figure 2.2. On the left, the boundaries of the convex hull of the MCMC sample statistics (the dots) are represented by dotted lines. Since the true MLE $\hat{\boldsymbol{\xi}}_\infty = \hat{\boldsymbol{\xi}} = \mathbf{g}(\mathbf{y}^{\text{obs}})$ is not inside the boundaries, we apply step-halving two times to find a value $\hat{\boldsymbol{\xi}}_1$ that is inside. On the right, the true log-likelihood ratio $l_\infty(\boldsymbol{\eta}) - l_\infty(\boldsymbol{\eta}_0)$ is not well-approximated, particularly at $\hat{\boldsymbol{\eta}}_\infty$. However, substituting $\hat{\boldsymbol{\xi}}_1$ for $\mathbf{g}(\mathbf{y}^{\text{obs}})$ yields a new log-likelihood ratio, also shown here, that is much better approximated in the neighborhood of its maximum.

As stated earlier, the choice of γ in Equation (2.9) should ensure that $\hat{\boldsymbol{\xi}}_1$ is inside the convex hull of the sampled statistics. Thus, implementation of the stepping algorithm requires some means for checking this condition. The question of whether the vector $\hat{\boldsymbol{\xi}}_t$ is in the convex hull of the sample of vectors $\mathbf{g}(\mathbf{y}_1), \dots, \mathbf{g}(\mathbf{y}_m)$ may be expressed as a linear programming problem (Fukuda, 2004):

$$\text{maximize } \mathbf{z}^\top \hat{\boldsymbol{\xi}}_t - z_0 \text{ subject to } \mathbf{z}^\top \mathbf{g}(\mathbf{y}_i) - z_0 \leq 0 \text{ for all } i = 1, \dots, m. \quad (2.10)$$

Note that the maximum is always at least zero, since $(\mathbf{z} = \mathbf{0}, z_0 = 0)$ always satisfies the constraints. But if the maximum is strictly positive, then the hyperplane $\{\mathbf{x} : \mathbf{z}^\top \mathbf{x} = z_0\}$ separates the sample points $\mathbf{g}(\mathbf{y}_i)$ from the point $\hat{\boldsymbol{\xi}}_t$, so $\hat{\boldsymbol{\xi}}_t$ cannot be in the convex hull of the sample points. We implement this procedure in the R package `ergm` using the R interface to the GNU Linear Programming Kit package (Theussl and Hornik, 2010; Makhorn, 2006). The computer code for the examples is available upon request.

Note that we do not want the largest possible γ such that the maximizer in (2.10) is zero, since this actually puts $\hat{\boldsymbol{\xi}}_t$ exactly on the boundary of the convex hull. Instead, in practice, we first identify the largest such γ among some discrete subset of $[0, 1]$ —such as $\{2^{-\nu} : \nu = 0, 1, 2, \dots\}$ or $\{x/a : x = 1, \dots, a\}$ —and then multiply by some constant slightly less than unity (in our examples, we use 0.95).

2.5 Estimating Standard Errors

Standard error estimates for the approximate maximum likelihood estimator are necessary for performing statistical inference. Here, the error has two components: first is the usual error due to the sampling distribution, or the error inherent in estimating the true $\boldsymbol{\eta}$ by a maximizer of the likelihood function, $\hat{\boldsymbol{\eta}}$; and second is the MCMC error incurred when we approximate the true log-likelihood function using an equation such as (2.6) or (2.7) and estimate the true MLE, $\hat{\boldsymbol{\eta}}$, by an approximate MLE, which we denote here by $\tilde{\boldsymbol{\eta}}$ to distinguish it from the true MLE. Using standard asymptotic theory, we estimate and invert the Fisher information matrix to estimate the sampling distribution error. The MCMC error is approximated as in Hunter and Handcock (2006); details are provided below.

For an exponential family, the Fisher information, $\mathbf{I}(\boldsymbol{\eta})$, is given by $\text{Var}_{\boldsymbol{\eta}} \mathbf{g}(\mathbf{Y})$. Furthermore, the variance of the sample statistics under $\boldsymbol{\eta}$ can be written $\text{Var}_{\boldsymbol{\eta}} \mathbf{g}(\mathbf{Y}) = \nabla^2 \log \kappa(\boldsymbol{\eta})$, where $\kappa(\boldsymbol{\eta})$ is the normalizing constant of equation (2.1). Hence, the inverse of the usual error covariance matrix for the sampling error is given by

$$\mathbf{I}(\hat{\boldsymbol{\eta}}) = \text{Var}_{\hat{\boldsymbol{\eta}}}(\mathbf{g}(\mathbf{Y})) = \nabla^2 \log E_{\boldsymbol{\eta}_0} \left[\exp \left\{ (\boldsymbol{\eta} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{Y}) \right\} \right] \Big|_{\boldsymbol{\eta} = \hat{\boldsymbol{\eta}}}. \quad (2.11)$$

Using the naive approximation for the required expectation as in (2.6), we obtain

$$\mathbf{I}(\hat{\boldsymbol{\eta}}) \approx \hat{\mathbf{I}}(\tilde{\boldsymbol{\eta}}) \stackrel{\text{def}}{=} \sum_{i=1}^m w_i \mathbf{g}(\mathbf{y}_i) \mathbf{g}^\top(\mathbf{y}_i) - \left[\sum_{i=1}^m w_i \mathbf{g}(\mathbf{y}_i) \right] \left[\sum_{i=1}^m w_i \mathbf{g}(\mathbf{y}_i) \right]^\top, \quad (2.12)$$

where the weights $w_i \propto \exp\{(\tilde{\boldsymbol{\eta}} - \boldsymbol{\eta}_0)^\top \mathbf{g}(\mathbf{y}_i)\}$ are normalized so that $\sum_i w_i = 1$.

If we use the lognormal approximation as in (2.7), then simply $\text{Var}_{\boldsymbol{\eta}} \mathbf{g}(\mathbf{Y}) \approx \boldsymbol{\Sigma}_0$, the

covariance for $\mathbf{g}(\mathbf{Y})$, which depends on $\boldsymbol{\eta}_0$ but not $\boldsymbol{\eta}$. This underscores the importance of choosing $\boldsymbol{\eta}_0$ close to $\hat{\boldsymbol{\eta}}$, since both approximations to $\ell(\hat{\boldsymbol{\eta}}) - \ell(\boldsymbol{\eta}_0)$ are best when $\boldsymbol{\eta}_0 = \hat{\boldsymbol{\eta}}$.

The approximate MCMC standard errors may be derived by first Taylor-expanding the approximate gradient around the point $\tilde{\boldsymbol{\eta}}$:

$$\nabla \hat{\ell}(\hat{\boldsymbol{\eta}}) \approx \nabla \hat{\ell}(\tilde{\boldsymbol{\eta}}) + \nabla^2 \hat{\ell}(\tilde{\boldsymbol{\eta}})(\hat{\boldsymbol{\eta}} - \tilde{\boldsymbol{\eta}}) = 0 + \nabla^2 \hat{\ell}(\tilde{\boldsymbol{\eta}})(\hat{\boldsymbol{\eta}} - \tilde{\boldsymbol{\eta}}),$$

which results in

$$\sqrt{m}(\tilde{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}}) \approx -[\nabla^2 \hat{\ell}(\tilde{\boldsymbol{\eta}})]^{-1} [\sqrt{m} \nabla \hat{\ell}(\hat{\boldsymbol{\eta}})]. \quad (2.13)$$

Conditional on the data, the true MLE is fixed and so Expression (2.13) measures the MCMC-based variability inherent in estimating $\hat{\boldsymbol{\eta}}$ by $\tilde{\boldsymbol{\eta}}$. From expression (2.13), we obtain

$$\text{Var}(\tilde{\boldsymbol{\eta}} \mid \text{data}) \approx \frac{1}{m} [\nabla^2 \hat{\ell}(\tilde{\boldsymbol{\eta}})]^{-1} \text{Var}[\sqrt{m} \nabla \hat{\ell}(\hat{\boldsymbol{\eta}})] [\nabla^2 \hat{\ell}(\tilde{\boldsymbol{\eta}})]^{-1} \approx \frac{1}{m} [\hat{\mathbf{I}}(\tilde{\boldsymbol{\eta}})]^{-1} \tilde{V} [\hat{\mathbf{I}}(\tilde{\boldsymbol{\eta}})]^{-1} \quad (2.14)$$

where

$$\tilde{V} = \frac{1}{m^2} \left[\sum_{i=1}^m \exp \left\{ (\boldsymbol{\eta}_0 - \tilde{\boldsymbol{\eta}})^\top \mathbf{g}(\mathbf{Y}_i) \right\} \right]^2$$

is an estimate of the variance of $\sqrt{m} \nabla \hat{\ell}(\hat{\boldsymbol{\eta}})$, assuming that the $\mathbf{Y}_i$ are drawn sufficiently far apart in the Markov chain to be considered roughly independent. See Geyer (1994) and Hunter and Handcock (2006) for additional details on the $\tilde{V}$ estimator.

2.6 Application to a Gilbert–Erdős–Rényi model

We illustrate these ideas on the Gilbert–Erdős–Rényi example in Section 2.3. This network has 272 edges, with MLE $\hat{\eta} = \log(272/508) = -0.625$. Starting from $\eta_0 = 1.099$ as depicted in Figure 2.1(b), corresponding to $b_0 = .75$, we sample 100 random $\text{binomial}(780, b_0)$ variables, each of which can be viewed as the number of edges in a randomly generated Gilbert–Erdős–Rényi network with $\eta_0 = 1.099$. The mean of the sample is $\bar{\xi}_0 = 584.46$, while the minimum and maximum are 556 and 616. As in Figure 2.1(c), the naive approximation (2.6) will not have a maximizer; this is due to the fact

that $g(\mathbf{y}^{\text{obs}}) = 272$ is not in the interval $(556, 616)$.

With $\hat{\xi}_{\infty} = 272$ and $\bar{\xi}_0 = 584.46$, our step-halving γ -selection scheme applied to the approximation (2.6) first tests the midpoint 428.23, followed by 506.35, 545.40, and finally 564.93 before finding a point contained in $(556, 616)$. We thus take $\hat{\xi}_1 = 564.93$ as the new “observed” value of the network statistic and find the maximum of the corresponding (naive) log-likelihood ratio

$$\ell(\eta) - \ell(\eta_0) \approx (\eta - 1.099) \times 564.93 - \log \left[\frac{1}{100} \sum_{i=1}^{100} \exp \{ (\eta - 1.099) \times g(\mathbf{Y}_i) \} \right].$$

This maximum occurs at $\eta = 0.909$, which becomes the value of η_1 , and we iterate.

After 10 such iterations, we finally achieve $\hat{\xi}_{11} = 272$, which means that the random sample using $\eta_{10} = -0.571$ includes values both larger and smaller than 272. We then take one final, much larger sample of size $m = 10^5$ and, based on this sample, we find a final approximate MLE of $\hat{\eta} = -0.6248$ (compared to the true MLE $\hat{\eta} = -0.625$). In this example, this process took ten steps to find an η_t value that produced a sample containing $\hat{\xi}_{\infty}$. The first and last iterations of this process are depicted in Figure 2.3.

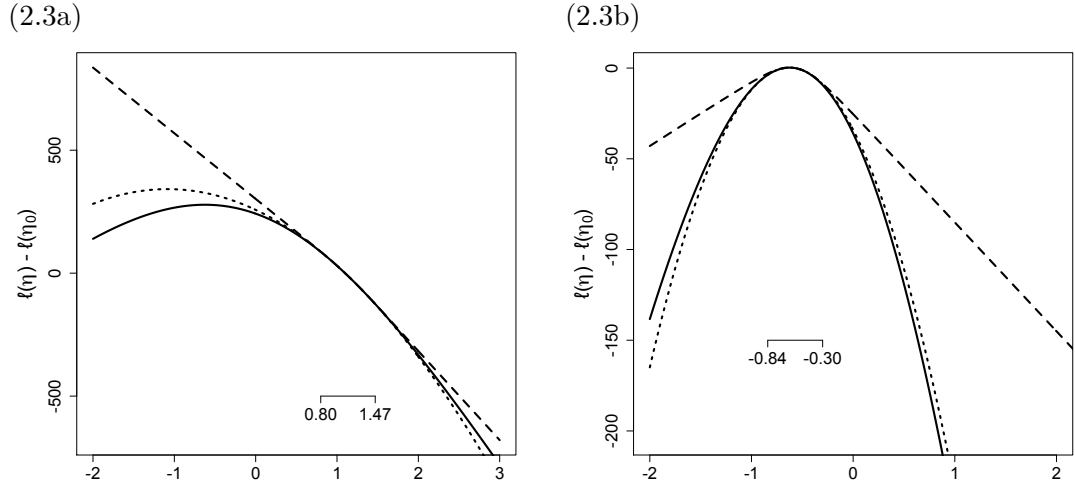


Figure 2.3. The solid line is the true log-likelihood ratio function, the dotted line is Approximation (2.6), and the dashed line is Approximation (2.7). The values of η_0 are 1.099 in (a) and -0.571 in (b). Since the closed-form map from mean-value parameters ξ to canonical parameters η is known, the range of η values corresponding to the range of sampled statistics is shown as a short segment with its endpoints marked in each plot.

Figure 2.3 suggests that the naive approximation used in our illustration above might

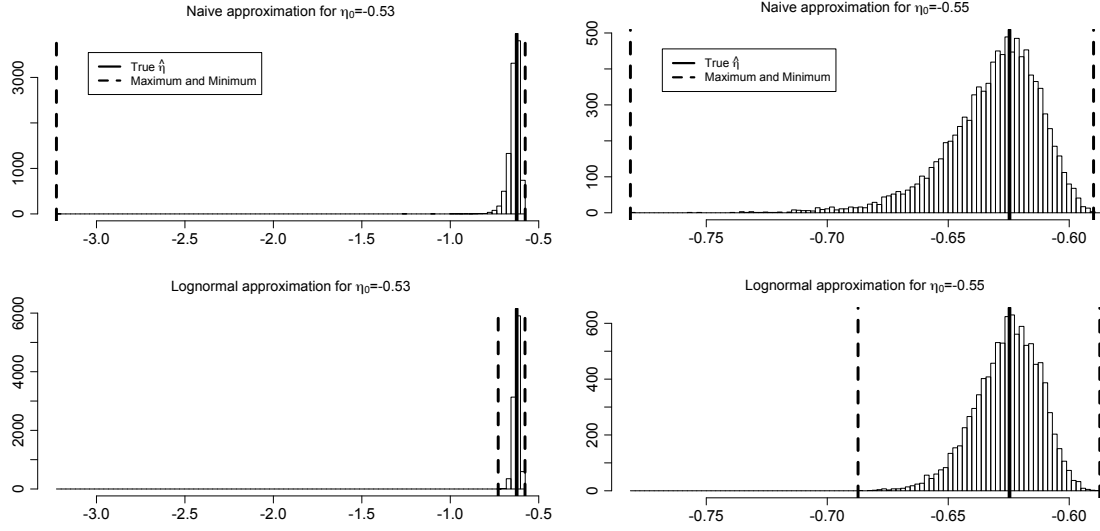


Figure 2.4. Distribution of the approximate MLE for the final step of the example, for the naive approximation (top) and the lognormal approximation (bottom), for $\eta_0 = -0.53$ (on left) and $\eta_0 = -0.55$ (on right). The thick dashed lines indicate the minimum and maximum η values found by each approximation, and the thick solid line marks the true MLE.

not perform as well as the lognormal approximation (2.7). In fact, Equation (2.7) in this example amounts to the normal approximation to the binomial. To compare the performance of the two approximations, we focus on the final step of our Gilbert–Erdős–Rényi example. Figure 2.4 shows histograms of 10,000 values of $\hat{\eta}$ obtained from a hypothetical “final” step of the example, taking $\eta_0 = -0.53$ in the left column and $\eta_0 = -0.55$ in the right column.

The true MLE corresponding to $g(y^{obs}) = \hat{\xi} = 272$ is $\hat{\eta} = -0.625$. The approximations for several nearby values for η_0 are shown in Table 2.1. When $\eta_0 = -0.625$, the MLE, both estimates are quite similar, as are their variances. When η_0 moves away from the MLE, however, the estimate from the naive approximation deteriorates, and the standard deviation of the naive approximation begins to explode in comparison to the standard deviation of the lognormal approximation.

Table 2.1. Compare means to the true MLE, -0.6246769 . In some of the 10,000 samples drawn for each η_0 , $\hat{\xi} = g(y^{obs})$ is not in the convex hull of the sampled statistics, so, when necessary, we show the results with these cases removed for both approximations.

η_0	Sample size*	Mean $\tilde{\eta}_{naive}$	Mean $\tilde{\eta}_{lognormal}$	StDev $\tilde{\eta}_{naive}$	StDev $\tilde{\eta}_{lognormal}$
-0.625	10000	-0.625	-0.625	0.0078	0.0077
-0.57	10000	-0.628	-0.625	0.0134	0.0110
-0.56	10000	-0.629	-0.626	0.0162	0.0122
-0.55	10000	-0.632	-0.626	0.0199	0.0134
-0.54	10000	-0.635	-0.626	0.0267	0.0146
-0.53	10000	-0.639	-0.626	0.0422	0.0158
-0.52	9999	-0.647	-0.626	0.0728	0.0171
-0.51	9985	-0.660	-0.625	0.1158	0.0182
-0.5	9948	-0.683	-0.625	0.2069	0.0196
-0.45	6615	-0.979	-0.617	0.7830	0.0231
-0.4	1360	-1.106	-0.601	0.9614	0.0264

2.7 Application to network datasets

2.7.1 A biological network

Here, we consider the E. coli transcriptional regulation network of Shen-Orr et al. (2002), based on the RegulonDB data of Salgado et al. (2001) and depicted in Figure 2.5. This network is analyzed using ERGMs, along with other biological networks, by Saul and Filkov (2007). In this network, the nodes represent operons and the edges indicate a regulating relationship between two nodes. Specifically, a directed edge from A to B means that A encodes a transcription factor that regulates B. Self-regulation is possible and occurs for 59 nodes. We call these “self-edges” or simply “loops.”

Although the network is directed and contains self-edges, here we will mimic the analysis of Saul and Filkov (2007) and treat it as undirected with 519 edges and 418 nodes, ignoring the self-loop information (five of the original 423 nodes were deleted because they had *only* self-edges and no other edges). For comparison, we then include an analysis that includes the self-loop information as a nodal covariate.

Saul and Filkov (2007) fit an ERGM in which the statistics $\mathbf{g}(\mathbf{y})$ are those listed in Table 2.2, though they used MPLE to produce parameter estimates because approximate

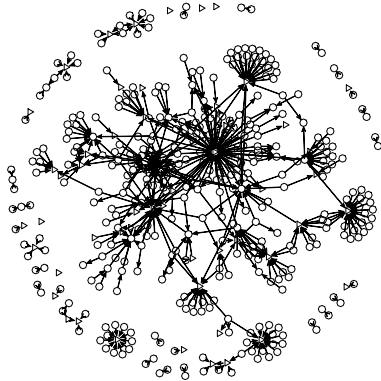


Figure 2.5. The *E. coli* transcriptional regulation network of Shen-Orr et al. (2002), with self-regulating nodes depicted as triangles.

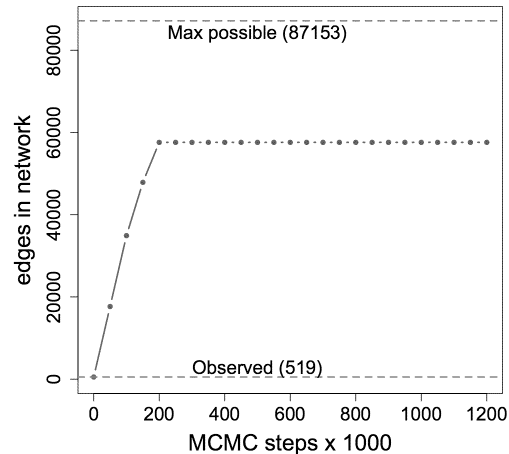


Figure 2.6. The model including a 2-star term and fit using MPLE produces unrepresentative random networks.

maximum likelihood estimates proved unobtainable. We nearly duplicate their MPL estimates here, obtaining the results shown in column 3 of Table 2.2. It was not possible to duplicate the results exactly because Saul and Filkov (2007) do not report the value of the θ_s parameter they used for the GWDeg statistic, though we came close using $\theta_s = 3.75$.

When we run a Markov chain to simulate from the first model of Table 2.2, we encounter problems: Figure 2.6 is a time-series plot of the edge-count of 25 networks sampled at intervals of 50,000 iterations from the Markov chain: The chain yields only networks that are very dense, nothing like the observed network. The problem here is the presence of the 2-star term in the model: For a high-degree node, adding one edge can increase the 2-star count by a lot. Thus, when the 2-star coefficient is positive as in this case, the Markov chain tends to favor networks with more and more edges. This problem, which afflicts models that contain certain types of terms such as the 2-star, is called model degeneracy by Handcock (2003). A full treatment of the topic of degeneracy is beyond the scope of this work, but Schweinberger (2010) describes in detail how certain terms, when included among the $\mathbf{g}(\mathbf{y})$ statistics, have the effect of essentially reducing the dimension of the space of parameter values for which this degeneracy behavior does not occur. As a result, models containing these problematic terms essentially render

Table 2.2. Model terms and MPLEs for the E. coli dataset of Shen-Orr et al. (2002). The first model is the model specified and fit in Saul and Filkov (2007). The second model is our adaptation to avoid issues of degeneracy.

Term(s)	Description:	Model 1	Model 2
Edges	# of Edges	2.82	-5.35
2-Deg	# of nodes with degree 2, $D_2(\mathbf{y})$	-1.20	-2.58
3-Deg	# of nodes with degree 3, $D_3(\mathbf{y})$	-1.75	-3.06
4-Deg	# of nodes with degree 4, $D_4(\mathbf{y})$	-1.45	-2.39
5-Deg	# of nodes with degree 5, $D_5(\mathbf{y})$	-1.34	-1.85
2-Star	# of pairs of edges with one end in common, $S_2(\mathbf{y})$	0.01	
GWDeg	Weighted sum of 1-Deg, 2-Deg, $\dots$, $(n-1)$ -Deg, with weights tending to 1 at a geometric rate: $u(\mathbf{y}; \theta_s) = e^{\theta_s} \sum_{i=1}^{n-1} \left\{ 1 - (1 - e^{\theta_s})^i \right\} D_i(\mathbf{y})$, with $\theta_s = 3.75$; with $\theta_s = 0.25$	-3.89	8.13

simulation-based estimation methods useless. On the other hand, terms such as GWDeg (and GWESP and GWDSP as described in Section 2.7.2), when used with positive θ_s parameters as we do here, allow modelers to circumvent these vexing model degeneracy problems. Further discussion of the GWDeg, GWESP, and GWDSP terms is provided by Hunter (2007) and Robins et al. (2007b). Since Model 1 cannot be fit using simulation-based methods, we drop the 2-star term and fit a new model, using a different value of $\theta_s = 0.25$ for the GWDeg term.

Taking the MPLE as the initial value $\boldsymbol{\eta}_0$ in fitting Model 2, the stepping procedure using the lognormal approximation (2.7) finds an approximate MLE in seven stepping iterations, plus a final iteration using the true observed statistics (see Figure 2.7). The naive approximation (2.6) gives parameter estimates very similar to those of the lognormal approximation, so we omit them from Table 2.3, which also gives standard error estimates. There is a large discrepancy between the MPLE and the approximate MLEs. In addition, it is apparent in Figure 2.7a that the sample generated from the MPLE does not correspond very closely to the observed statistics. In contrast, Figure 2.7d shows that the mean of the sample generated from the approximate MLE is quite close to the

observed statistics.

Table 2.3. Approximate MLEs (with standard errors and MCMC standard errors in parentheses) obtained using the stepping algorithm and the lognormal approximation (2.7). Model 2 is based on the analysis of Saul and Filkov (2007) but omits the problematic 2-star term.

Parameter	Model 2	Model 2 plus self-edges
Edges	−5.07 (0.042, 0.003)	−5.85 (0.069, 0.007)
2-Deg	−1.46 (0.141, 0.004)	−1.36 (0.147, 0.014)
3-Deg	−2.35 (0.193, 0.005)	−2.02 (0.203, 0.013)
4-Deg	−2.30 (0.228, 0.006)	−1.76 (0.233, 0.008)
5-Deg	−2.93 (0.413, 0.009)	−2.30 (0.424, 0.024)
GWDeg(.25)	1.81 (0.283, 0.005)	2.34 (0.341, 0.021)
noself↔self	—	1.57 (0.033, 0.0002)
self↔self	—	1.24 (0.121, 0.001)

Although Saul and Filkov (2007) did not use the self-regulation information in the *E. coli* dataset, we may consider it in a simplistic manner by adding two additional terms to Model 2. The noself↔self and self↔self coefficients are estimates of the additional log-odds of an edge corresponding to node pairs in which one is a self-regulator and the other is not, in the first case, and in which both are self-regulators, in the second. Here, the case in which neither node is a self-regulator serves as a “baseline” to which the other cases are compared. This analysis could be improved if the MCMC algorithm were expanded to allow for self-edges, since currently the sample space $\mathcal{Y}$ of possible networks disallows these loops. Since the `ergm` package does not currently allow for this type of simulation, we do not present such an analysis here.

2.7.2 A Sociological Dataset

We analyze a well-known social network dataset involving “sociational” (friendship and socioemotional) interactions among a group of 39 workers in a tailor shop in Zambia. The data were collected from June 1965 to August 1965, the first of two time periods during which Kapferer (1972) collected data on these individuals. In these networks, an edge between two individuals indicates an interactional relationship, which in most cases is a reciprocal relationship. Originally, Kapferer also recorded some asymmetrical interactional relationships in which one individual received more than he gave; however,

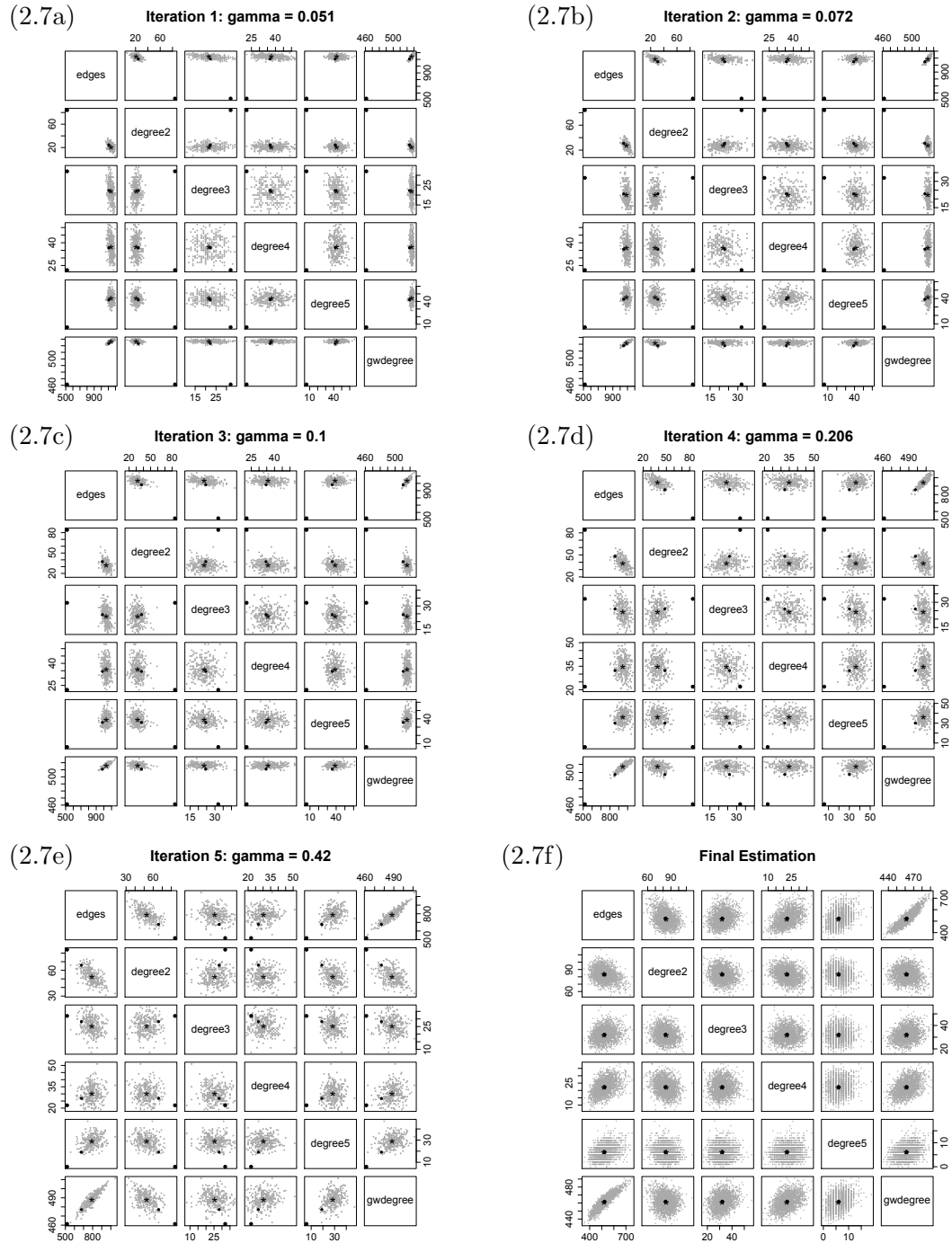


Figure 2.7. Plots of the sampled statistics from selected iterations, showing that ξ (the small black point) is moving from the mean of the sample (the star) towards the observed statistics (the large black point) without leaving the convex hull of the sample. In the final plot, ξ is the same as the observed statistics.

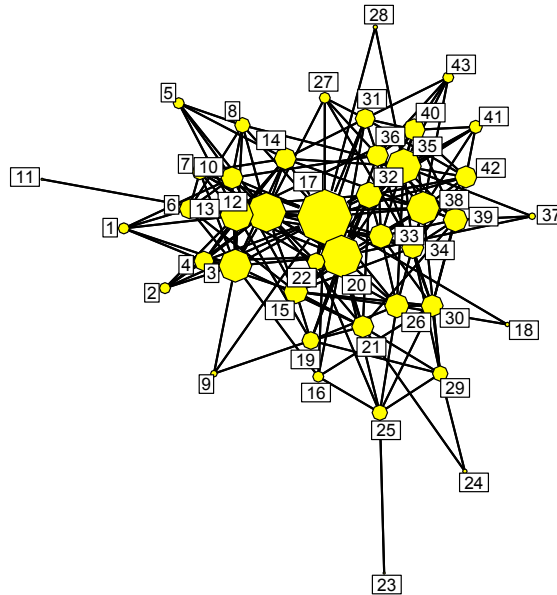


Figure 2.8. Sociational network of Kapferer (1972). An edge between nodes i and j indicates that Kapferer observed person i and person j interacting in their work environment on more than one occasion during time period 1. The vertices have size proportional to their sociality (number of workers with whom they interacted during time 1).

here we simply consider such asymmetric relationships as symmetric, which is common for this particular dataset. We do not analyze the dataset for the second period (take in early 1966) here, and, although there are actually 43 individuals in the network at the first time period (shown in Figure 2.8), we only consider the 39 who were present for both time periods. This smaller 1965 dataset (missing the workers named Lenard, Peter, Lazarus, and Laurent) is much better-known than the full network.

This network has of 158 edges, corresponding to density $158/\binom{39}{2} = 0.213$ and average degree $158/39 = 4.05$. Typically, increasing the number of nodes of a social network does not dramatically increase the average degree, which means that larger networks tend to have smaller densities. Yet they also tend to be harder to fit using MCMC MLE techniques due to the fact that the Markov chain cannot more easily visit a larger proportion of the sample space. On the other hand, for degenerate models, even small networks can cripple MCMC MLE algorithms. In addition to the 2-star term described in Section 2.7, terms such as the number of k -stars (for general $k > 2$) and the number of triangles can also result in degeneracy, as explained in Schweinberger (2010). Thus, ERG

models containing these terms are not viable despite their theoretical appeal (Frank and Strauss, 1986).

Alternatives to these terms are introduced by Snijders et al. (2006) and expressed by Hunter (2007) as the “geometrically weighted” terms `GWDegree`, `GWESP`, and `GWDSP`. Roughly speaking, these terms capture the spread of the degree distribution, the tendency to transitive closure, and the preconditions for transitivity, respectively. More detailed explanations of these terms are given by Snijders et al. (2006), where they are discussed in their alternative forms: alternating k -stars, alternating k -triangles, and alternating k -twopaths, respectively. For now, we mention only that these geometrically weighted statistics are shown by Schweinberger (2010) to lead to models free from degeneracy as long as the special parameter associated with each term is greater than $-\ln 2$. In the models that follow, we set these parameters at 0.25 for each of the three statistics.

We illustrate the use of the `GWDegree`, `GWESP`, and `GWDSP` terms in a model by providing R commands and output that can be used to fit a model with these four terms, along with the edges terms. These commands are intended to be used in conjunction with the `ergm` package version 2.2-6:

```
> load ("kapferer.soc1.RData")
> kformula <- kapferer.soc1 ~ edges + gwdegree(0.25, fixed=TRUE) +
                                gwesp(0.25, fixed=TRUE) +
                                gwdsp(0.25, fixed=TRUE)
> km <- ergm(kformula, maxit=1, seed=123,
             control=control.ergm(stepMCMCsize=1000,
                                   style="Stepping", metric="lognormal"))
> summary(km)
...
Monte Carlo MLE Results:
```

	Estimate	Std. Error	MCMC s.e.	p-value
edges	-3.085361	0.059624	0.065	<1e-04
gwdegree	0.291581	1.082166	1.992	0.788
gwesp.fixed.0.25	1.483683	0.018605	0.008	<1e-04
gwdsp.fixed.0.25	-0.122583	0.006203	2.07e-05	<1e-04

The output reveals that the `GWDegree` effect does not appear to be statistically significant; thus, we may refit the model without this term:

```
> kformula2 <- kapferer.soc1 ~ edges + gwesp(0.25, fixed=TRUE) +
```

```

                                gwdsp(0.25, fixed=TRUE)
> km2 <- ergm(kformula2, maxit=1, seed=123,
              control=control.ergm(stepMCMCsize=1000,
                                   style="Stepping", metric="lognormal"))
> summary(km2)
...
Monte Carlo MLE Results:

```

	Estimate	Std. Error	MCMC s.e.	p-value
edges	-3.067248	0.467562	0.422	<1e-04
gwesp.fixed.0.25	1.458001	0.331176	0.299	<1e-04
gwdsp.fixed.0.25	-0.116672	0.006145	0.000439	<1e-04

Roughly speaking, these parameter values reveal a propensity to completion of triads in this network but no overall tendency to form two-paths except those that lead to completed triads. These models are each fit using the stepping algorithm together with the lognormal approximation to the log-likelihood; each model requires only four stepping iterations when used with a stepping sample size of `stepMCMCsize=1000`.

2.8 Discussion

The procedure in this chapter addresses the serious problem encountered when implementing approximate maximum likelihood techniques to find estimates for model parameters in Exponential-family Random Graph models (assuming that the Y_{ij} variables are not independent): Approximations to the log-likelihood ratio $\ell(\boldsymbol{\eta}) - \ell(\boldsymbol{\eta}_0)$ tend to be poor when $\boldsymbol{\eta}$ is far from $\boldsymbol{\eta}_0$. Our algorithm moves $\boldsymbol{\eta}_0$, step by step, closer to $\hat{\boldsymbol{\eta}}$ until the MCMC sample generated from $\boldsymbol{\eta}_0$ does in fact cover $\mathbf{g}(\mathbf{y}^{\text{obs}})$. By combining this stepping procedure with the lognormal approximation described in Section 2.2, we are able to estimate approximate MLEs in many scenarios in which this was not previously possible. The techniques described here are implemented in the `ergm` package of Handcock et al. (2010).

One benefit of our method is that it could remove the current necessity that MCMC MLE algorithms choose an initial value $\boldsymbol{\eta}_0$ that is already close to the true MLE. Currently there are no better options in the literature than starting at the MPLE, which is problematic for at least two reasons. First, multiple authors have shown that the MPLE may have poor statistical properties. Second, sometimes we may wish to fit an ERGM

based solely on the vector $\mathbf{g}(\mathbf{y}^{\text{obs}})$ of network statistics; after all, the ERGM of Equation (2.1) depends on the network only through $\mathbf{g}(\mathbf{y})$. However, finding an MPLE requires an *actual network* that actually possesses the given network statistics. Such a network might not exist if $\mathbf{g}(\mathbf{y}^{\text{obs}})$ arises as some vector of theoretical interest rather than from a network dataset—for instance, sometimes $\mathbf{g}(\mathbf{y})$ must consist of integers (as it does for all but the GWDeg statistic in our Section 2.7 example) but we might want to estimate the model parameters using a non-integer-valued vector of “observed” statistics. Even when possible, finding such a network introduces a potentially unnecessary additional computational step. Furthermore, the vector $\mathbf{g}(\mathbf{y})$ does not uniquely determine a network, which means that different possible “observed” networks could produce $\mathbf{g}(\mathbf{y}^{\text{obs}})$ and there is no guarantee that all of these will yield the same MPLEs. For all of these reasons, approximate methods for finding the true maximum likelihood estimator are preferable to MPLE methods; and these approximate methods continue to improve with algorithmic refinements such as those we present here.

Several extensions and improvements to the technique presented here are possible. For instance, instead of choosing γ in Section 2.4 so that the new “observed” statistics are just barely inside the convex hull of sampled statistic vectors, we might make the choice based on some notion of data depth within these statistics (Liu et al., 2006). Or we could start the algorithm by fixing some of the parameters at zero, bringing others on target one at a time or in groups using the steplength routine. Finally, it is possible that our algorithm, or indeed any simulation-based algorithm, would work better if the underlying MCMC sampler were improved. Although improving the MCMC algorithm is not the focus of our current work, it is certainly an interesting topic of research unto itself. We do not claim that the current algorithm is a panacea, capable of moving straight to an approximate MLE for any model, from any starting parameter value; however, it represents one more tool to employ in the tricky task of estimating these models, and our experience is that it has succeeded in certain problems that were formerly intractable.

Bias-adjusted maximum likelihood estimation for exponential-family random graph models

3.1 Introduction

A statistical model for observed network data $\mathbf{y}^{obs}$ allows us to both summarize quantitatively the effects that may have given rise to the network and to simulate new networks from a probability distribution that resembles the generating distribution of the original network. The work in this chapter advances the current practice of parameter estimation so that we can improve accuracy in prediction and inference for network models.

Statistical models for networks can be written as exponential-family models, and this ERGM (exponential-family random graph model) class for simple, loopless, unweighted networks is of the form

$$P_{\boldsymbol{\theta}}(\mathbf{Y} = \mathbf{y}) = \frac{\exp\{\boldsymbol{\theta}^t \mathbf{g}(\mathbf{y})\}}{\kappa(\boldsymbol{\theta})}, \text{ where } \kappa(\boldsymbol{\theta}) = \sum_{\substack{\text{all possible} \\ \text{graphs } \mathbf{z}}} \exp\{\boldsymbol{\theta}^t \mathbf{g}(\mathbf{z})\} \quad (3.1)$$

where $\boldsymbol{\theta}$ is a parameter vector to be estimated and $\mathbf{g}(\mathbf{y})$ is a user-defined vector of graph statistics. The log-likelihood is $\ell(\boldsymbol{\theta}) = \boldsymbol{\theta}^t \mathbf{g}(\mathbf{y}^{obs}) - \log \kappa(\boldsymbol{\theta})$.

Straightforward maximization of the resulting loglikelihood is unrealistic, as discussed in Chapters 1 and 2, even for small networks, because the likelihood involves a sum that includes a number of terms which increases geometrically with the number of nodes in the network. Two well-established alternatives to straightforward maximum likelihood estimation are maximum pseudolikelihood estimation (MPLE, as in Strauss and Ikeda, 1990) and approximate maximum likelihood estimation as obtained through Markov chain Monte Carlo methods (MCMC-MLE, as in Geyer and Thompson, 1992).

MPLE simply ignores the possible dependence among dyads and maximizes the likelihood equation that results. By treating the Y_{ij} as independent, the complicated normalizing constant simplifies, and the MPLE can be found as the MLE for a logistic regression model with each Y_{ij} as an observation:

$$\text{logit}(P(Y_{ij} = 1)) = \boldsymbol{\theta}^t \boldsymbol{\delta}_g(\mathbf{y})_{ij}, \quad (3.2)$$

where $\boldsymbol{\delta}_g(\mathbf{y})_{ij}$ is the change in $\mathbf{g}(\mathbf{y})$ when y_{ij} is changed from 0 to 1. The vector $\boldsymbol{\delta}_g(\mathbf{y})_{ij}$ is called the *vector of change statistics* (Wasserman and Pattison, 1996) and is the conditional log-odds that $Y_{ij} = 1$ given the rest of the network. For an undirected graph, since there are n nodes and $\binom{n}{2}$ dyads, there will be $\binom{n}{2}$ such equations (3.2).

MPLE has strong historical usage, largely because until rather recently approximate MLE was not usually possible. It is also convenient that, when independence of the arcs is an appropriate assumption, the MPLE is exactly the MLE. However, unlike the properties of the MLE, the properties of the MPLE are not well-understood, and a growing body of literature is beginning to examine the shortcomings of the MPLE.

MCMC-MLE is discussed in greater length in Chapters 1, 2, and 4 of this thesis, and in Geyer and Thompson (1992), Corander et al. (1998), Corander et al. (2002), Snijders (2002), and Hunter and Handcock (2006). Some of the reasons that approximate MLE has frequently failed in the past are poor model choice (frequently resulting in degeneracy of the probability distribution for random graphs) and computing problems due to computationally expensive algorithms or yet-unsolved computational problems. Recent developments in model specification (Robins et al., 2007b; Snijders et al., 2006)

and algorithms (see Chapter 2) are enabling approximate MLEs to be found in many cases where this was not previously possible. In light of these new developments in approximating the MLE, further comparisons between MPLE and MLE can be made.

Wasserman and Robins (2005) and Lubbers and Snijders (2007) discuss problems with the pseudolikelihood standard errors, and Corander et al. (1998), Wasserman and Robins (2005), Robins et al. (2007b), and Lubbers and Snijders (2007) reveal troubling divergence between the MPLE and the approximate MLE in a variety of cases. Specifically, these papers examine the behavior of the MPLE and the approximate MLE in Markov models, models with 2-stars and triangles, and models with alternating k-stars and alternating k-triangles, concluding that the divergence between the MPLE and MLE is highly dependent on the data, with the MPLE and MLE more similar for low-dependence structures and less similar for higher-dependence structures. Corander et al. (1998) suggests that the divergence between the MPLE and the MLE grows smaller as the network size increases, but this conclusion is not well-substantiated in general.

Each of these papers compared the MPLE to the MLE, noting a “bias” in the MPLE as compared to the MLE, but none compared the estimates to a true underlying parameter. For this reason, none of these authors could reflect on the possibility of bias in the MLE. This, of course, is difficult to do in general, as we are unlikely to know the true underlying generative process, much less the corresponding model parameters, for an observed network; however, we can estimate a true bias by constructing a simulation study where we fix the generative parameter and compare the estimates to this “truth.” This was done in van Duijn et al. (2009), which fixed the true parameter to be the MLE obtained on the original data and then compared the MPLE, a bias-corrected version of the MPLE, and the MLE in a number of simulations generated from this fixed parameter. Improvements were seen for this bias-corrected MPLE (called the *MBLE*) over the usual MPLE in both the canonical and the mean value parameterizations.

3.2 Bias-reduction theory

The evidence presented in Corander et al. (1998), Wasserman and Robins (2005), Robins et al. (2007b), Lubbers and Snijders (2007), and van Duijn et al. (2009) suggests that, in many circumstances, there is a substantial bias in the MPLE, and possibly in the MLE. For this reason we explore further the conditions under which the bias seems to occur, and, for the first time, we apply a bias-correction to the MLE and compare its performance to the bias-adjusted MPLE, the usual MPLE, and the usual approximate MLE.

Note that bias-reduction may not reduce overall mean squared error. Bias correction can result in larger standard errors than for the MLE (Cox and Hinkley, 1974; Efron and Tibshirani, 1993). We will also look at these standard errors to compare the confidence intervals of the original parameters to the confidence intervals of their bias-corrected versions.

3.2.1 Asymptotic bias

One way to reduce the bias of the maximum likelihood estimate is to first calculate the MLE and then subtract its first-order asymptotic bias (Cox and Hinkley, 1974; Firth, 1993). The bias for a univariate MLE, $\hat{\theta}$, can be found by a Taylor series expansion of the score function, $U.(\theta) = \sum_{i=1}^m U_i(\theta)$ for $\mathbf{Y}_1, \dots, \mathbf{Y}_m$ at the MLE:

$$0 = U.(\hat{\theta}) = U.(\theta) + (\hat{\theta} - \theta) \left[\frac{d}{d\theta_j} U.(\theta) \right] + \frac{1}{2} (\hat{\theta} - \theta)^2 \left[\frac{d^2}{d\theta_j^2} U.(\theta) \right] + O_p(m^{-1/2}), \quad (3.3)$$

and hence, as in Cox and Hinkley (1974),

$$\begin{aligned} E(\hat{\theta} - \theta) E \left[\frac{d}{d\theta} U.(\theta) \right] + \text{cov} \left[\hat{\theta}, \frac{d}{d\theta} U.(\theta) \right] + \frac{1}{2} E \left((\hat{\theta} - \theta)^2 \right) E \left[\frac{d^2}{d\theta^2} U.(\theta) \right] \\ + \frac{1}{2} \text{cov} \left[(\hat{\theta} - \theta)^2, \frac{d^2}{d\theta^2} U.(\theta) \right] = O(m^{-1/2}). \end{aligned} \quad (3.4)$$

Here, the second term, $\text{cov} \left[\hat{\theta}, (d/d\theta)U.(\theta) \right]$ can be written as

$$\text{cov} \left[\hat{\theta}, \frac{d}{d\theta}U.(\theta) \right] = (\mathbf{i}(\theta))^{-1} E \left[U_1(\theta) \frac{d}{d\theta}U_1(\theta), \right].$$

The first-order asymptotic covariance $E \left[m(\hat{\theta} - \theta)^2; \theta \right]$ is the inverse of the Fisher information of θ , $\mathbf{i}^{-1}(\theta)$, with equality between the expected information, $\mathbf{i}(\theta)$, and the observed information, $\mathbf{I}(\theta)$, in any exponential family (Cox and Hinkley, 1974). By considering the derivatives of $U.(\theta)$ in terms of the likelihood function, $L(\theta)$, (so that, for example, $(d^2/d\theta^2)U.(\theta) = L''(\theta)/L(\theta) + L'(\theta)/L(\theta)$), the expected value of the second derivative of the score function can be found to be

$$E \left[\frac{d^2}{d\theta^2}U.(\theta) \right] = E \left[-3U.(\theta) \frac{d}{d\theta}U.(\theta) - U.^3(\theta) \right].$$

The fourth term, $\text{cov} \left[(\hat{\theta} - \theta)^2, (d^2/d\theta^2)U.(\theta) \right]$ is $o(m^{-1})$, and by definition of the score function, the multiplier attached to the bias, $E [(d/d\theta)U.(\theta)]$, is $-m\mathbf{i}(\theta)$.

Thus we have that the asymptotic first-order bias of the MLE in an exponential family is

$$E(\hat{\theta} - \theta) = -\frac{E\{U.(\theta) \frac{d}{d\theta}U.(\theta) + U.^3(\theta)\}}{2m\mathbf{i}^2(\theta)} + o(m^{-1}) = b_1(\theta) + o(m^{-1}). \quad (3.5)$$

By estimating $b_1(\theta)$ by $\hat{b}_1(\hat{\theta})$, we obtain $\tilde{\theta} \approx \hat{\theta} - \hat{b}_1(\hat{\theta})$ as a bias-corrected version of the MLE.

Likewise, the bias for multivariate $\hat{\boldsymbol{\theta}}$ can be found by a component-wise (in $\boldsymbol{\theta}$) Taylor series expansion of the score function, $U.(\boldsymbol{\theta}) = \sum_{i=1}^m U_i(\boldsymbol{\theta})$ for $\mathbf{Y}_1, \dots, \mathbf{Y}_m$ at the MLE.

3.2.2 Bootstrap and jackknife estimates of bias

Bootstrap and jackknife methods are another way to adjust for the bias in the MLE. The bootstrap, first proposed in general and first applied to bias-correction by Efron (Efron, 1979, 1990; Efron and Tibshirani, 1993), is an algorithm that takes many independent size n sub-samples from the original size n sample, with replacement, and calculates the value

of interest on these subsamples in order to develop an empirical sampling distribution from which to make inference. The simple bootstrap bias-corrected estimate is, for the value of interest, the MLE $\hat{\theta}$,

$$\hat{\theta}_{BC} = 2\hat{\theta} - \bar{\theta}^* \quad (3.6)$$

where $\bar{\theta}^*$ is the mean of the MLEs from the bootstrap samples.

A bootstrap bias-corrected estimator can also be constructed as

$$\hat{\theta}_{BC} = \hat{\theta} + \bar{\theta} - \bar{\theta}^*, \quad (3.7)$$

where $\bar{\theta}$ is the MLE calculated on *the average* of the resampled data. In other words, while $\bar{\theta}^*$ is the average of the MLEs calculated from each sample, this $\bar{\theta}$ is the MLE calculated on the “average sample.” This second version seems to estimate the bias much better than the simple estimate (see Efron and Tibshirani, 1993, sections 10.4 and 23.3).

The jackknife is a computationally simpler, and linear, approximation to the bootstrap, which actually predates the bootstrap in the literature (Quenouille, 1949; Tukey, 1958; Efron and Tibshirani, 1993). The jackknife samples can be called “leave one out” samples, in that the statistic of interest is calculated on n samples of size $n - 1$, each sample leaving out, systematically, one observation. The *delete-d jackknife* is an extension of the original jackknife such that the statistic is calculated on $\binom{n}{d}$ samples (or a random subset of these samples), where d is chosen such that $n = r * d$ for some integer r , and $\sqrt{n} < d < n$. This second condition is a rough guideline to ensure consistency in the jackknife estimate of the standard error, which is not met for the usual leave-one-out jackknife for non-smooth statistics.

The leave-one-out jackknife bias-corrected estimate is as follows (Efron and Tibshirani, 1993):

$$\tilde{\theta} = n\hat{\theta} - (n - 1)\bar{\theta}_s, \quad (3.8)$$

where $\bar{\theta}_s$ is the mean of the statistic of interest calculated for each subsample.

3.2.3 Inapplicability of these methods to ERGMs

Using the bootstrap or jackknife methods in ERGMs would require that, for each subsample, we either sample from the dyad pairs in the network (bootstrap) or leave out k dyads (jackknife). In this context, use of the bootstrap or jackknife for bias correction in this way would require deep theoretical understanding of how the estimates are related to the (changing) size of the network, a concept that is not at all intuitive or currently well-understood in the field in general, as well as completely new algorithms for computing the estimators. This inapplicability of the bootstrap or jackknife to ERGMs stems fundamentally from the changing concept of an *observational unit*. While in some ways we use the information from the dyads as if the dyads themselves were the observational units, the unit is actually the entire observed network itself, and we typically only have one of these.

In addition to the conceptual differences in how to apply the bootstrap or jackknife to ERGMs, all of the bias-reducing methods introduced thus far require calculation of the MLE before the correction can be applied. This restricts the applicability of these corrections to cases in which the MLE exists. In the ERGM setting, infinite MLEs can easily exist with positive probability (Geyer, 2009; Rinaldo et al., 2009), and hence any of these methods of bias-correction may be quite limited in this setting.

3.2.4 Firth penalty

A third method of bias-adjustment for the approximate MLE is introduced in Firth (1993). Notably, this bias-reduction is possible even for an infinite MLE. The Firth bias-reduction differs from several other bias-corrections in that, unlike, say, the jackknife and true bias correction, the Firth approach does not require that the uncorrected MLE is calculated first (or, again, that it is even finite). Instead, the Firth approach adjusts the score equation instead of directly adjusting the calculated estimator.

The idea is to modify the score function, $\mathbf{U}$, by a function of the local gradient, $\nabla \mathbf{U}$,

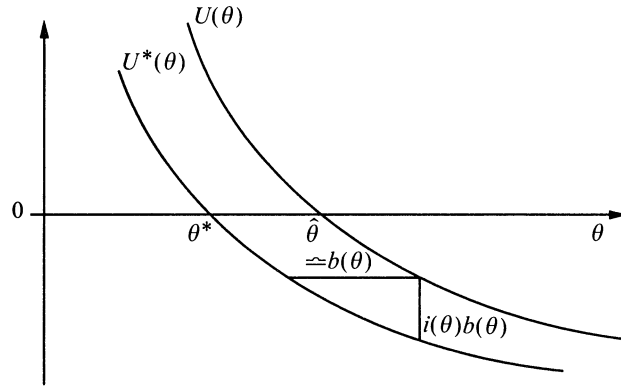


Figure 3.1. Modification of the unbiased score function.

and the bias of the MLE, $\mathbf{b}(\boldsymbol{\theta})$, like so:

$$\mathbf{U}^*(\boldsymbol{\theta}) = \mathbf{U}(\boldsymbol{\theta}) + \nabla \mathbf{U} \mathbf{b}(\boldsymbol{\theta}) = \mathbf{U}(\boldsymbol{\theta}) - \mathbf{i}(\boldsymbol{\theta}) \mathbf{b}(\boldsymbol{\theta}). \quad (3.9)$$

(The $\mathbf{i}(\boldsymbol{\theta})$ is the Fisher information matrix, as above.) By introducing a small bias in the score function before searching for a zero, some of the estimated bias in the MLE is corrected.

The intuition behind this modification for a one-parameter exponential family model is illustrated in Figure 3.1, taken from Firth (1993). Rather than taking $\tilde{\theta} = \hat{\theta} - b(\theta)$, (where $b(\theta)$ is the bias of the MLE) as in the usual asymptotic bias-correction and bootstrap and jackknife methods, we can find θ^* as a zero of a shifted version of the score function, $U^*(\theta)$. The point here is subtle but powerful: rather than finding the MLE and then subtracting off its bias, we can find a bias-corrected estimator *without requiring the existence of the original MLE*.

Since the gradient of the score function is $U'(\theta) = -i(\theta)$, shifting $U(\theta)$ by $b(\theta)$ on the θ -axis in the one-parameter case can be linearly approximated by shifting the function $U(\theta)$ by $-i(\theta)b(\theta)$; that is,

$$U^*(\theta) = U(\theta) - i(\theta)b(\theta) \approx U(\theta - b(\theta)).$$

More generally (for any dimension and for families other than canonical exponential families), Firth (1993) shows that a vector-valued shift $\mathbf{A}(\boldsymbol{\theta})$ to the (vector) score func-

tion will remove the first-order bias of the MLE, $\mathbf{b}_1(\boldsymbol{\theta})$, for any $\mathbf{A}$ satisfying $E(\mathbf{A}) = -\mathbf{i}(\boldsymbol{\theta})\mathbf{b}_1(\boldsymbol{\theta})/n + \mathbf{O}(n^{1/2})$. In fact, this shift will result in an additional bias-reduction up to the second-order for either choice (observed or expected) of the information matrix in $\mathbf{A}$ (Efron, 1975). This criterion for the vector $\mathbf{A}$ results from defining a modified score function for a general $\mathbf{A}$ with components $O_p(1)$

$$U_r^*(\boldsymbol{\theta}) = U_r(\boldsymbol{\theta}) - A_r,$$

where each component of the score function, $U_r(\boldsymbol{\theta})$, has been shifted by the components A_r . By expanding this modified score function about the true θ_r , the component-wise bias of the new estimator, $\boldsymbol{\theta}^*$, can be shown, as in section (3.2.1), to be

$$b(\theta_r^*) = \frac{1}{n}b_1(\theta_r) + \frac{1}{n}\kappa^{r,s}E(a_r) + O(n^{-3/2}),$$

where $\frac{1}{n}\mathbf{b}_1(\boldsymbol{\theta})$ is the vector-valued first-order bias of the MLE with components $b_1(\theta_r) = -\kappa^{r,s}\kappa^{t,v}(\kappa_{s,t,t} + \kappa_{s,tv})/2$. Here, $\kappa^{r,s}$ denotes the corresponding component of the inverse, $\boldsymbol{\kappa}^{-1}$, of the Fisher information matrix, $\boldsymbol{\kappa}$, with components $\kappa_{r,s} = \frac{1}{n}E(\mathbf{U}_r\mathbf{U}_s)$, and $\kappa_{s,t,t}$ and $\kappa_{s,tv}$ denote the joint cumulants $\frac{1}{n}E(U_sU_tU_t)$ and $\frac{1}{n}E(U_sd/d\theta_vU_{tv})$, respectively, with summation over subscripts other than r .

Hence, $\mathbf{A}$ will remove the first-order term of its bias if its components, a_r , have expectation $E(a_r) = -\kappa_{r,s}b_1(\theta_s) + O(n^{-1/2})$, which will be true for $\mathbf{A} = -\mathbf{i}(\boldsymbol{\theta})\mathbf{b}_1(\boldsymbol{\theta})/n$, using the expected information, and $\mathbf{A} = -\mathbf{I}(\boldsymbol{\theta})\mathbf{b}_1(\boldsymbol{\theta})/n$, using the observed information.

For an exponential family model, where, for all components, $\kappa_{r,st} = 0$ for all r , s , and t , $E(a_r) = -\kappa_{r,s}b_1(\theta_s)$ reduces to $E(a_r) = -n\kappa_{r,s}b_1(\theta_s) = -\kappa^{u,v}\kappa_{ruv}/2$, which can be written as $E(a_r) = 1/2\text{tr}(\mathbf{i}^{-1}(d\mathbf{i}/d\theta_r))$, and hence the penalty $\mathbf{A} = 1/2\log|\mathbf{i}(\boldsymbol{\theta})|$ satisfies the first-order bias reduction. The choice of using the expected information versus the observed information is a non-issue in the canonical parametrization of an exponential family because the observed information does not involve the data and hence, as mentioned above, the expected and observed information are the same.

It is shown in Firth (1993) that this logical adjustment to the score function is also the Jeffreys invariant prior for the canonical parametrization in an exponential

family, and Efron (1975), Amari (1985), and Lehman and Casella (1998) show that this bias-adjusted version of the MLE achieves second-order optimality among all bias-corrected, first-order efficient estimators. Li (1998) argues that minimization of the mean squared error without removing the first-order bias term is counterproductive since, “unless the bias is removed, the maximum likelihood estimate may be better in one parameterisation [sic] but worse in another” compared to any other first-order efficient estimator. However, by removing the first-order bias, the MSE is guaranteed to be smaller in all parameterizations (up to the second order).

3.2.5 Choice of parameterization

This optimality result ensures that the bias-adjusted MLE will improve estimation in either parameterization, and certainly estimation in both parameterizations can be considered simultaneously. However, there are different benefits and limitations to consider in each. The mean value parameterization (MVP) boasts a more intuitively “meaningful” connection to the data, in that the mean value parameters are on the same scale as the observed statistics. Assuming that the corresponding canonical parameter is known, the expected value and mean squared error of the mean value parameter can easily be estimated, allowing for direct comparison of how well the mean value parameter matches the observed statistics. This is often the major purpose of the network analysis: to find a model that can produce other networks with similar statistics.

Another seeming benefit of the mean value parameterization for usual exponential families is that the unadjusted maximum likelihood estimate for the mean value parameter is unbiased and therefore does not require an adjustment for the purpose of bias-reduction. This might seem to be an argument for doing estimation in the mean value parameterization; however, the distribution of the MVP-MLE may deviate substantially from the normal distribution, particularly via skewness, suggesting that the mean of this distribution may not be the best estimate of centrality for the MVP-MLE, and hence the unbiasedness may not be as valuable here as in the canonical parameterization, which tends to exhibit a closer approximation to the normal distribution for many families of models (Firth, 1993).

The non-normality of the mean value parameter is particularly relevant in binary logistic regression and rare-events modeling, where the log-odds parameters are much more likely to be approximately normally distributed than the number of events might be (Heinze and Schemper, 2002). In addition, this is a case where the log-odds may be the more useful scale, particularly for generalization to other groups of different sizes, where the counts would not retain their meaning. (One notable exception to this would be in insurance applications where the expected number of events and a confidence interval for this mean are the primary parameters of interest.) For the Erdos-Renyi-Gilbert example in ERGMs (Erdős and Rényi, 1959), which can be parameterized exactly as a binary logistic regression model and is one of the rare examples of an ERG model with a clear (and invertible) functional connection between the two parameterizations, namely that $\theta = \log(\mu/(1-\mu))$, van Duijn et al. (2009) shows that the rate of change of the canonical parameter, θ , as a function of μ , is unbounded as the probability goes toward 1 or 0, while the rate of change for the mean value parameter, μ , as a function of θ , is bounded between 0 and 1/4. Further discussion of the choice between parameters for log-linear models can be found in Agresti (2002).

Another consideration about the choice of parameterizations, features of the Firth adjustment, and the relationship between log-linear model and ERGMs is that, in models with a propensity for full- or quasi-separation in the data based on classification in one or in a combination of variables (rare-events, logistic regression with full separation, etc.), such situations will result in infinite maximum likelihood parameter estimates. Geyer (2009) and Rinaldo et al. (2009) discuss this phenomenon in the context of ERGMs in depth. Firth (1993) simplifies an example from Copas (1988) to illustrate a small logistic regression model with infinite MLEs for some possible sufficient statistics. In the example of Copas and Firth, the bias-adjusted (shrinkage) estimator can be found, even for the parameters with infinite MLEs. Firth also shows the adjustment necessary in general for estimation of the log-odds parameters in binary logistic regression. For the case of a single binomial observation (a one-proportion test), this adjustment results in

the bias-reduced empirical logit (Haldane, 1955; Anscombe, 1956; Cox and Snell, 1989),

$$p = \text{logit}\left(\frac{x + \frac{1}{2}}{n - x + \frac{1}{2}}\right).$$

Note that this type of behavior can happen even in perfectly well-behaved models such as the Erdős-Rényi-Gilbert, since it requires only that the observed graph statistics happen to be on the boundary of the sample space. This behavior is well understood (Rinaldo et al., 2009; Geyer, 2009) and does not depend on the choice of model, only on the location of the observed statistics in the sample space. Although Handcock (2003) refers to this behavior as degeneracy, it is qualitatively different from the more serious model-specific degeneracy exhibited when the fitted model places most of the probability mass on only a few possible graph configurations, in effect reducing the sample space to a very small, finite set. This type of model degeneracy, often occurring in Markov random graph models, for example, is a reflection of the inability of the proposed model to describe the true generative process (Schweinberger, 2010), whereas boundary-degeneracy can afflict any model.

A final point regarding parameterization is the fact that inference for the mean value parameter relies on knowledge of the corresponding canonical parameter. Since the MVP MLE, $\hat{\boldsymbol{\mu}}$, is simply the observed statistics, $\mathbf{g}(\mathbf{y}^{\text{obs}})$, standard errors can be based on $\text{cov}_{\boldsymbol{\theta}}\{\hat{\boldsymbol{\mu}}\} = \text{cov}_{\boldsymbol{\theta}}\{\mathbf{g}(\mathbf{Y})\}$, where $\text{cov}_{\boldsymbol{\theta}}\{\mathbf{g}(\mathbf{Y})\}$ is estimated as a sample mean from an MCMC sample of $\mathbf{g}(\mathbf{Y})$. Essentially, the difference between the parameterizations regarding estimation and measuring uncertainty is in the where and why of the need for MCMC approximations, but not in the computational complexity. The canonical parameter requires MCMC methods in order to approximate the MLE, but the variance of the canonical parameter, which can be expressed as $[-\nabla^2\ell(\boldsymbol{\theta})]^{-1} = \mathbf{i}^{-1}(\boldsymbol{\theta})$, can be estimated from this original MCMC sample. The mean value parameter MLE can be obtained immediately as $\mathbf{g}(\mathbf{y}^{\text{obs}})$, but its variance, which has no simple expression in general via the derivatives, must be obtained by taking an MCMC sample in order to find the canonical MLE, and then using that sample to obtain an estimate for $\text{cov}_{\boldsymbol{\theta}}\{\mathbf{g}(\mathbf{Y})\}$. Since inference in either parameterization requires knowledge of the estimate in the other

parameterization, and the MCMC sample used for inference in both parameterizations could be the same sample, it might be most sensible to consider both simultaneously.

3.3 Bias-adjusted MLE

Here we introduce the Firth correction to our approximate MLE, since van Duijn et al. (2009) suggests that the MLE has even more bias than the uncorrected MPLE in the natural parameterization. (By construction, the MLE has no bias in the mean value parameterization.) Following Firth (1993) directly, we can write this penalized true log-likelihood as

$$\ell_b(\boldsymbol{\theta}, \mathcal{Y}) = \ell(\boldsymbol{\theta}, \mathcal{Y}) + 1/2 \log |\mathbf{i}(\boldsymbol{\theta})| \quad (3.10)$$

where $\mathbf{i}(\boldsymbol{\theta})$ is the Fisher information matrix for the $\boldsymbol{\theta}$.

Since this penalized likelihood function is impossible to directly maximize, we employ Markov chain Monte Carlo methods to find an approximate MLE via the log-likelihood-ratio function $\ell(\boldsymbol{\theta}) - \ell(\boldsymbol{\theta}_0)$. As described in Chapter 2, we can use a naive sample-mean MCMC approximation for the unknown expectation in the log-likelihood-ratio

$$\ell(\boldsymbol{\theta}) - \ell(\boldsymbol{\theta}_0) = (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^t \mathbf{g}(\mathbf{y}^{\text{obs}}) - \log E_{\boldsymbol{\theta}} \exp \{ (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^t \mathbf{g}(\mathbf{Y}_i) \},$$

or we can use an approximation based on the assumption that the statistics are normally distributed. This latter option seems to work slightly better in practice (even when this assumption is not strongly met) and is the default approximation for maximum likelihood estimation in the **ergm** package in R (Handcock et al., 2010). This (unadjusted) approximation can be written:

$$\ell(\boldsymbol{\theta}) - \ell(\boldsymbol{\theta}_0) \approx (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^t \mathbf{g}_{\hat{c}}(\mathbf{y}^{\text{obs}}) + \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^t \hat{\boldsymbol{\Sigma}}_0 (\boldsymbol{\theta} - \boldsymbol{\theta}_0), \quad (3.11)$$

where

$$\mathbf{g}_{\hat{c}}(\mathbf{y}^{\text{obs}}) = \mathbf{g}(\mathbf{y}^{\text{obs}}) - \frac{1}{m} \sum \mathbf{g}(\mathbf{Y}_i), \text{ and } \hat{\boldsymbol{\Sigma}}_0 = \frac{1}{m-1} \sum \mathbf{g}(\mathbf{Y}_i) \mathbf{g}(\mathbf{Y}_i)^t.$$

The maximizer of the lognormal approximation is therefore

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_0 + \hat{\boldsymbol{\Sigma}}_0^{-1} \mathbf{g}_{\hat{c}}(\mathbf{y}^{\text{obs}}). \quad (3.12)$$

When we apply the Firth penalty to this lognormal approximate log-likelihood-ratio, we have

$$\ell_b(\boldsymbol{\theta}) - \ell_b(\boldsymbol{\theta}_0) \approx (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^t \mathbf{g}_{\hat{c}}(\mathbf{y}^{\text{obs}}) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^t \hat{\boldsymbol{\Sigma}}_0 (\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \frac{1}{2} \log \frac{|\mathbf{i}(\boldsymbol{\theta})|}{|\mathbf{i}(\boldsymbol{\theta}_0)|}, \quad (3.13)$$

which no longer has a closed-form maximizer; however, this maximizer can be found by numerically optimizing the approximate log-likelihood in 3.13.

The Fisher information $\mathbf{i}(\hat{\boldsymbol{\theta}})$ here is the inverse of the covariance of the simulated graph statistics generated in the MCMC sample, and we use the Fisher information matrix both for calculating the penalty and for estimating the curvature of the estimated log-likelihood-ratio, $\ell_b(\boldsymbol{\theta}) - \ell_b(\boldsymbol{\theta}_0)$. Firth (1993) suggests using the unadjusted Fisher information for the standard errors of the bias-adjusted estimator, as the first-order asymptotic covariance of both estimators will be the same.

The asymptotic standard errors for the mean value parameterization can be easily calculated because of the equality relationship between the mean value parameter and the original observed graph statistics: we simply use the covariance of the simulated graph statistics (which, as mentioned above, is also the inverse Fisher information matrix) to calculate the standard errors for the MLE. For the bias-adjusted estimate, we use the covariance of the simulated graph statistics under $\hat{\boldsymbol{\theta}}^*$, which is the inverse of the altered Fisher information matrix. We can obtain this covariance matrix through importance sampling, where the weights (the likelihood ratio) are computed as

$$\mathbf{w}_i = \frac{\exp\{(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^t (\mathbf{g}(\mathbf{Y}_i) - \frac{1}{m} \sum_{j \in (1,m)} \mathbf{g}(\mathbf{Y}_j))\}}{\sum_{i \in (1,m)} \exp\{(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^t (\mathbf{g}(\mathbf{Y}_i) - \frac{1}{m} \sum_{j \in (1,m)} \mathbf{g}(\mathbf{Y}_j))\}} \quad (3.14)$$

so that the diagonal elements of

$$SE^2(\hat{\boldsymbol{\mu}}^*) = \text{cov}_{\hat{\boldsymbol{\theta}}^*}(\mathbf{g}(\mathbf{Y})) = E \left[\left(\mathbf{g}(\mathbf{Y}) - E(\mathbf{g}(\mathbf{Y})) \right) \left(\mathbf{g}(\mathbf{Y}) - E(\mathbf{g}(\mathbf{Y})) \right)^t \right]$$

$$\approx \sum_{i \in (1,m)} \left(\mathbf{g}(\mathbf{Y}_i) - \frac{1}{m} \sum_{j \in (1,m)} \mathbf{g}(\mathbf{Y}_j) \right) \left(\mathbf{g}(\mathbf{Y}_i) - \frac{1}{m} \sum_{j \in (1,m)} \mathbf{g}(\mathbf{Y}_j) \right)^t \mathbf{w}_i \quad (3.15)$$

are the squared standard errors of the mean value parameter components, μ_i .

3.4 Study Design

We compare the performance of the adjusted and non-adjusted estimators for a small (36-node) network. For the chosen model an approximate MLE is calculated for the network. This MLE (found in `ergm` using the stepping method and the lognormal approximation, with `burnin` = 500000 and `interval` = 3000, MCMC sample size 300 for the steps, MCMC sample size 5000 for the final two iterations, and `seed` = 12345) is then treated as the *true underlying parameter* defining a population of possible networks. One thousand networks are then generated from this distribution, and the MPLE, bias-adjusted MPLE (MBLE), approximate MLE, and penalized MLE (pMLE) for the canonical parameters are found. The MPLE and MBLE are found using the option `MPLOnly = TRUE`, and, for the MBLE, `control = control.ergm (MPLEtype = "penalized")`, in the call to the `ergm` function, and the MLE and pMLE are found using the `style = "Stepping"` control option in `ergm`, with `metric = "lognormal"` for the usual MLE and `metric = "penalized.lognormal"` for the pMLE. The mean value parameter estimate corresponding to each canonical parameter estimate is found by simulating 5000 networks from the canonical estimate and taking the mean of the resulting graph statistics.

Perceived standard errors are found for each estimate. We adopt the use of the term “perceived standard errors” from van Duijn et al. (2009) to indicate that the use of asymptotic results in this context is suspect. For the canonical parameters, these are the asymptotic standard errors from the standard software output, for those methods that are available in standard software. In the `ergm` package, this covariance matrix is found and inverted to find the Fisher information matrix (the diagonals of which are the asymptotic standard errors for the canonical parameters). Hence the mean value parameter standard errors can also be found as the diagonals of the inverse of the covariance of the canonical parameters.

As in van Duijn et al. (2009), perceived 95% confidence intervals are constructed using the perceived standard error estimates and a multiplier from a t-distribution where the degrees of freedom are determined by the number of undirected node pairs less the number of parameters to be estimated. In van Duijn et al. (2009), these confidence intervals have coverage close to the nominal 95% coverage for the MLE (slightly liberal for the canonical parameters and moderately liberal for the mean value parameters), but they are surprisingly conservative for the MPLE and MBLE in the canonical parameterization, and extremely liberal in the mean value parameterization, with the MBLE intervals having coverage consistently as high as or higher than those of the MPLE in both parameterizations.

Relative efficiency is calculated as the ratio of the mean squared error (MSE) for each method to the MSE of the approximate MLE. A relative efficiency less than one indicates an reduction in mean squared error compared to the MLE.

3.5 Description of the data and model

These estimators are compared in a lawyer collaboration network. For this network, we duplicate the results found in van Duijn et al. (2009) on the corporate lawyer partnerships and include new analysis for the penalized maximum likelihood estimator introduced to ERGMs in this paper in order to compare our present extended results to the results found for the MBLE and the ordinary MPLE and MLE.

These data are from a 1988-1991 survey of the multiplex relationships among 71 attorneys (partners and associates) in the New England law firm *Spencer, Grace & Robbins*, collected by Emmanuel Lazega and analyzed in regard to the corporate interactions of production and exchange systems, lateral control regimes, and regulatory mechanisms in his 2001 book, *The collegial phenomenon: the social mechanisms of cooperation among peers* (Lazega, 2001). The relationships studied in this work include the collaboration network of the attorneys (“With which members of your firm have you spent time together on at least one case, have you been assigned to the same case, have they read or used your work product or have you have read or used their work product?”), their basic

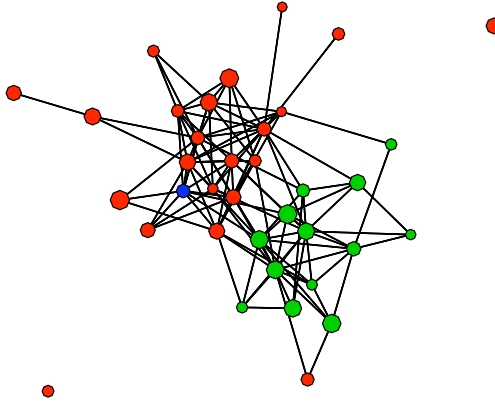


Figure 3.2. The Lazega collaboration network for the 36 partners, with the size of the nodes proportional to the partner’s seniority at the firm, and with the nodes colored according to the primary office location of the partner. The Boston partners are shown in red and include the two isolates. The Hartford partners are in green; and the lone partner in Providence is shown in blue.

advice network (“In the past year, to whom did you go for basic professional advice?”), and their friendship network (“With which other members of the firm do you socialize outside of work?”). Additional member attributes collected include the attorneys’ gender, age, status (36 are partners; 35 are associates), seniority, years with the firm, practice (litigation or corporate), office location (Boston, Hartford, or Providence), and law school attended (Yale or Harvard, University of Connecticut, or any other).

As in van Duijn et al. (2009), we focus on the collaboration network of the 36 partners at *SG&R*. An illustration of this network is shown in Figure 3.2. We use a model with edges, the geometrically-weighted edgewise shared partner statistic with a fixed weight parameter of 0.7781, individual (nodal) covariate effects for seniority and for practice, and homophily (nodes sharing the characteristic) effects for practice, gender, and office. These data are also analyzed in Snijders et al. (2006).

3.6 Results

Figure 3.3 shows the least bias occurring for the MLE in the canonical parameterization for Edges. The adjusted MLE overcorrects the bias of the MLE. In the mean value parameterization, rather unexpectedly, given that the MLE should have no bias here,

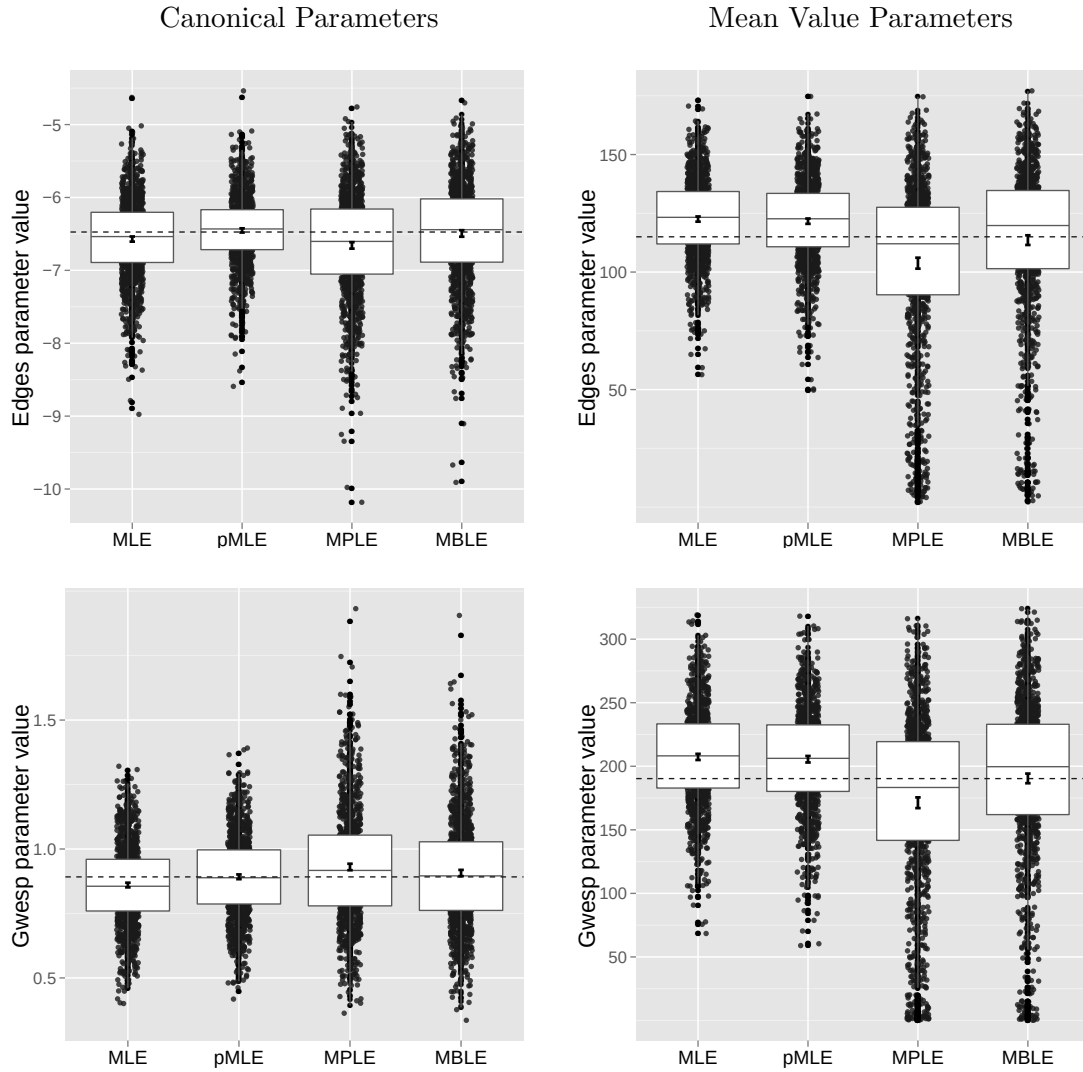


Figure 3.3. Boxplots of the *Edge* and *Gwesp* parameters estimated by each of the four methods for the same 1,000 random networks, each network generated from a known underlying parameter (denoted by dashed line). Bootstrapped 95% confidence limits for the means of each method are also shown.

the MPLE and MBLE have less bias than the MLE or the adjusted MLE. The mean value parameterization plot for the GWESP again shows the least bias occurring for the MPLE.

As seen in Figure 3.4, the MPLE and the MBLE have the least bias for Nodal Seniority and Nodal Practice in either parameterization. The adjusted MLE again overcorrects for the bias of MLE in the canonical parameterization for Nodal Seniority, but for Nodal

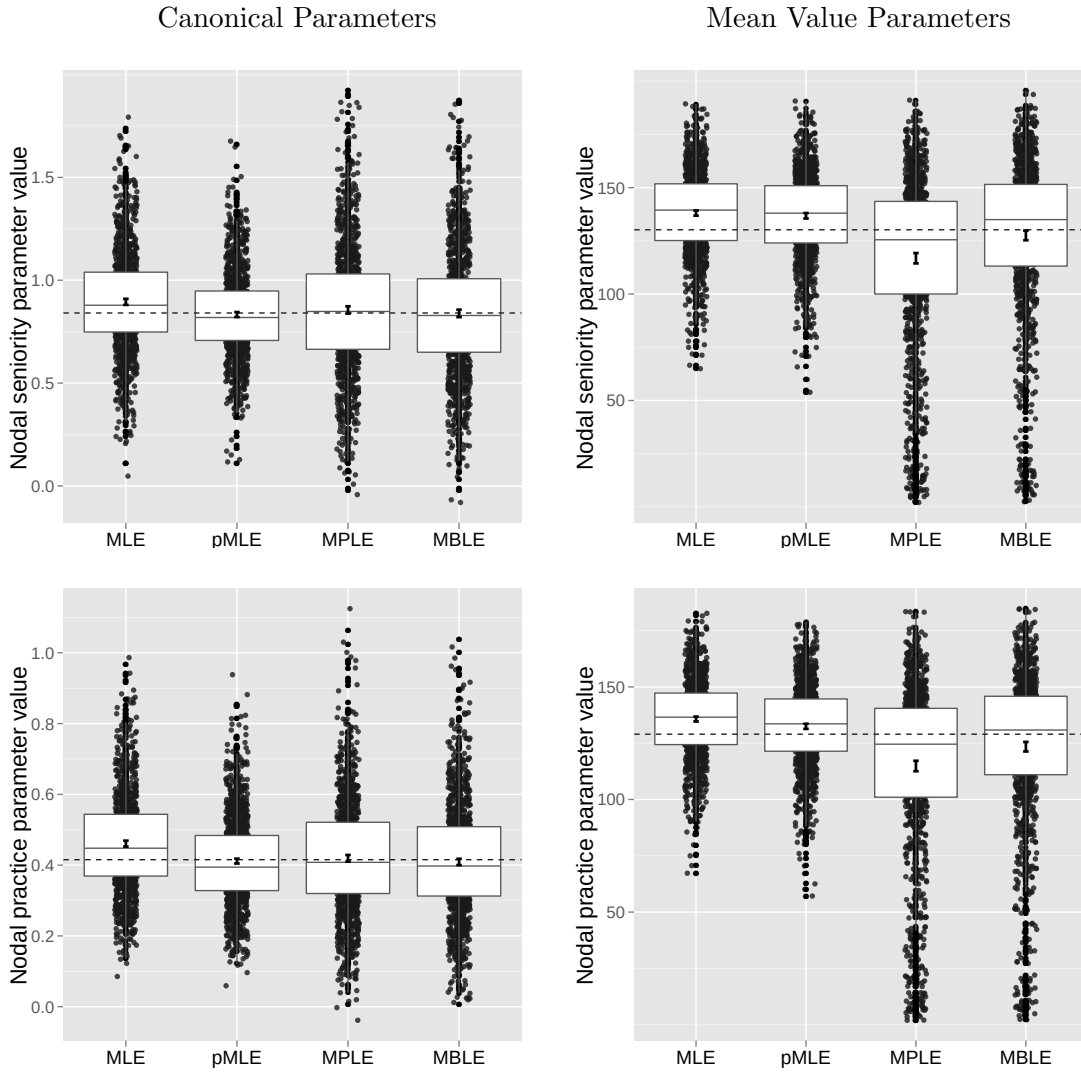


Figure 3.4. Boxplots of the nodal covariate parameters for *Seniority* and *Practice* estimated by each of the four methods for the same 1,000 random networks, each network generated from a known underlying parameter (denoted by dashed line). Bootstrapped 95% confidence limits for the means of each method are also shown.

Practice and in the mean value parameterization for both Nodal effects, the adjusted version does appropriately correct some bias in the MLE.

In Figure 3.5, we can see that while MPLE has least bias for Practice Homophily, and Gender Homophily in the canonical parameterization, it is only by a small margin over the adjusted MLE, which has significantly reduced variance in comparison to the MPLE. For Office Homophily, the adjusted MLE has the least bias and the least spread. In the

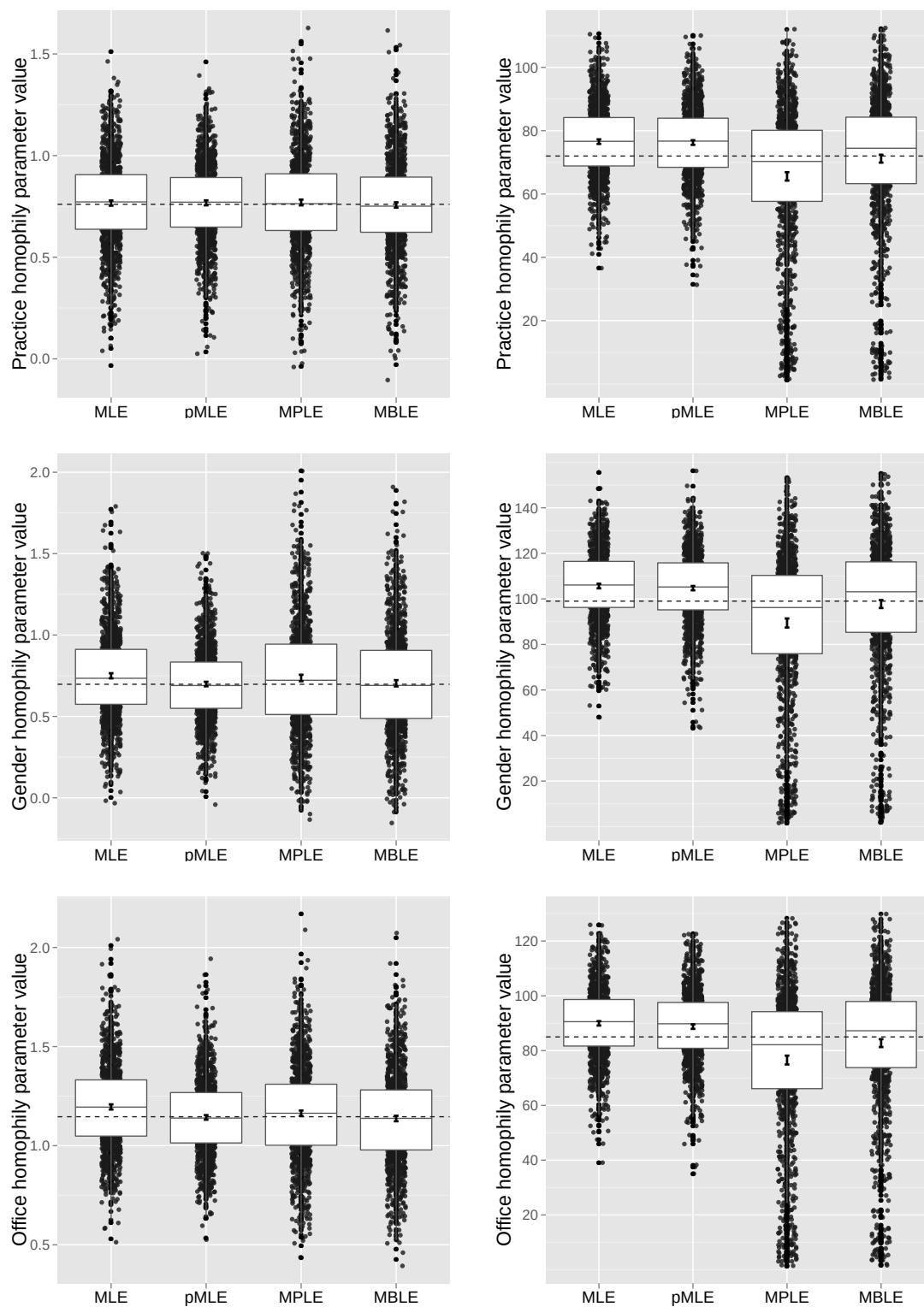


Figure 3.5. Boxplots of the homophily parameters for *Practice*, *Gender*, and *Office* estimated by each of the four methods for the same 1,000 random networks, each network generated from a known underlying parameter (denoted by dashed line). Bootstrapped 95% confidence limits for the means of each method are also shown.

Table 3.1. Coverage rates of perceived (95%) t-confidence intervals for the approximate MLE, the bias-adjusted approximate MLE (pMLE), the MPLE, and the bias-adjusted MPLE (MBLE).

	canonical parameter				mean value parameter			
	MLE	pMLE	MPLE	MBLE	MLE	pMLE	MPLE	MBLE
<i>structural</i>								
edges	75.10	81.50	98.40	98.40	89.40	91.65	47.15	46.70
Gwesp	93.00	93.30	75.50	75.50	89.65	91.40	59.55	59.25
<i>nodal</i>								
seniority	20.80	20.70	97.80	97.90	99.95	99.85	47.05	46.95
practice	34.40	36.30	98.40	98.50	100.00	99.85	54.00	57.95
<i>homophily</i>								
practice	34.00	34.60	97.30	97.40	99.95	99.95	54.10	53.60
gender	21.30	22.70	98.70	98.80	99.85	99.85	45.90	47.05
office	27.80	31.40	98.30	98.10	99.95	99.90	49.35	52.20

Table 3.2. Relative efficiency (as the ratio of mean squared errors) of the bias-adjusted approximate MLE (pMLE), the MPLE, and the bias-adjusted MPLE (MBLE), compared to the MLE.

	canonical parameter			mean value parameter		
	pMLE	MPLE	MBLE	pMLE	MPLE	MBLE
<i>structural</i>						
edges	0.70	1.02	1.64	3.92	1.47	2.93
Gwesp	0.98	1.01	1.92	2.85	1.76	2.27
<i>nodal</i>						
seniority	0.71	1.00	1.60	3.97	1.54	2.94
practice	0.67	0.97	1.24	4.71	1.18	3.39
<i>homophily</i>						
practice	0.84	1.03	1.13	3.60	1.08	2.73
gender	0.70	1.00	1.51	3.65	1.39	2.76
office	0.76	0.97	1.23	3.93	1.16	2.90

mean value parameterization for these terms, the MPLE and MBLE seem to perform with much less bias than the MLE and adjusted MLE.

Overall, the variance of the MLE and adjusted MLE are considerably smaller than the variances of the MPLE and the MBLE. However, the bias does not seem to be mitigated very well by the adjustment of the MLE as it often overcorrects, and these results show that the MPLE and its bias-adjusted version have the least bias in more

Table 3.3. Asymptotic standard errors for the approximate MLE, the bias-adjusted approximate MLE (pMLE), the MPLE, and the bias-adjusted MPLE (MBLE).

	canonical parameter				mean value parameter			
	MLE	pMLE	MPLE	MBLE	MLE	pMLE	MPLE	MBLE
<i>structural</i>								
edges	0.33	0.34	0.26	0.33	0.29	0.31	0.32	0.31
Gwesp	0.34	0.33	0.36	0.33	0.27	0.38	0.31	0.33
<i>nodal</i>								
seniority	0.33	0.34	0.29	0.35	0.28	0.31	0.28	0.33
practice	0.32	0.28	0.33	0.28	0.31	0.27	0.39	0.31
<i>homophily</i>								
practice	0.32	0.13	0.30	0.27	0.29	0.31	0.33	0.34
gender	0.30	0.27	0.32	0.30	0.35	0.29	0.26	0.30
office	0.29	0.28	0.26	0.35	0.32	0.32	0.31	0.28

model terms and across parameterizations than do the MLE and its adjustment. This is quite a surprise, and contradicts the results found in van Duijn et al. (2009) on the same data.

Confidence interval coverage rates for the t intervals are shown in Table 3.1, and relative efficiency is in Table 3.2. Estimators with relative efficiency less than one had reduced mean squared error in comparison to the MLE.

We might expect the variance of the unadjusted MLE to come closer to the Cramer-Rao lower bound, since the unadjusted log-likelihood-ratio approximation function will be a more accurate approximation to the true log-likelihood-ratio function at its maximizer, and the true function will have exactly the curvature described by the Cramer-Rao lower bound. However, judging by the efficiency ratios, it seems that, at least in this example, the adjusted approximate MLE is not plagued by excessive additional variation compared to the unadjusted approximate MLE.

The computational details used here are similar to those in van Duijn et al. (2009); however, some minor differences exist. In van Duijn et al. (2009), each of the 1,000 networks generated from the fixed “true” parameter (the MLE for the original Lazega network) is subject to two or three stages of model fitting, each including two to four sub-iterations (defined by `maxit`). These stages and sub-iterations are an attempt to

mitigate the impact of generating a sample far from the observed statistics, on both the estimate itself and on its estimated covariance. Instead, we adopt the stepping method introduced in Chapter 2 and forego these additional stages and iterations entirely. At the final step of estimation we use the same `burnin` (500000), `interval` (3000), and `MCMCsamplesize` (5000) as are used in van Duijn et al. (2009). We use samples of size 300 for the sample size for the preliminary steps.

3.7 Discussion

Bias-reduction is not always an improvement over an existing estimator. For example, Firth showed that bias-modification in the score function for the curved exponential family of $N(\mu, \mu^2)$ distributions results in an inflation of the asymptotic variance approximately three times the size of the reduction achieved in squared bias (Firth, 1993). However, in general, the estimator derived through the Firth modification to the score function exhibits several attractive properties: this estimator can be found without requiring that the maximum likelihood estimate be finite, bias can be reduced for a wide variety of model classes (such as curved exponential families), and, for exponential family models in their canonical parameterization, the Firth penalty is the Jeffreys prior.

The non-necessity of a finite MLE is of considerable use in the ERGM for cases in which the observed sample statistics are on the boundary of the convex hull of the sample space. In these cases, the MLE is infinite. For such situations, the Firth adjustment provides an estimate with shrinkage toward log-odds of one, an adjustment with a history of use in estimating probabilities and modeling logistic regression for rare events (Heinze and Schemper, 2002).

Bias-correction properties that extend from canonically-parameterized ERGMs to ERGMs with curved parameterizations can be widely applied in the statistical modeling of networks, particularly as they allow comparison of the same likelihood-based estimator for a variety of possible models, in particular through likelihood ratio tests of goodness of fit between more and less restricted versions of the same model.

It is also convenient that the properties of the posterior mode resulting from the

Jeffreys prior can be applied to the adjusted MLE, and vice versa. For example, the asymptotic bias-reduction property of the adjusted MLE guarantees that the Jeffreys prior posterior mode will have second-order bias reduction for canonical exponential family models and first-order bias reduction for other standard families of models. Likewise, the invariance property of the Jeffreys prior estimate is applicable to the maximizer of the Firth-penalized likelihood.

The connection between bias reduction and adjusted profile likelihood is still a largely unexplored field, as suggested in Firth (1993), Barndorff-Nielsen (1983), and Cox and Reid (1987). Ehm (1991) discusses this connection for problems with many parameters in the context of estimation from the Jeffreys prior, and Li (1998) presents strong evidence that the second-order mean-squared-error optimality of the Firth penalty results from the additional cumulant identity, $\kappa_{r,st} = 0$ for all r , s , and t , for canonical exponential family models, but a theoretical exploration of the application to profile likelihood of bias reduction via a penalized likelihood has yet to be undertaken. Composite likelihood, which will appear in some detail in the next chapter, can be a variation of profile likelihood, and is another mode of conditional inference on which the Firth penalty has not yet been explicitly studied. These related likelihood functions and the effects of the Firth penalty on estimation through them may prove useful for parameter estimation in ERGMs for situations well-described using profile or composite likelihoods, e.g., networks with complicated parameter restrictions or structural dependencies between dyads.

In the simulations shown in this chapter, as well as in, e.g., van Duijn et al. (2009), the information matrix is being calculated using importance sampling (a weighted sample mean) of the covariance in terms of expected value, where the weights (the likelihood ratio) are computed as in (3.14). Hunter and Handcock (2006) makes the point that this approximation to the likelihood ratio deteriorates as θ gets farther from θ_0 , and suggests bridge sampling as an alternative that might result in a better estimate for the log-likelihood-ratio. This has not yet been implemented in the `ergm` package, but would be interesting to study in the future.

Contrastive divergence likelihood estimation in ERGMs

4.1 Introduction

This chapter introduces a statistical perspective on the emerging application of a popular machine learning technique, *contrastive divergence*, to ERGMs. CD is an iterative optimization technique, essentially minimizing the difference between two Kullback-Leibler divergences through gradient descent, and can be used for many kinds of likelihood maximization. CD has generated a substantial literature in machine learning, where it has been applied to learning the structures of deep belief networks, neural networks such as restricted and unrestricted Boltzmann machines (Hinton, 2000; Asuncion et al., 2010), Markov random fields (Carreira-Perpinan and Hinton, 2005; Asuncion et al., 2010), and conditional random field Ising models and ERGMs (Asuncion et al., 2010; Tieleman and Hinton, 2009). It is the application of CD to model-fitting in ERGMs that is explored in this chapter.

Several statistical results regarding contrastive divergence in ERGMs have been referenced in the literature, and are summarized in Section 2. In Section 3, CD-1 is applied to an ERGM, and the connection between CD-1 and MPLE is discussed. In Section 4, Gibbs sampling in this context is discussed, and efficiency results are derived to compare

full-scan CD to random-scan CD. Section 5 introduces the class of functions called composite likelihoods, and shows how a version of contrastive divergence methods can be used to find the composite likelihood maximizer. Section 6 applies these methods directly to ERGMs, and Section 7 concludes with some general comments on the application of CD estimation to ERGMs.

4.2 Contrastive divergence

The reader is reminded (as was discussed in Chapters 1 and 2 of this thesis) that the true loglikelihood function for an ERGM involves a sum over all possible graph configurations, a number so prohibitively large, except in rather trivial cases, as to be computationally intractable. For this reason, maximum likelihood estimation can be approximated using MCMC methods. It is not necessary to use MCMC for MPLE, but here we will show how MPLE may be approximated using a simulation-based technique that puts MPLE and MLE at opposite ends of a continuum of methods.

The CD algorithm maximizes a likelihood function by taking samples from the current model, calculating a gradient based on these samples and the original network, optimizing the function parameters using this estimated gradient, and iterating this process. When CD is optimized for a sample consisting of networks updated for only one iteration of a Gibbs sampler, this process is equivalent to a stochastic MPLE optimization (Hyvarinen, 2006, 2007). If the sampler is run ad-infinitum (i.e., until the Markov chain converges, assuming it can), optimizing the CD function based on these samples is equivalent to a stochastic MLE optimization (Carreira-Perpinan and Hinton, 2005).

In Asuncion et al. (2010), Contrastive Divergence techniques are considered as points in a continuum with endpoints the MPLE and the MCMC MLE. In such a continuum we may envision an intermediate estimator that maximizes the estimated loglikelihood using an MCMC sample generated using exactly n steps of the sampler. This estimator would presumably be some sort of an intermediary between the maximizer assuming the specified model dependence structure (the MLE) and the maximizer assuming dyad independence (the MPLE). Of course, if dyad independence is the structure proposed

by the model, then the MLE, the MPLE, and any intermediaries will all be equal. If the model specifies any dyadic dependence, however, this intermediary estimator might prove to be somehow “closer” to the MLE than the MPLE would be.

We now describe this continuum relationship in regard to the full likelihood and the pseudolikelihood. The full loglikelihood can be written:

$$\ell(\boldsymbol{\theta}) = \log P(\mathbf{Y} = \mathbf{y}^{obs}) = \boldsymbol{\theta} \mathbf{g}(\mathbf{y}^{obs}) - \kappa(\boldsymbol{\theta}) \quad (4.1)$$

By choosing to maximize the full likelihood function, we obtain asymptotic consistency and normality of the estimator (Lehman and Casella, 1998). This variance is optimal, as the MLE achieves the Cramer-Rao lower bound (and therefore no other consistent estimator has lower asymptotic variance than the MLE).

We can approximate this MLE using Markov chain Monte Carlo methods. The idea of MCMC MLE consists of randomly sampling networks $Y_1, \dots, Y_S$ from the distribution determined by some fixed $\boldsymbol{\theta}_0$ using MCMC, reweighting the sample through importance sampling, and taking the empirical distribution of these reweighted networks as an approximation to the distribution under $\hat{\boldsymbol{\theta}}$. These networks can be generated using Gibbs sampling, where the presence of an edge joining two nodes is updated, for any pair of nodes, with a probability determined by the status of the dyads in the rest of the graph and the fixed parameter value, $\boldsymbol{\theta}_0$. Each sampled network involves exactly n Gibbs updates from the original state of the network. In estimating the $\hat{\boldsymbol{\theta}}$ distribution through MCMC, there is a tradeoff between n and S , the number of Gibbs updates per sampled network and the number of sampled networks. One possible objective of studying the reduction of n (which we do here through contrastive divergence) is to reach a practical balance between accuracy of estimation and computational burden.

Maximizing the loglikelihood for the true (not approximate) likelihood function is equivalent to minimizing the Kullback-Leibler divergence between the observed (empirical) data distribution $P_0(\mathbf{y})$ and the true model distribution $P_\infty(\mathbf{y})$. In the case of only a single observed network (as examined in this chapter and throughout this thesis), the observed data distribution is simply a point mass at $\mathbf{y}^{obs}$. When we es-

timate the MLE using MCMC, the problem of maximum likelihood estimation can now be framed as minimizing the difference between two KL divergences: $CD_n = \operatorname{argmin}\{KL(P_0||P_\infty) - KL(P_n||P_\infty)\}$, where $P_0(\mathbf{y})$ is the distribution of the observed data, $P_\infty(\mathbf{y})$ is the (true stationary) model distribution, and $P_n(\mathbf{y})$ is the distribution of the n -step MCMC samples (Carreira-Perpinan and Hinton, 2005). This formulation of the problem is the source of the term *contrastive divergence*. The resulting contrastive divergence estimate is described by the number of steps used in each iteration of the sampler, i.e., CD- n for the minimizer of the difference between the KL divergence of the observed data from the (unknown) MLE and the KL divergence of the distribution of the n -step sample and the MLE. This optimization problem can be described as finding a zero of the function

$$\nabla CD_n = E_0(\mathbf{g}(\mathbf{Y})) - E(\mathbf{g}(\mathbf{Y})) = \mathbf{g}(\mathbf{Y}^{\text{obs}}) - E_n(\mathbf{g}(\mathbf{Y})), \quad (4.2)$$

where $E_0(\mathbf{g}(\mathbf{Y}))$ denotes the expectation of the statistics under the observed data distribution and $E_n(\mathbf{g}(\mathbf{Y}))$ denotes this expectation under the distribution of the n -step MCMC samples. Hence, finding the CD optimizer can be approximated by optimizing the naive approximation to the log-likelihood-ratio function presented in Chapter 2, as long as the $\mathbf{Y}_i$ are sampled using an n -step Gibbs sampler that always starts from the original observed network:

$$\ell(\boldsymbol{\theta}) - \ell(\boldsymbol{\theta}_0) \approx (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\top \mathbf{g}(\mathbf{y}^{\text{obs}}) - \log \left[\frac{1}{S} \sum_{i=1}^S \exp \left\{ (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\top \mathbf{g}(\mathbf{Y}_i) \right\} \right]. \quad (4.3)$$

Equivalently, CD can be written as a stochastic approximation algorithm using MCMC to approximate the expectation under P_n in the steepest descent equation (Younes, 1999), with λ_t a parameter of the optimization and t from 1 to T indexing the iteration:

$$\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t = \lambda_t \left[\nabla_{\boldsymbol{\theta}} [\ell(\boldsymbol{\theta}_t) - \ell(\boldsymbol{\theta})] \right]. \quad (4.4)$$

For $T \rightarrow \infty$ this is equivalent to iterating an approximate Fisher scoring method to find

the maximizer of the approximation (4.3) (Hyvarinen, 2007).

Both the MCMC-MLE and the CD- n algorithms are initiated by taking S samples, $\mathbf{y}^1, \dots, \mathbf{y}^S$, from MCMC chains defined by updating the dyads of the original observed network according to Gibbs samplers with update probabilities corresponding to the starting value, $\boldsymbol{\theta}_0$.

For MCMC-MLE, the S samples should come from well-burned-in chains (i.e., using *many* steps of the sampler), and the maximizer, $\boldsymbol{\theta}_1$, of the log-likelihood-ratio function found. Here only *one* such MCMC run for the S samples is necessary (although the process could be repeated for numerical stability, if desired).

For CD- n , the S samples will come from very short (n -step) chains of the sampler. Typically, rather than maximizing the log-likelihood-ratio function, the gradient is calculated and the optimizer, $\boldsymbol{\theta}_1$, found; however, this is a point of practice and not a definitive difference between methods (Hyvarinen, 2007). For CD- n using gradient descent, this process is repeated T times, always beginning the chains again at the observed network, but substituting the new $\boldsymbol{\theta}_t$ in the Gibbs update probabilities. Hence the estimator $\boldsymbol{\theta}_1$ is the final estimate in the MCMC MLE algorithm, but is only the first of T iterations in CD- n using gradient descent.

Yuille (2005) derives conditions which ensure convergence of exact CD (CD- ∞) to the parameters $\boldsymbol{\theta}$ that minimize the Kullback-Leibler divergence between the observed data $P_0(\mathbf{y})$ and the model $P_{\boldsymbol{\theta}}(\mathbf{y})$: that each expected update must move in the direction of the true gradient, that the expected squared magnitude of the update is bounded, and that the damping coefficients (the λ_t in equation 4.4) decrease steadily with time, at a rate not faster than $\lambda_t = 1/t$. Yuille also claims that these are necessary and sufficient conditions for the gradient descent CD solution to be unbiased for the maximizer of its objective function, deriving these results from more general results for stochastic approximation algorithms, as in Kushner and Clark (1978), Kushner (1987), Younes (1999), and Orr and Leen (1993). In practice, however, such convergence results may be worthless since sample sizes required for good approximation can be extremely large (as in the simple example shown in Figure 2.1 of Chapter 2).

CD- n can also substitute the log-likelihood-ratio function as the objective function.

In this case, we can even apply the stepping algorithm of Chapter 2 directly to CD estimation. Now the relationship between CD- n and MCMC-MLE is very clear: CD- n is the very same optimization routine as MCMC-MLE, but performed with *exactly* n steps for the MCMC chains, with no assumption that the MCMC samples are well-approximating the desired distribution.

Carreira-Perpinan and Hinton (2005) show that the fixed points of CD- n and MCMC-MLE are frequently not the same, and they call this difference the *bias* of CD- n . They show that the set of data distributions in which the fixed points will be equal will have measure zero for some models, and argue that this will be the case for “most” models. Note that this “bias” results from two differences. First, when CD is estimated via equation (4.4), there may not be convergence to the optimizer of approximation (4.3) if T is not sufficiently large. Second, the objective functions for CD- n and MCMC-MLE are not the same: while MCMC-MLE assumes that the sample mean in approximation (4.3) will be taken with respect to $\hat{\theta}$, the CD- n mean is expected to come from the distribution described by only n steps of the MCMC sampler. These distributions may not have the same fixed points. However, Carreira-Perpinan and Hinton (2005) also show empirical evidence that this difference between the CD- n and MCMC-MLE estimators seems to be small, especially relative to the gain in computational simplicity. They suggest appending a few steps of ML to the end of the CD scheme, thereby reducing the bias in the CD estimate (versus MCMC-MLE) but keeping the total computational time still considerably below that of pure MCMC-MLE.

4.3 Pseudolikelihood example

Proposed by Besag (1974) for spatial models, advanced by Besag (1977) and Gong and Samiengo (1981), and adapted by Strauss and Ikeda (1990) to social network models, the pseudolikelihood estimate maximizes the loglikelihood function for the case when the existence of an edge for a randomly chosen dyad is independent of any information contained in the rest of the graph. That is, if the marginal probability of an edge is the same as the conditional probability, conditioning on the rest of the graph, the

loglikelihood function could be written as

$$p\ell(\boldsymbol{\theta}) = \sum_i \sum_{j \neq i} \log P(Y_{ij} = 1 | \mathbf{Y}_{ij}^c = (\mathbf{y}^{\text{obs}})^c_{ij}) \quad (4.5)$$

where the sum is over all the possible dyads (i, j) and y_{ij}^c indicates all of the information contained in the network, $\mathbf{y}$, except the information associated with nodes i and j . When this independence assumption does not hold, the maximizer of this function can still be found, and it is called the *maximum pseudolikelihood estimator*, or *MPLE*, to emphasize that this loglikelihood is no longer the true loglikelihood for the data.

As CD- n will be maximizing a version of the loglikelihood ratio function that replaces the MLE distribution with the distribution defined by an n -step MCMC chain, the CD-1 estimate will be the maximizer of

$$\ell(\boldsymbol{\theta}) - \ell(\boldsymbol{\theta}_0) \approx (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\top \mathbf{g}(\mathbf{y}^{\text{obs}}) - \log \left[\frac{1}{S} \sum_{i=1}^S \exp \left\{ (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\top \mathbf{g}(\mathbf{Y}_i) \right\} \right],$$

as long as the $\mathbf{Y}_i$ are sampled using a 1-step Gibbs sampler that always starts from the original observed network — that is, the sampled networks differ from the observed by no more than a single toggle.

To show that CD-1 is a stochastic approximation to the MPLE, we can approximate the pseudo-log-likelihood as:

$$p\ell(\boldsymbol{\theta}) \approx \sum_i \sum_{i \neq j} \log \frac{1}{S} \sum_s I(Y_{s;ij} = 1 | \mathbf{Y}_{ij}^c = \mathbf{y}^{\text{obs}^c}_{ij}), \quad (4.6)$$

where the conditional probabilities can be estimated by summing the number of occurrences of edges in the (i, j) position for S MCMC samples of Gibbs probabilities defined by the desired conditional probabilities (explicit below in step (1)), and divided by S . This is clearly unnecessary to do to maximize the pseudolikelihood, as we already know the conditional probabilities (and are in fact using them to generate the MCMC sample); however, by expressing the pseudo-log-likelihood this way, we demonstrate that this relationship between estimating the expectation (or probability) in a likelihood by taking means of MCMC samples approximates the MLE when the sampler draws from

the true (dependence) joint likelihood, and it approximates the MPLE when the sampler draws from the independence joint likelihood (that is, from the product of the conditional likelihoods).

A CD-1 algorithm for ERGMs can be summarized as:

1. Choose (randomly or systematically) a dyad from the set of nodes, where each dyad has equal selection probability (or occurs exactly once in the systematic scheme). Update this dyad using Gibbs probabilities defined by

$$P(y_{ij}^{\text{obs}} = 1 | (\mathbf{y}_{ij}^{\text{obs}})^c, \boldsymbol{\theta}) = \frac{\exp\{\boldsymbol{\theta}^\top \boldsymbol{\delta}_g(\mathbf{y}^{\text{obs}})_{ij}\}}{1 + \exp\{\boldsymbol{\theta}^\top \boldsymbol{\delta}_g(\mathbf{y}^{\text{obs}})_{ij}\}},$$

where $\boldsymbol{\delta}_g(\mathbf{y})_{ij}$ is the change in $\mathbf{g}(\mathbf{y})$ when y_{ij} is changed from 0 to 1. The vector of change statistics, $\boldsymbol{\delta}_g(\mathbf{y})_{ij}$, is the conditional log-odds that $Y_{ij} = 1$ given the rest of the network. (See Section 4.4.1 for more details about this probability.)

2. Continue to choose and update n dyads according to these probabilities.
3. Do this S times, always beginning with the original network.
4. Use these S samples to approximate the gradient of the log-likelihood-ratio function of equation (4.3).
5. Update the parameter as in equation (4.4), with $\lambda_t = 1/t$.
6. Repeat this process for T iterations.

We demonstrate this for the collaboration network of the 36 partners at *SG&R*, discussed in Chapter 3 (see Figure 4.1). These data are from a 1988-1991 survey of relationships among attorneys in the New England law firm *Spencer, Grace & Robbins* (Lazega, 2001). The collaboration network is determined by the attorneys' responses to the question "With which members of your firm have you spent time together on at least one case, have you been assigned to the same case, have they read or used your work product or have you have read or used their work product?" Additional member attributes collected include the attorneys' gender, age, seniority, years with the firm, practice (litigation or corporate), office location (Boston, Hartford, or Providence), and

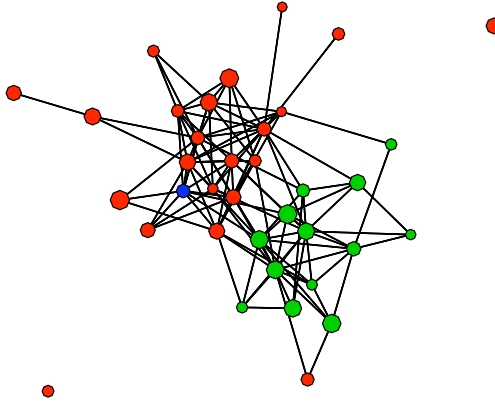


Figure 4.1. The Lazega collaboration network for the 36 partners, with the size of the nodes proportional to the partner's seniority at the firm, and with the nodes colored according to the primary office location of the partner. The Boston partners are shown in red and include the two isolates. The Hartford partners are in green; and the lone partner in Providence is shown in blue.

law school attended (Yale or Harvard, University of Connecticut, or any other). As in the analysis of this network in Chapter 3, we use a model with edges, the geometrically-weighted edgewise shared partner statistic with a fixed weight parameter of 0.7781, individual (nodal) covariate effects for seniority and for practice, and homophily (nodes sharing the characteristic) effects for practice, gender, and office.

Table 4.1. Simulation arguments used for CD- n estimation using the `ergm` function in the `ergm` package (version 2.3) in R.

	CD-1	CD-n	MCMC MLE
<i>simulation arguments</i>			
<code>MCMCsamplesize</code>	50000	50000	5000
<code>burnin</code>	1	n	500000
<code>interval</code>	1	n	3000
<code>sequential</code>	FALSE	FALSE	FALSE
<code>seed</code>	65489	65489	65489
<i>arguments to control.ergm for the optimization method and iteration steps</i>			
<code>style</code>	"Stepping"	"Stepping"	"Stepping"
<code>metric</code>	"naive"	"naive"	"naive"
<code>stepMCMCsize</code>	500	500	500

All methods were found using the `ergm` function in the `ergm` package (Handcock

et al., 2010, version 2.3) with the arguments specified in Table 4.1. The MPLE is found in `ergm` by specifying `MPLEonly = TRUE`. The MCMC-MLE is found in `ergm` using the stepping method and the naive approximation, with `burnin = 500000` and `interval = 3000`, the zero vector as the starting value, θ_0 , MCMC sample size 500 for the steps, and MCMC sample size 5000 for the final two iterations. The CD-1 estimator is found in `ergm` by simulating $S = 50,000$ samples with `burnin = 1`, `interval = 1`, and the zero vector as θ_0 . These 1-step samples are then used to optimize equation (4.3). This can be accomplished through gradient descent, as in equation (4.4), or, as mentioned above, through a direct approach of maximization using the stepping algorithm introduced in Chapter 2. In this simulation at each iteration we use a call to the stepping method with the naive approximate CD- n -based likelihood instead of a single step of gradient descent, as in the following code.

```
> load ("LazegaNet.RData")
> LazFormula <- LazegaNet ~ edges + gwesp(0.7781, fixed=T) +
  nodecov("seniority") + nodefactor("practice") +
  nodematch("practice") + nodematch("gender") +
  nodematch("office")

> MPLEfit <- ergm(LazFormula, theta0='MPLE', MPLEonly = TRUE)

> MLEfit <- ergm(LazFormula, theta0=rep(0,7), maxit=1,
  MCMCsamplesize=5000, burnin = 500000,
  interval=3000, sequential=FALSE, seed=65489,
  control=control.ergm(style="Stepping", metric="naive",
  stepMCMCsize=500))

> # CD-n fit using TT=12 iterations and S=50,000 MCMC
  samples for specified n #
> TT <- 12; S <- 50000; n <- 1; coef <- rep(0, 7)
> for(i in 1:TT) {
  LazCDfit<-ergm(formula=LazFormula, theta0=coef,
  MCMCsamplesize=S, burnin, interval,
  sequential=FALSE, seed=65489,
  control=control.ergm(style="Stepping", metric="naive"
  stepMCMCsize=500))
  print(coef <- LazCDfit$coef)
  cat("----- end ", i, " -----\n")
}
> summary(LazCDfit)
```


Other CD- n estimators are found in `ergm` as for CD-1, with `burnin = n` and `interval = n`. These simulation results (estimates and MCMC standard errors) are displayed in Table 4.2. The MCMC standard error values are quite high for small values of n , but this can be controlled by increasing the value of S . So-called “asymptotic” standard errors (usually based on the Hessian of the objective function) are not reported for the CD- n estimates, as the objective function may be linear in some directions for CD- n . Computation of these standard errors, particularly for small n , is an area for future research.

Table 4.2. Simulation results for parameter estimation using MPLE, CD-1, CD-2, CD-5, and MCMC-MLE (CD- ∞), with MCMC standard errors. (Usual standard errors are reported for the MPLE in square brackets.)

model term	MPLE	CD-1	CD-2	CD-3	CD-5	MCMC MLE
<i>structural</i>						
edges	-6.43 [.81]	-5.73 (2.7)	-5.78 (.63)	-6.01 (.46)	-6.41 (.28)	-6.30 (.30)
Gwesp	0.90 [.11]	0.66 (.85)	0.88 (.22)	0.89 (.16)	0.90 (.10)	0.89 (.13)
<i>nodal</i>						
seniority	0.89 [.36]	1.02 (.13)	0.17 (.03)	0.55 (.02)	0.69 (.01)	0.77 (.01)
practice	0.38 [.19]	0.25 (.18)	0.41 (.01)	0.43 (.01)	0.33 (.01)	0.38 (.01)
<i>homophily</i>						
practice	0.72 [.25]	0.83 (.07)	0.66 (.01)	0.55 (.01)	0.77 (.01)	0.74 (.01)
gender	0.64 [.39]	1.08 (.10)	0.67 (.01)	0.57 (.01)	0.65 (.01)	0.66 (.01)
office	1.15 [.28]	2.08 (.07)	1.31 (.02)	1.45 (.01)	1.40 (.01)	1.12 (.01)

From Table 4.2 we can see that for this model the CD-1 parameter estimate is very close to the MPLE estimate. This is as expected, since the CD-1 estimate should be a stochastic approximation to the MPLE. However, the significance of this result is not entirely intuitive. This example is evidence that starting with a completely naive parameter value and running chains of length 1 in an MCMC-MLE algorithm can give a meaningful result. Further examples (particularly with greater difference between the MPLE and the approximate MLE) will be explored in the future.

4.4 Gibbs sampling in contrastive divergence

Here we develop further the Gibbs sampling details of contrastive divergence. For simplicity we derive these results for the CD-1 case — the case where we only use one step of the Gibbs sampler, equivalent to the MPLE — but the extension to an increased number of steps in the sampler, CD- n , requires only repeating the sampling scheme (after replacing the observed network with the most recent update) for a total of n iterations before finding the estimated derivative and then repeating this entire scheme for T iterations.

4.4.1 Formulation of two sampling schemes: random scan and full scan

The pseudo-log-likelihood for an undirected network can be written as

$$p\ell(\boldsymbol{\theta}|\mathbf{y}^{obs}) = \log \prod_i \prod_{j \neq i} P(y_{ij}^{obs} | (\mathbf{y}_{ij}^{obs})^c, \boldsymbol{\theta}),$$

the log-product of Bernoulli probability mass functions. For a bernoulli random variable X with $P(x) = \alpha^x(1-\alpha)^{1-x} = \exp\{\lambda x\}$, the relationship between the α -parameterization and the λ -parameterization is that $\lambda = \text{logit}(\alpha)$, $\alpha = \text{logit}^{-1}(\lambda)$. Hence,

$$P(y_{ij}^{obs} = 1 | (\mathbf{y}_{ij}^{obs})^c, \boldsymbol{\theta}) = \frac{\exp\{\boldsymbol{\theta}^\top \boldsymbol{\delta}_g(\mathbf{y}^{obs})_{ij}\}}{1 + \exp\{\boldsymbol{\theta}^\top \boldsymbol{\delta}_g(\mathbf{y}^{obs})_{ij}\}},$$

where $\boldsymbol{\delta}_g(\mathbf{y})_{ij}$, called *the vector of change statistics*, is the change in $\mathbf{g}(\mathbf{y})$ when y_{ij} is changed from 0 to 1. The product $\boldsymbol{\theta}^\top \boldsymbol{\delta}_g(\mathbf{y})_{ij}$ is the conditional log-odds that $Y_{ij} = 1$ given the rest of the network. This fact allows us to simplify the pseudo-log-likelihood:

$$\begin{aligned} p\ell(\boldsymbol{\theta}|\mathbf{y}^{obs}) &= \sum_i \sum_{i \neq j} [\log P_{\boldsymbol{\theta}}(\mathbf{Y} = \mathbf{y}^{obs}) - \log P_{\boldsymbol{\theta}}((\mathbf{Y}_{ij})^c = (\mathbf{y}_{ij}^{obs})^c)] \\ &= \binom{n}{2} \log \left[\frac{\exp\{\boldsymbol{\theta}^\top \boldsymbol{\delta}_g(\mathbf{y}^{obs})\}}{\kappa(\boldsymbol{\theta})} \right] - \sum_{j=1}^n \sum_{k \neq j} \log \sum_{z=0,1} P_{\boldsymbol{\theta}}(\mathbf{Y} = [z, (\mathbf{y}_{ij}^{obs})^c]). \end{aligned}$$

Note the introduction of a new index, z , here. This is indexing the final sum over all of the possibilities for y_{jk} , which in our case are the existence of an edge (1) or its absence (0). The expression $\mathbf{Y} = [z, (\mathbf{y}_{ij}^{obs})^c]$ indicates the network formed by replacing the (i,j) th

element in $\mathbf{y}^{obs}$ with the value z . This z could range over a larger set of possibilities if the network were weighted.

Here the $-\log \kappa(\boldsymbol{\theta})$ at the beginning of the expression cancels with the $\log \kappa(\boldsymbol{\theta})$ implicit in the last term, so we are able to write the pseudo-log-likelihood finally as

$$p\ell(\boldsymbol{\theta}|\mathbf{y}^{obs}) = \binom{n}{2} \left[\boldsymbol{\theta}^t \mathbf{g}(\mathbf{y}^{obs}) - \frac{1}{\binom{n}{2}} \sum_i \sum_{j \neq i} \log \sum_{z=0,1} \exp\{\boldsymbol{\theta}^t \mathbf{g}([z, (\mathbf{y}_{ij}^{obs})^c])\} \right]$$

Random Scan: If we now consider sampling by choosing a dyad pair (I, J) uniformly at random and then updating this dyad according to the probability model for the current $\boldsymbol{\theta}$, the gradient of the pseudo-log-likelihood under this scheme can be expressed as

$$\begin{aligned} \nabla p\ell &= C - \sum_i \sum_{j \neq i} \left[\frac{\kappa(\boldsymbol{\theta})}{\sum_{z=0,1} \exp\{\boldsymbol{\theta} \mathbf{g}([z, (\mathbf{y}_{ij}^{obs})^c])\}} \sum_{z=0,1} \frac{\mathbf{g}([z, (\mathbf{y}_{ij}^{obs})^c]) \exp\{\boldsymbol{\theta} \mathbf{g}([z, (\mathbf{y}_{ij}^{obs})^c])\}}{\kappa(\boldsymbol{\theta})} \right] \\ &= C - \sum_i \sum_{j \neq i} \sum_{z=0,1} \mathbf{g}([z, (\mathbf{y}_{ij}^{obs})^c]) P(I=i, J=j) P(Z=z | \mathbf{Y}_{ij}^c = (\mathbf{y}_{ij}^{obs})^c, \boldsymbol{\theta}), \end{aligned}$$

where $C = \binom{n}{2} \mathbf{g}(\mathbf{y}^{obs})$. This results in:

$$\nabla p\ell = \binom{n}{2} \left[\mathbf{g}(\mathbf{y}^{obs}) - E_{\boldsymbol{\theta}} \mathbf{g}([Z, (\mathbf{y}_{IJ}^{obs})^c]) \right]$$

with (I, J) jointly uniform and $(Z|I, J)$ Bernoulli with conditional log-odds of success equal to $\boldsymbol{\theta}^\top \boldsymbol{\delta}_g(\mathbf{y})_{IJ}$. We can then approximate the $p\ell$ gradient for one iteration of R_a Gibbs updates to randomly chosen dyads by

$$\binom{n}{2} \left[\mathbf{g}(\mathbf{y}^{obs}) - \frac{1}{R_a} \sum_{r=1}^{R_a} \mathbf{g}([Z^{(r)}, (\mathbf{y}_{I^{(r)}, J^{(r)}}^{obs})^c]) \right],$$

where $I^{(r)}, J^{(r)}$ are the r -th randomly chosen indices, indicating the r -th dyad that should be updated, and $Z^{(r)}$ denotes the random update status (0 or 1) for this dyad.

Full-scan: Alternatively, we can write the gradient as:

$$\begin{aligned}\nabla p\ell &= \binom{n}{2} \left[\mathbf{g}(\mathbf{y}^{obs}) - \frac{1}{\binom{n}{2}} \sum_i \sum_{j \neq i} \sum_{z=0,1} \mathbf{g}([z, (\mathbf{y}_{ij}^{obs})^c]) P(z | (\mathbf{y}_{ij}^{obs})^c, \boldsymbol{\theta}) \right] \\ &= \binom{n}{2} \left[\mathbf{g}(\mathbf{y}^{obs}) - \frac{1}{\binom{n}{2}} \sum_i \sum_j \sum_{z=0,1} E_{\boldsymbol{\theta}} \mathbf{g}([Z_{ij}, (\mathbf{y}_{ij}^{obs})^c]) \right]\end{aligned}$$

where Z_{ij} are Bernoulli according to $P(Z_{ij} | (\mathbf{y}_{ij}^{obs})^c, \boldsymbol{\theta})$.

For this full-scan Gibbs sampler for all $\binom{n}{2}$ dyad pairs taken R_b times, we can approximate the gradient by

$$\nabla p\ell \approx \binom{n}{2} \left[\mathbf{g}(\mathbf{y}^{obs}) - \frac{1}{\binom{n}{2}} \sum_i \sum_{j \neq i} \frac{1}{R_b} \sum_{r=1}^{R_b} \mathbf{g}([Z_{ij}^{(r)}, (\mathbf{y}_{jk}^{obs})^c]) \right].$$

While the full-scan Gibbs sampler requires $\binom{n}{2}$ times the work for one step of the sampler, it will always be at least as efficient as the random-scan sampler with equivalent work (i.e. taking $\binom{n}{2}$ random samples instead of one systematic sample).

4.4.2 Variation and efficiency

For the random-scan approximation using $R_a = a\binom{n}{2}$ randomly chosen updates in each iteration (for some multiple a),

$$\boldsymbol{\theta}_{RS} = \mathbf{g}(\mathbf{y}^{obs}) - \frac{1}{R_a} \sum_{r=1}^{R_a} \mathbf{g}[Z^{(r)}, (\mathbf{y}_{J^{(r)}K^{(r)}}^{obs})^c],$$

the variance will be

$$\begin{aligned}\frac{1}{R_a} \text{Var}(\boldsymbol{\theta}_{RS}) &= \frac{1}{R_a} \text{Var} \left[\mathbf{g}[Z^{(r)}, (\mathbf{y}_{IJ}^{obs})^c] \right] \\ &= \frac{1}{R_a} \text{Var} \left[E \left[\mathbf{g}[Z, (\mathbf{y}_{IJ}^{obs})^c | I, J, (\mathbf{y}_{IJ}^{obs})^c, \boldsymbol{\theta}] \right] \right] \\ &\quad + \frac{1}{R_a} E \left[\text{Var} \left[\mathbf{g}[Z^{(r)}, (\mathbf{y}_{IJ}^{obs})^c | I, J, (\mathbf{y}_{IJ}^{obs})^c, \boldsymbol{\theta}] \right] \right].\end{aligned}$$

For the full-scan approximation using R_b updates for each of the $\binom{n}{2}$ undirected dyads in each iteration,

$$\boldsymbol{\theta}_{FS} = \mathbf{g}(\mathbf{y}^{obs}) - \frac{1}{\binom{n}{2}} \sum_i \sum_{i < j} \frac{1}{R_b} \sum_{r=1}^{R_b} \mathbf{g}[Z_{ij}^{(r)}, (\mathbf{y}_{ij}^{obs})^c],$$

the variance will be

$$\begin{aligned} \frac{1}{R_b} \text{Var}(\boldsymbol{\theta}_{FS}) &= \frac{1}{R_b} \text{Var} \left[\frac{1}{\binom{n}{2}} \sum_i \sum_{i < j} \mathbf{g}[Z_{ij}, (\mathbf{y}_{ij}^{obs})^c] \right] \\ &= \frac{1}{R_b} \text{Var} \left[E \left(\frac{1}{\binom{n}{2}} \sum_i \sum_{i < j} \mathbf{g}[Z_{ij}, (\mathbf{y}_{ij}^{obs})^c | (\mathbf{y}_{ij}^{obs})^c] \right) \right] \\ &\quad + \frac{1}{R_b} E \left[\text{Var} \left(\frac{1}{\binom{n}{2}} \sum_i \sum_{i < j} \mathbf{g}[Z_{ij}, (\mathbf{y}_{ij}^{obs})^c | (\mathbf{y}_{ij}^{obs})^c] \right) \right]. \end{aligned}$$

A measure of efficiency can be calculated to compare the variances of the two CD-1 estimators:

$$\text{eff} = \frac{\text{Var}_b}{\binom{n}{2} R_a} \frac{R_b}{\text{Var}_a}$$

Since the variance depends on the vector of statistics, g , the efficiency must be derived case-by-case. If the vector of statistics contains only the number of edges, and all edges have the same probability (as in the Erdős-Rényi-Gilbert graph), then the probability distribution of $\mathbf{g}[Z_{d=IJ}, (\mathbf{y}_{d=IJ}^{obs})^c]$ conditional on the rest of the graph is the same for all (I, J) , and

$$\begin{aligned} \text{Var}(\boldsymbol{\theta}_{RS}) &= \text{Var}(\boldsymbol{\theta}_{FS}) = \text{Var} \left[E(\mathbf{g}[Z_{IJ}, (\mathbf{y}_{IJ}^{obs})^c] | I, J, (\mathbf{y}_{IJ}^{obs})^c) \right] + E \left[\text{Var}(\mathbf{g}(Z_{IJ}) | I, J, \boldsymbol{\theta}) \right] \\ &= 0 + P_{\boldsymbol{\theta}}(Z = 1) [1 - P_{\boldsymbol{\theta}}(Z = 1)] \end{aligned}$$

for equivalent amounts of work, $R_a = \binom{n}{2} R_b$. When different edges have different probabilities, this is a much harder calculation.

Although the efficiency of a full-scan sampler compared to a random-scan sampler may be inextricably model-dependent, we can state these general comments: First, that for the simplest possible random graph model, the Erdős-Rényi-Gilbert model of inde-

pendent dyads with identical probabilities for each edge, the full-scan is no more or less efficient than the random-scan, and second, that the first term in the formula for the variance of the full-scan estimator will be no larger than the corresponding term for the random-scan estimator, due to the additional variability coming from the randomness in the random scan, and hence the full-scan will always be at least as efficient as the random-scan for dyad-independent models.

4.5 Composite likelihood and blocked contrastive divergence

When there is dependence inherent in the model, we might consider an alternate continuum with the true full likelihood on one end and the pseudolikelihood on the other. As in the discussion of the CD-1 to CD- ∞ continuum, at one of these ends is the computationally difficult function of interest, and at the other is an inaccurate approximation that is orders of magnitude simpler to find. Again we examine an intermediary on this spectrum and investigate the properties of this intermediate likelihood function.

The *composite likelihood* is a function that fulfills this “partial dependence” criteria. First proposed in Lindsay (1988) and explored further in, e.g., Varin and Vidoni (2005) and Parner (2001), a composite likelihood is any function that can be written as a product of conditional probabilities such that each conditional probability $P(Y_{A_c} = y_{A_c} | Y_{B_c} = y_{B_c})$ is defined by y_{A_c} and y_{B_c} being any mutually exclusive graph configurations with non-zero probability calculable on the observed network.

In the seminal paper by Lindsay (Lindsay, 1988), the following illustrative example about contrastive divergence is given. Suppose two observations, y_1 and y_2 , come from a distribution, f . Three possible “likelihood-type objects” include:

$$L_1 = f(y_2|y_1)f(y_1), \text{ the whole likelihood;}$$

$$L_2 = f(y_2|y_1)f(y_1|y_2), \text{ the conditional composite likelihood; and}$$

$$L_3 = f(y_1)f(y_2), \text{ the marginal composite likelihood.}$$

Supposing it is impossible or undesirable to use L_1 , Lindsay asks, “Is one of L_2 or L_3 inherently more informative?” He then demonstrates that, if y_1 and y_2 are marginally standard normal, then the purely conditional likelihood is fully informative when y_1 and y_2 are uncorrelated, and the purely marginal likelihood is fully informative when y_1 and y_2 are completely correlated (Lindsay, 1988).

We can develop a family of composite likelihoods in the ERGM framework, as well. For example, if we define the collection of all node pairs $\Omega = \{(i, j) : 1 \leq i \leq n, 1 \leq j \leq n, i \neq j\}$, and the sets $c = (i, j) \in \Omega$ and $A_c = \{c\}$ and $B_c = \Omega \setminus \{c\}$, then the composite likelihood would be the product of the conditional probabilities for the existence of an edge for each dyad (the log-odds in the graph configuration consisting of exactly two nodes), conditioned on the rest of the graph. This is, of course, the pseudo-log-likelihood:

$$C\ell(\eta) = \sum_{c \in \Omega} \log P(Y_{A_c} = y_{A_c} | Y_{B_c} = y_{B_c}) = \sum_{(i,j) \in (1:I, 1:J)} \log P(Y_{ij} = y_{ij} | \mathbf{Y}_{ij}^c = \mathbf{y}_{ij}^c).$$

If $A_c = A = \{(i, j)\}$ for all $(i, j) \in (I, J)$, and $B_c = B = \emptyset$, then the composite likelihood will equal the full likelihood:

$$C\ell(\eta) = \sum_{c \in \Omega} \log P(Y_{A_c} = y_{A_c} | Y_{B_c} = y_{B_c}) = \log P(\mathbf{Y} = \mathbf{y}^{obs} | \emptyset).$$

For general A_c and $B_c = \emptyset$, the resulting composite likelihood is called a *marginal composite likelihood*. For general A_c and $B_c = \Omega \setminus \{A_c\}$, we have the *conditional composite likelihood*. It is this class of conditional composite likelihoods, with the additional requirement that the size of each A_c , $|A_c| = m$ is fixed and is the same for all A_c , (m is therefore the number of dyad pairs involved in the configuration), that we consider further.

Conditional composite likelihood provides a framework poised between declaring all dyads independent of each other (MPLE) and requiring maximization under the full likelihood (MLE). By allowing dependence only for dyads within a block, maximizing the conditional composite likelihood is more computationally efficient than maximizing

the full likelihood, and is more statistically efficient (decreased variance) than maximizing the pseudolikelihood (Liang and Jordan, 2008; Dillon and Lebanon, 2009). As with MLE and MPLE, MCLE is asymptotically consistent for its true maximizer (Lindsay, 1988). This is a drawback if the MCL is not the function we want to maximize, but is a benefit is we have reason to believe that the MCL is a sufficient representation of the ML. Specifically, the composite likelihood estimator can be consistent even when the MLE is not (Liang, 1987). Hence, when the blocks are designed such that the blocking structure captures the dependence structure in the graph, then blocking will be accurate, and much more computationally efficient.

To estimate the maximizer of this conditional composite likelihood, Asuncion et al. (2010) suggests a variant of contrastive divergence called *blocked contrastive divergence*. The difference between contrastive divergence as introduced above and BCD is the use of blocked Gibbs sampling in the MCMC. We can now define a random-scan blocked Gibbs sampler which randomly selects (with equal probability) a subset A_c and updates this entire subset jointly, conditional on the rest of the graph. Similarly, we can define a full-scan blocked Gibbs sampler which systematically updates the blocks.

As in the case of the usual contrastive divergence, a full-scan blocked Gibbs sampler will always be at least as efficient, and often more efficient, compared to a random-scan scenario for the equivalent computational work.

4.6 Contrastive divergence using composite likelihood in ERGMs, with triads as blocks

As mentioned in Section 4.5, we are requiring all blocks to be of a fixed size, and each block depends on the remainder of the graph that is not contained in the block. We describe the specific arrangement of the blocked contrastive divergence scheme as $Bm\text{-}CDn$, where m is the size of the block and n is the number of Gibbs updates we will advance in each iteration. Since analysis of the network treats the dyad as the smallest unit for MCMC updates, we must consider that a block of size $m=1$ is a block including only one dyad, conditional on the rest of the graph; that is, for any n , $B1\text{-}CDn$ is simply

CD-n.

B3-CD n , then, is blocked contrastive divergence for blocks containing randomly or systematically chosen groups of three dyads. Most simply, but also probably least useful, would be to select these three dyads from all possible sets of three dyads, and update this group jointly, conditional on the rest of the graph. In this case the transitional joint probability,

$$P \left[(i, j) = z_1, (k, l) = z_2, (r, s) = z_3 | \mathbf{Y} = \mathbf{y}_{ij,kl,rs}^{obs\ c} \right],$$

where z_1 , z_2 , and z_3 indicate the presence ($z=1$) or absence ($z=0$) of an edge in that dyad, will be equivalent to the product of the individual probabilities,

$$P \left[(i, j) = z_1 | \mathbf{Y} = \mathbf{y}_{ij,kl,rs}^{obs\ c} \right] P \left[(k, l) = z_2 | \mathbf{Y} = \mathbf{y}_{ij,kl,rs}^{obs\ c} \right] P \left[(r, s) = z_3 | \mathbf{Y} = \mathbf{y}_{ij,kl,rs}^{obs\ c} \right].$$

This specific composite likelihood does not provide a considerable gain in information about the dependence structure of the graph.

Alternatively, we can require that the three dyads are in fact one triad, (i, j) , (j, k) , (k, i) . Updating this triad jointly, conditional on the rest of the graph, will now capture a specific theoretical source of dependence: that of members in a cyclic group of three basing their likelihood of a particular triad configuration on the status of the rest of the group. Now we have $P \left[(i, j, k) = T_k | \mathbf{Y} = \mathbf{y}_{ij,jk,ki}^{obs\ c} \right]$, where T_k is any of the eight possible undirected configurations of a triad, with probabilities equivalent to the frequency of occurrence of each configuration in the remainder of the graph. Another useful arrangement for a B3-CD n arrangement is to choose configurations of potential 3-stars. These would be node sets satisfying that each dyad includes a common node, as in (i, j) , (i, k) , (i, l) . Here, $P \left[(i : j, k, l) = S_k | \mathbf{Y} = \mathbf{y}_{ij,ik,il}^{obs\ c} \right]$, where S_k is any of the eight possible undirected configurations of a 3-star, with probabilities again proportional to the occurrence of these configurations in the remainder of the network. These k-star arrangements can capture some of the dependency inherent in an individual's choice to create or dissolve new ties in the presence of other ties involving the individual.

The triangle version of B3-CD n can be implemented as follows:

1. Choose (randomly or, preferably, systematically) triads from the set of nodes,

where each triad has equal selection probability (or occurs exactly once in the systematic scheme). Update this triad using Gibbs probabilities defined, as above, by $P\left[(i, j, k) = T_k | \mathbf{Y} = (\mathbf{y}_{ij,jk,ki}^{obs})^c\right]$, where T_k is any of the eight possible undirected configurations of a triad.

2. Complete a total of n updates, fulfilling the CDn part of B3- CDn .
3. Do this S times, always beginning again with the original network.
4. Use these S samples to approximate the $\ell(\boldsymbol{\theta})$ in the steepest gradient descent Equation (4.4) with a gradient descent step size of $\lambda_t = \lambda = 0.0001 * t/1000$ (where the $t/1000$ is the decay of this parameter at iteration t):

$$\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t = \lambda_t \left[\nabla_{\boldsymbol{\theta}} [\ell(\boldsymbol{\theta}_t) - \ell(\boldsymbol{\theta})] \right].$$

5. Repeat this process for T iterations, using the previous estimate as the starting value for the next repetition.

In Asuncion et al. (2010), comparisons are shown for B1-CD1, B2-CD1, B3-CD1, and B4-CD1 with “random blocks of variables [dyads] of the same order,” but it is unclear how these blocks are chosen or even what the block size really means. It is possible that these simulations are for any random collection of m dyads, which are then updated jointly conditional on the rest of the graph. This scenario would not provide a particularly illuminating choice of blocks, but does still illustrate the general properties of Bm - CDn estimation for arbitrarily chosen blocks. According to the simulations shown in Asuncion et al. (2010), using λ_t as above and $T = 20,000$ iterations, higher-order blocks perform consistently as well as or better than lower-order blocks for increasing numbers of iterations, and where performance is judged by the L1 error between the current parameter estimates and those deemed the “truth” from using MCMC-MLE in the `ergm` package in R.

4.7 Discussion

Although Yuille (2005) derives some self-described “elementary and preliminary” convergence conditions and conditions for unbiasedness, he suggests that many relevant results in, e.g., Kushner and Clark (1978), Kushner (1987), and Younes (1999), remain to be applied to contrastive divergence. Some of these results, specifically those relating MPLE and CD-1, are unique to MCMC using Gibbs samplers. Much of the literature on statistical properties of CD algorithms is specific to certain models (Ising, restricted Boltzmann machines, and unrestricted Boltzmann machines), and many of these model-specific results have not yet been extended.

In addition, work has been done to improve the computational efficiency of contrastive divergence methods. These developments include *persistent CD*, which initializes the chain at the current iteration with the sample at the end of the previous iteration (Younes, 1999; Tieleman and Hinton, 2009); *herding*, which uses iterated conditional modes rather than Gibbs sampling (Welling, 2009); and more complicated schemes, such as *parallel tempering* (Desjardins et al., 2010). The application of these developments to model-fitting in ERGMs has yet to be explored.

Summary and Conclusions

5.1 General summary

We began by introducing the field of social network analysis, including defining field-specific terminology, describing the many and varied branches of social network analysis, and detailing the historical developments in social network modeling that directly impacted the methodologies improved upon or developed in this thesis.

The Erdos-Renyi-Gilbert model is the simplest example of an exponential family graph model, but is restricted to allowing only one constant probability of edge-formation in common for all dyad pairs in the network — a scenario not likely to be found in naturally-occurring networks. The Erdos-Renyi-Gilbert model is described in Chapter 1 and is used for illustrative purposes in Chapter 2.

Other dyadic independence models such as the transitivity models and the p_1 model of Holland and Leinhardt allow network effects to be tested against null models of reciprocity and homophily. That is, these models could test the additional effects of mutuality, density, node productivity, and node attractiveness compared to the expected effects of these characteristics after accounting for certain covariate reciprocity and homophily between nodes.

Markovian dependence models were the first class of social network models to allow a dependence structure for the dyads. Specifically, a Markov graph requires a particular dyad (i, j) to depend only on any other edges that exist with node i or node j in the

dyad. This dependence structure has seen considerable use for lattice models in spatial statistics and statistical physics.

As we note, model degeneracy, a frequent curse of the model configurations mentioned thus far, is a sign that the proposed model configuration poorly explains the underlying social process that generated the observed network. Model improvement can be gained by adding new higher-order network terms to the model, and some of these terms are described.

We then define the class of exponential-family random graph models (ERGMs) for simple, loopless, unweighted networks, giving expressions for the probability model and the loglikelihood. Direct maximization of the loglikelihood is shown to be computationally burdensome, motivating the use of MCMC methods to approximate the MLE. An alternative method, maximizing the pseudolikelihood, is also introduced, and the unknown properties and often poor performance of the MPLE in practice are mentioned as reasons to focus on improving MLE methods rather than settling for the use of the MPLE as an all-purpose estimator.

In Chapter 2 we suggest two qualitatively distinct improvements to a standard MCMC MLE algorithm as described in Geyer and Thompson (1992). First, the standard approximation is replaced by a lognormal-based approximation. Second, we describe a “stepping” algorithm that moves step-by-step toward a maximizer of the loglikelihood.

Details on the computation of the approximate MLE and the MPLE are given. We discuss that maximum pseudolikelihood estimation simply ignores the possible dependencies among arcs and maximizes the likelihood equation that results. By treating the arcs as independent, the complicated normalizing constant cancels out, and the MPLE can be found as the MLE for a logistic regression model with each arc as an observation. Evidence of the poor performance of the MPLE is shown in the biological *E. coli* example.

In implementing approximate maximum likelihood techniques to find estimates for model parameters in exponential-family random graph models, we encounter a significant problem: the approximation to the log-likelihood ratio is less accurate far from the starting value $\boldsymbol{\eta}_0$. In particular, the naive approximation to the log-likelihood has

no maximizer unless the MCMC-sampled graph statistics contain the observed graph statistics in the interior of their convex hull. For these reasons we propose an algorithm to move the starting value, by steps, closer to $\hat{\boldsymbol{\eta}}$, until the MCMC sample generated by the newly proposed starting value does in fact cover $\mathbf{g}(\mathbf{y}^{\text{obs}})$. In addition, by combining this stepping procedure with the lognormal approximation described in Chapter 2, we are able to estimate approximate MLEs in many scenarios in which this was not previously possible. We have implemented our proposed procedure in the `ergm` package of Handcock et al. (2010).

One benefit of our method is that it could remove the current necessity that MCMC MLE algorithms choose an initial value $\boldsymbol{\eta}_0$ that is already close to the true MLE. This is often a difficult task, and currently there are no better options in the literature than starting at the MPLE, which is problematic for at least two reasons. First, multiple authors have shown that the MPLE may have poor statistical properties. Second, sometimes we wish to fit an ERGM based solely on the vector $\mathbf{g}(\mathbf{y}^{\text{obs}})$ of network statistics. After all, the ERGM depends on the network only through $\mathbf{g}(\mathbf{y})$. However, to find an MPLE it is necessary to know a network that actually possesses the given network statistics. Note that this may not always be possible, for instance when $\mathbf{g}(\mathbf{y})$ must consist of integers but we wish to estimate the model parameters using a non-integer-valued vector of “observed” statistics. Even when it is possible, finding such a network introduces a potentially unnecessary additional computational step. Furthermore, the vector $\mathbf{g}(\mathbf{y})$ does not uniquely determine a network, which means that different possible “observed” networks could produce $\mathbf{g}(\mathbf{y}^{\text{obs}})$ and there is no guarantee that all of these will yield the same MPLEs. For all of these reasons, approximate methods for finding the true maximum likelihood estimator are preferable to MPLE methods; and these approximate methods continue to improve with algorithmic refinements such as those we present here.

Chapter 3 explores the performance of several closely-related alternatives to the MLE, including the MPLE and a bias-adjusted version of the MLE. This chapter again discusses the weaknesses of maximum pseudolikelihood estimation, such as the poorly-understood properties of the MPLE, problems with the pseudolikelihood standard errors, and troubling divergence between the MPLE and the approximate MLE in a variety of cases.

Due to this evidence of a substantial bias in the MPLE as an estimator of the MLE, van Duijn et al. (2009) applied the penalty of Firth (1993) to the pseudolikelihood in order to create a version of the MPLE that corrects some of the bias, which they compared to the MPLE and the approximate MLE in a number of simulations generated from a known true parameter. Their work concluded that the bias-corrected MPLE was, in general, an improvement over the MPLE; however, the results also revealed evidence of the true bias of the approximate MLE in the canonical parameter.

To address this bias in the approximate MLE, several methods of bias-correction are described, including bias-correction through subtracting the expected bias from the maximum likelihood estimate obtained from the sample. The bootstrap and jackknife methods are also discussed. These methods are less appealing in the ERGM context due to differences in the definition of the observational unit, and for the limitation that these methods all require that the MLE is finite and is estimated prior to applying the bias-correction. In the ERGM setting, infinite MLEs are a non-trivial issue; hence, any of these methods of bias-correction may be quite limited in this setting.

Conveniently, the Firth penalty to the likelihood, applied to the pseudolikelihood function in van Duijn et al. (2009), does not require the existence of the MLE in order to find the bias-corrected estimator. For this reason, we apply the Firth penalty to the lognormal approximate log-likelihood and present the resulting estimator, noting that the estimator found by applying this Firth penalty to the likelihood function is also the Bayesian maximum posterior estimator based on assigning a Jeffreys prior to the parameter.

The performance of the bias-adjusted approximate MLE, the bias-adjusted MPLE, the usual MPLE, and the usual approximate MLE are investigated in the collaboration network of the 36 partners at *SG&R* (Lazega, 2001) for a model with edges, the geometrically-weighted edgewise shared partner statistic, individual (nodal) covariate effects for seniority and for practice, and homophily (nodes sharing the characteristic) effects for practice, gender, and office, as in van Duijn et al. (2009). The MLE for the original data in this model is treated as the true parameter. A sample of networks are then generated from this distribution, and the MPLE, bias-adjusted MPLE, approximate

MLE, and bias-adjusted MLE for the canonical parameters are found, as are perceived standard errors for each estimate. Since bias-reduction may not be beneficial in comparison to the new standard errors, we compare the confidence intervals (coverage and width) of the original parameters to the confidence intervals of their bias-corrected versions. The perceived (95%) confidence intervals, relative efficiency, and confidence interval coverage for several model parameters are shown for these four estimators, leading to a conclusion that the bias-adjustment was not an outstanding improvement in this case.

We also introduce a statistical perspective on the emerging application of contrastive divergence to the field of model-fitting in ERGMs. Here the method of contrastive divergence is compared to maximum likelihood for network data, and a new class of estimators, the maximizer of a composite likelihood, is described. Several statistical results regarding contrastive divergence in ERGMs are summarized, along with a discussion of MCMC sampling schemes under which these properties hold, and contrastive divergence for these composite likelihoods is demonstrated in ERGMs.

When we estimate the MLE using MCMC, the problem of maximum likelihood can be framed as the minimization of the difference between two Kullback-Liebler divergences. This formulation of the problem is called contrastive divergence. For both the MCMC-MLE and CD framing of the problem, it is necessary for the Markov chain to reach equilibrium in order for the resulting estimate to inherit these asymptotic properties; however, it is not yet well-understood how to assess the convergence of the Markov chain to equilibrium, and the number of steps required to obtain the MLE may be unreasonably large. We follow Asuncion et al. (2010) in considering the CD- n estimator as an intermediate estimator, between MPLE and MLE, that maximizes the estimated loglikelihood using an MCMC sample generated using exactly n steps of the sampler. The hope for using this shorter-chain CD- n estimate is that when we reduce the number of steps of the chain, we reduce the computational effort enough to balance or outweigh the price we pay in the consistency and variance properties of our new estimator.

A brief discussion of Gibbs sampling in contrastive divergence is given, and the choice of a random-scan versus a full-scan sampler is discussed, with the conclusion that the choice is completely situation-dependent. More practically, it is noted that the most

efficient algorithm will always be the one with the best mixing.

A continuum can be described by defining intermediate dependence structures in the likelihood function. When there is dependence inherent in the model, we might consider an intermediary between the true full likelihood and the pseudolikelihood. The conditional composite likelihoods can be considered as intermediaries on this spectrum, since conditional composite likelihood provides a framework poised between declaring all dyads independent of each other (MPLE) and requiring maximization under the full likelihood (MLE), where dependence may be possible between all dyads.

We investigate the properties of these composite likelihood functions, noting that if the blocking structure captures the dependence structure in the graph, then blocking will be accurate, and much more computationally efficient. To estimate the maximizer of the composite likelihood, we follow Asuncion et al. (2010) in using a variant of contrastive divergence called blocked contrastive divergence, using a full-scan blocked Gibbs sampler to systematically update the blocks.

Contrastive divergence using composite likelihood is illustrated explicitly in ERGMs for an example using triads as blocks. This allows us to capture a specific theoretical source of dependence: that of members in a cyclic group of three basing their likelihood of a tie on the status of the rest of the group. The application to ERGMs was originally suggested in Asuncion et al. (2010), but is fully described for the first time here.

5.2 Summary of major contributions

In this thesis, a novel computational algorithm for improving MCMC maximum likelihood estimation has been introduced, and this method is shown to be successful in certain situations where approximate MLE has previously been impossible. This contribution, which we have implemented in the `ergm` package in R (Handcock et al., 2010; R Development Core Team, 2010), holds great promise in improving the detectability of the MLE for current MCMC-based computational methods.

Other alternative estimators such as the pseudolikelihood maximizer, composite likelihood maximizers, and bias-adjusted MLEs and MPLEs have been described and applied

to ERGMs and, for some estimators, performance is compared. Alternative methods of approximation are also discussed (contrastive divergence and blocked contrastive divergence), and statistical properties of the resulting estimators are explored as these estimators and methods of estimation are applied to exponential family random graph models.

Some theory in related work is applied (in many cases, for the first time) to ERGMs in this dissertation, including a penalized maximum likelihood estimator, motivated by Firth (1993) and van Duijn et al. (2009), contrastive divergence as a maximum likelihood approximation/alternative (Asuncion et al., 2010), and maximum composite likelihood estimation through contrastive divergence for dependence structures defined through cliques of fixed order (Lindsay, 1988).

5.3 Future work

The work in this thesis has resulted in some novel approaches to the practice of parameter estimation in ERGMs and has generated numerous possibilities for further investigation.

From our work on the stepping algorithm, we would like to extend the steplength selection options. In high dimensions, it may be unreasonable to attempt to move in a direct path from the mean of the sampled statistics to the observed statistics. Consider even in two dimensions a long, thin sample space, such that the “deepest” dimension of the data is not in the direction of the observed statistics vector from the sample mean. A more practical move toward $\mathbf{g}(\mathbf{y}^{\text{obs}})$ may be a nonlinear route. We would like to explore some alternative stepping schemes, perhaps based on measuring the “data depth” at various points in the sample space, or making a move toward a region of the convex hull of the sample with minimum distance from $\mathbf{g}(\mathbf{y}^{\text{obs}})$.

After finding such surprising results for the penalized MLE of Chapter 3, we are especially interested in extending the study of this and other estimators described in this thesis in a wider set of networks of varying size, generative process, and level of dependence present. Additional improvements in estimation may be found through the use of bridge sampling, which we have recently begun exploring. Besides using bridge

sampling for the estimation itself, we can also use bridge sampling to reweight the MCMC sample when calculating the standard errors for the penalized estimate. This would allow us to find the standard error for the Firth estimator relative to the adjusted log-likelihood (the function under which we find the Firth estimator), rather than relative to the unadjusted function (the log-likelihood maximized to find the MLE).

The less accurate coverage for all three estimators in the mean value parameterization may suggest that the t-distribution does not well match the true distribution of the mean value parameters. It is recommended in Clark and Thayer (2004) that confidence intervals calculated for the mean value parameter for exponential families in general be calculated using a functional form such as the gamma distribution or the lognormal distribution with the first two moments matched with the estimated mean value parameter (the mean of the observed statistics in the network from which the canonical parameter was estimated) and the estimated covariance of this parameter (the inverse Fisher information from the networks sampled from this estimated canonical parameter). We would like to continue our exploration of these estimators by generating alternative confidence intervals for the mean value parameters using the gamma and the lognormal distributions for the expected sampling distribution.

We also plan to implement blocking structures for composite likelihood CD into the `ergm` package and extend our current application of the Firth penalty to the missing data and directed graph cases in `ergm`.

The application of developments in contrastive divergence such as persistent CD, which initializes the chain at the current iteration with the sample at the end of the previous iteration (Younes, 1999; Tieleman and Hinton, 2009), *herding*, which uses iterated conditional modes rather than Gibbs sampling (Welling, 2009), and more complicated schemes, such as *parallel tempering* (Desjardins et al., 2010), to model-fitting in ERGMs has yet to be explored. There appears to be a wealth of information relevant to improving MCMC estimation methods to be gained from the machine learning community. We hope to contribute further to illuminating the statistical theory behind some of these methods.

Bibliography

- Agresti, A. (2002). *Categorical Data Analysis, 2nd Edition*. Wiley and Sons, Inc.
- Alba, R. (1973). A graph-theoretic definition of a sociometric clique. *Journal of Mathematical Sociology*, 3:113–126.
- Alon, U. (2007). Network motifs: theory and experimental approaches. *Nature Review Genetics*, 8:450–461.
- Amari (1985). *Differential-Geometrical Methods in Statistics*. New York: Springer.
- Anderson, C. J., Wasserman, S., and Crouch, B. (1999). A p^* primer: logit models for social networks. *Social Networks*, 21:37–66.
- Anscombe, F. (1956). On estimating binomial response relations. *Biometrika*, 43:461–464.
- Asuncion, A. U., Liu, Q., Ihler, A. T., and Smyth, P. (2010). Learning with blocks: Composite likelihood and contrastive divergence. *submitted*.
- Barndorff-Nielsen, O. E. (1978). *Information and Exponential Families in Statistical Theory*. New York: Wiley series in probability and mathematical statistics.
- Barndorff-Nielsen, O. E. (1983). On a formula for the distribution of the maximum likelihood estimator. *Biometrika*, 70:343–365.
- Batagelj, V. and Mrvar, A. (2009). *Pajek – Program for Large Network Analysis*.
- Besag, J. E. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, Series B*(36):192–236.
- Besag, J. E. (1977). Efficiency of pseudo-likelihood estimation for simple gaussian fields. *Biometrika*, 64:616–618.
- Boer, P., Huisman, M., Snijders, T. A. B., and Zeggelink, E. (2003). Stocnet: an open software system for the advanced analysis of social networks.

- Borgatti, S., Everett, M., and Freeman, L. (1999). *UCINET 5.0*. Natick: Analytic Technologies.
- Brown, L. D. (1986). *Fundamentals of Statistical Exponential Families*. Hayward: Institute of Mathematical Statistics.
- Carreira-Perpinan, M. and Hinton, G. (2005). On contrastive divergence learning. *Artificial Intelligence and Statistics*, pages 33–41. Cowell, R.G. and Ghahramani, Z. (Ed.
- Clark, D. and Thayer, C. (2004). A primer on the exponential family of distributions. *A Call Paper for Program on Generalized Linear Models, Casualty Actuarial Spring Forum*, pages 117–148.
- Copas, J. (1988). Binary regression models for contaminated data. *Journal of the Royal Statistical Society, B*, 50:225–265.
- Corander, J., Dahmstrom, K., and Dahmstrom, P. (1998). Maximum likelihood estimation for markov graphs. *Research Report 1998:8*.
- Corander, J., Dahmstrom, K., and Dahmstrom, P. (2002). Maximum likelihood estimation for exponential random graph models. In *Contributions to social network analysis, information theory and other topics in statistics: A festschrift in honour of Ove Frank.*, pages 1–17. Stockholm: Department of Statistics, University of Stockholm. Hagberg, J. (Ed.).
- Cox, D. and Hinkley, D. (1974). *Theoretical Statistics*. Chapman and Hall, London.
- Cox, D. and Reid, N. (1987). Parameter orthogonality and approximate conditional inference. *Journal of the Royal Statistical Society, B*, 49:1–39.
- Cox, D. and Snell, E. (1989). *Analysis of binary data, 2nd Edition*. Chapman and Hall, London.
- Crouch, B. and Wasserman, S. (1998). Fitting p^* : Monte Carlo maximum likelihood estimation. In *Paper presented at International Conference on Social Networks*. Sitges, Spain, May 28–31.
- Davis, J. (1970). Clustering and hierarchy in interpersonal relations: testing two theoretical models on 742 sociograms. *American Sociological Review*, 35:843–852.
- Desjardins, G., Courville, A., Bengio, Y., Vincent, P., and Delalleau, O. (2010). Tempered markov chain monte carlo for training of restricted boltzmann machines. *Artificial Intelligence and Statistics*, pages 145–152.
- Dillon, J. and Lebanon, G. (2009). Statistical and computational tradeoffs in stochastic composite likelihood. *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*, pages 129–136.
- Efron, B. (1975). Defining the curvature of a statistical problem (with applications to second order efficiency). *The Annals of Statistics*, 3:1189–1242.

- Efron, B. (1979). Bootstrap methods: another look at the jackknife. *The Annals of Statistics*, 7:1–26.
- Efron, B. (1990). More efficient bootstrap computations. *Journal of the American Statistical Association*, 85:79–89.
- Efron, B. and Tibshirani, R. (1993). *An Introduction to the Bootstrap*. London: Chapman & Hall/CRC.
- Ehm, W. (1991). Statistical problems with many parameters: critical quantities for approximate normality and posterior density based inference. *Habilitationsschrift, University of Heidelberg*.
- Erdős, P. and Rényi, A. (1959). On random graphs I. *Publicationes Mathematicae (Debrecen)*, 6:290–297.
- Firth, D. (1993). Bias reduction of maximum likelihood estimates. *Biometrika*, 80:27–38.
- Frank, O. (1991). Statistical analysis of change in networks. *Statistica Neerlandica*, 45:283–293.
- Frank, O. and Strauss, D. (1986). Markov graphs. *Journal of the American Statistical Association*, 81(395):832–842.
- Freeman, L. (1977). A set of measures of centrality based on betweenness. *Sociometry*, 40:35–41.
- Freeman, L. (1979). Centrality in social networks: Conceptual clarification. *Social Networks*, 1:215–239.
- Fukuda, K. (2004). Is there an efficient way of determining whether a given point q is in the convex hull of a given set s of points in R^d ?
- Gest, S., Moody, J., and Rulison, K. (2007). Density or distinction? the roles of data structure and group detection methods in describing adolescent peer groups. *Journal of Social Structure*, 8(1).
- Geyer, C. (1994). On the convergence of Monte Carlo maximum likelihood calculations. *Journal of the Royal Statistical Society, Series B*, 56(1):261–274.
- Geyer, C. (2009). Likelihood inference in exponential families and directions of recession. *Electronic Journal of Statistics*, 3:259–289.
- Geyer, C. J. and Thompson, E. A. (1992). Constrained Monte Carlo maximum likelihood for dependent data (with discussion). *Journal of the Royal Statistical Society, Series B*, 54(3):657–699.
- Gilbert, E. N. (1959). Random graphs. *The Annals of Mathematical Statistics*, 30:1141–1144.

- Gile, K. and Handcock, M. S. (2006). Model-based assessment of the impact of missing data on inference for networks. *University of Washington, Center for Statistics and Social Sciences Working Paper*, Number 66.
- Goldenberg, A., Zheng, A., Fienberg, S., and Airoldi, E. (2009). A survey of statistical network models. *Foundations and Trends in Machine Learning*, 2(2):129–233.
- Gong, G. and Samiengo, F. J. (1981). Pseudo maximum likelihood estimation: theory and applications. *The Annals of Statistics*, 9:861–869.
- Goodreau, S. (2007). Advances in ergm (p*) applied to a large social network. *Social Networks*, 29.
- Goodreau, S., Handcock, M. S., Hunter, D. R., Butts, C. T., and Morris, M. (2008). A statnet tutorial. *Journal of Statistical Software*, 24:1–26.
- Haldane, J. (1955). The estimation and significance of the logarithm of a ratio of frequencies. *Annals of Human Genetics*, 20:309–311.
- Handcock, M. (2003). Assessing degeneracy in statistical models of social networks.
- Handcock, M. and Gile, K. (2010). Modeling social networks from sampled data. *The Annals of Applied Statistics*, 4(1):5–25.
- Handcock, M. S., Hunter, D. R., Butts, C. T., Goodreau, S. M., and Morris, M. (2003). Statnet: An R package for the statistical modeling of social networks. Funding support from NIH grants R01DA012831 and R01HD041877.
- Handcock, M. S., Hunter, D. R., Butts, C. T., Goodreau, S. M., and Morris, M. (2007). Statnet: Software tools for the representation, visualization, analysis and simulation of social network data. *Journal of Statistical Software*, 24.
- Handcock, M. S., Hunter, D. R., Butts, C. T., Goodreau, S. M., Morris, M., and Krivitsky, P. (2010). *ERGM: A Package to Fit, Simulate and Diagnose Exponential-Family Models for Networks*. Seattle, WA. Version 2.2-7.
- Heinze, G. and Schemper, M. (2002). A solution to the problem of separation in logistic regression. *Statistics in Medicine*, 21:2409–2419.
- Hinton, G. (2000). Training products of experts by minimizing contrastive divergence. *Neural Computing*, 14.
- Holland, P. and Leinhardt, S. (1976). *Local structure in social networks*, volume 7, pages 1–45. San Francisco: Josey-Bass. David Heise (Ed.).
- Holland, P. and Leinhardt, S. (1977a). A dynamic model for social networks. *Journal of Mathematical Sociology*, 5(1):5–20.
- Holland, P. and Leinhardt, S. (1977b). Social structure as a network process. *Zeitschrift für Soziologie*, 6:386–402.

- Holland, P. and Leinhardt, S. (1978). An omnibus test for social structure using triads. *Sociological Methods and Research*, 7:227–256.
- Holland, P. and Leinhardt, S. (1981). An exponential family of probability distributions for directed graphs (with discussion). *Journal of the American Statistical Association*, 76(373):33–65.
- Hunter, D. (2007). Curved exponential family models for social networks. *Social Networks*, 29:216–230.
- Hunter, D., Goodreau, S., and Handcock, M. (2008). Goodness of fit of social network models. *Journal of the American Statistical Association*, 103(481):248–258.
- Hunter, D. R. and Handcock, M. S. (2006). Inference in curved exponential family models for networks. *Journal of Computational and Graphical Statistics*, 15(3):565–583.
- Hunter, D. R., Handcock, M. S., Butts, C. T., Goodreau, S. M., and Morris, M. (2007). Ergm: A package to fit, simulate and diagnose exponential-family models for networks. *Journal of Statistical Software*, 24:1–29.
- Hyvarinen, A. (2006). Consistency of pseudolikelihood estimation of fully visible boltzmann machines. *Neural Computing*, 18(10):2283–2292.
- Hyvarinen, A. (2007). Connections between score matching, contrastive divergence, and pseudolikelihood for continuous-valued variables. *IEEE Transactions on Neural Networks*, 18(5):1529–1531.
- Kapferer, B. (1972). *Strategy and Transaction in an African Factory*. University of Manchester Press.
- Katz, L. (1952). The distribution of the number of isolates in a social group. *The Annals of Mathematical Statistics*, 23(2):271–276.
- Koskinen, J. (2004). Essays on bayesian inference for social network. PhD Dissertation.
- Koskinen, J. H. and Snijders, T. A. B. (2007). Bayesian inference for dynamic social network data. *Journal of Statistical Planning and Inference*, 137:3930–3938.
- Kushner, H. J. (1987). Asymptotic global behavior for stochastic approximation and diffusions with slowly decreasing noise effects: Global minimization via monte carlo. *SIAM Journal of Applied Mathematics*, 47:169–185.
- Kushner, H. J. and Clark, D. S. (1978). *Stochastic Approximation for Constrained and Unconstrained Systems*. Springer-Verlag.
- Lazega, E. (2001). *The Collegial Phenomenon: The Social Mechanisms of Cooperation Among Peers in a Corporate Law Partnership*. Oxford: Oxford University Press.
- Leenders, R. (1995). Models of network dynamics: A markovian framework. *Journal of Mathematical Sociology*, 20:1–21.

- Lehman, E. L. and Casella, G. (1998). *Theory of Point Estimation*. Springer Verlag.
- Li, B. (1998). An optimal estimating equation based on the first three cumulants. *Biometrika*, 85:103–114.
- Liang, K. Y. (1987). Extended mantel-haenszel estimating procedure for multivariate logistic regression models. *Biometrika*, 43:289–299.
- Liang, P. and Jordan, M. I. (2008). An asymptotic analysis of generative, discriminative, and pseudolikelihood estimators. *International Conference on Machine Learning*, pages 584–591.
- Lindsay, B. G. (1988). Composite likelihood methods. *Contemporary Mathematics*, 80(1):221–239.
- Liu, R., Serfling, R., and Souvaine, D. (2006). *Data Depth: Robust Multivariate Analysis, Computational Geometry, and Applications*. American Mathematical Society, Providence. Held at Rutgers University, May 14–16, 2003.
- Lubbers, M. and Snijders, T. (2007). A comparison of various approaches to the exponential random graph model: A reanalysis of 104 student networks in school classes. *Social Networks*, 29:489–507.
- Makhorin, A. (2006). *GNU Linear Programming Kit, Version 4.9*. GNU Software Foundation.
- Morris, M., Handcock, M., and Hunter, D. (2007). Ergm model specifications. *Journal of Statistical Software*, 24:1–24.
- Moxley, R. and Moxley, N. (1974). Determining point-centrality in uncontrived social networks. *Sociometry*, 37:122–130.
- Nosanchuk, T. (1963). A comparison of several sociometric partitioning techniques. *Sociometry*, 26:112–124.
- Orr, G. B. and Leen, T. K. (1993). Weight space probability densities on stochastic learning: Ii transients and basin hopping times. In *Advances in Neural Information Processing Systems, Vol. 5*. Morgan Kaufmann. Giles, Hanson and Cowan (Eds).
- Parner, E. (2001). A composite likelihood approach to multivariate survival data. *Scandinavian Journal of Statistics*, 28:295–302.
- Pattison, P. and Robins, G. (2002). Neighborhood-based models for social networks. *Sociological Methodology*, 32:301–337.
- Quenouille, M. (1949). Approximate tests of correlation in time series. *Journal of the Royal Statistical Society, Series B*, 11:18–44.
- R Development Core Team (2010). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

- Rapoport, A. (1949). Outline of a probabilistic approach to animal sociology: I. *Bulletin of Mathematical Biophysics*, 11:183–196.
- Rapoport, A. (1950). Contribution to the probabilistic theory of neural nets: I. randomization of refractory periods and of stimulus intervals. *Bulletin of Mathematical Biophysics*, 12:109–121.
- Rapoport, A. (1953). Spread of information through a population with a socio-structural bias: /ii. various models with partial transitivity. *Bulletin of Mathematical Biology*, 15:535–546.
- Rapoport, A. (1957). Contribution to the theory of random and biased nets. *Bulletin of Mathematical Biology*, 19:257–277.
- Resnick, M., Bearman, P., and Blum R.W., e. a. (1997). Protecting adolescents from harm. findings from the national longitudinal study on adolescent health. *Journal of the American Medical Association*, 278:823–832.
- Rinaldo, A., Fienberg, S. E., and Zhou, Y. (2009). On the geometry of discrete exponential families with application to exponential random graph models. *Electronic Journal of Statistics*, 3:446–484.
- Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 22:400–407.
- Robins, G. and Pattison, P. (2005). Interdependencies and social processes: Dependence graphs and generalized dependence structures. In *Models and Methods in Social Network Analysis*, pages 192–215. Cambridge University Press. Carington, P., Scott, J., Wasserman, S. (Eds.).
- Robins, G., Pattison, P., Kalish, Y., and Lusher, D. (2007a). An introduction to exponential random graph (p^*) models for social networks. *Social Networks*, 29(2):173–191.
- Robins, G., Snijders, T., Wang, P., Handcock, M., and Pattison, P. (2007b). Recent developments in exponential random graph (p^*) models for social networks. *Social Networks*, 29(2):192–215.
- Roistacher, R. (1974). A review of mathematical methods in sociometry. *Sociological Methods and Research*, 3:123–171.
- Salgado, H., Santos-Zavaleta, A., Gama-Castro, S., Millán-Zárate, D., Díaz-Peredo, E., Sánchez-Solano, F., Pérez-Rueda, E., Bonavides-Martínez, C., and Collado-Vides, J. (2001). Regulondb (version 3.2): Transcriptional regulation and operon organization in escherichia coli k-12. *Nucleic Acids Research*, 29(1):72–74.
- Sampson, F. S. (1968). A novitiate in a period of change: An experimental and case study of social relationships. PhD Dissertation, Cornell University.
- Saul, Z. M. and Filkov, V. (2007). Exploring biological network structure using exponential random graph models. *Bioinformatics*, 23(19):2604.

- Schweinberger, M. (2005). Statistical modeling of network dynamics given panel data: Goodness-of-fit tests. Submitted for publication.
- Schweinberger, M. (2010). Instability sensitivity, and degeneracy of discrete exponential families. *The Pennsylvania State University, Statistics Department Working Paper*, Number 10-07.
- Schweinberger, M. and Snijders, T. A. B. (2007). Random effects models for digraph panel data. Working paper.
- Shen-Orr, S., Milo, R., Mangan, S., and Alon, U. (2002). Network motifs in the transcriptional regulation network of *Escherichia coli*. *Nature Genetics*, 31(1):64–68.
- Snijders, T. (2001). The statistical evaluation of social network dynamics. *Sociological Methodology*, 31(1):361–395.
- Snijders, T. A. B. (2002). Markov chain Monte Carlo estimation of exponential random graph models. *Journal of Social Structure*, 3(2):1–40.
- Snijders, T. A. B. (2007). Introduction to dynamic social network analysis. Presentation given at SIENA workshop, Groningen.
- Snijders, T. A. B. and Baerveldt, C. (2003). A multilevel network study of the effects of delinquent behavior on friendship evolution. *Journal of Mathematical Sociology*, 27:123–151.
- Snijders, T. A. B., Koskinen, J. H., and Schweinberger, M. (2007a). Maximum likelihood estimation for social network dynamics. Submitted.
- Snijders, T. A. B., Pattison, P. E., Robins, G. L., and Handcock, M. S. (2006). New specifications for exponential random graph models. *Sociological Methodology*, 36(1):99–153.
- Snijders, T. A. B., Steglich, C., Ripley, R., and Boitmanis, K. (2007b). Siena 3.
- Snijders, T. A. B., Steglich, C. E. G., and Schweinberger, M. (2007c). Modeling the co-evolution of networks and behavior. In *Longitudinal Models in the Behavioral and Related Sciences*, pages 41–71. Mahwah, NJ: Lawrence Erlbaum. van Montfort, K. and Oud, H. and Satorra, A. (Eds.).
- Snijders, T. A. B., van de Bunt, G. G., and Steglich, C. E. G. (2009). Introduction to actor-based models for network dynamics. *Social Networks*. in press.
- Snijders, T. A. B. and van Duijn, M. A. J. (1997). *Simulation for Statistical Inference in Dynamic Network Models*, pages 493–512. Berlin: Springer. R. Conte, R. Hegselmann and P. Terna (Eds.).
- Strauss, D. and Ikeda, M. (1990). Pseudolikelihood estimation for social networks. *Journal of the American Statistical Association*, 85(409):204–212.

- Theussl, S. and Hornik, K. (2010). *R/GNU Linear Programing Kit Interface, Version 0.3-3*.
- Thompson, S. (2006). Adaptive web sampling. *Biometrics*, 62(4):1224–1234.
- Tieleman, T. and Hinton, G. (2009). Using fast weights to improve persistent contrastive divergence. In *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1033–1040. ACM Press, New York, NY.
- Tukey, J. (1958). Bias and confidence in not quite large samples (abstract). *The Annals of Mathematical Statistics*, 29:614.
- Udry, J. and Bearman, P. (1998). New methods for new research on adolescent sexual behavior. In *New Perspectives on Adolescent Risk Behavior*, pages 241–269. New York: Cambridge University Press. R. Jessor, (Ed.).
- van de Bunt, G. G., van Duijn, M. A. J., and Snijders, T. A. B. (1999). Friendship networks through time: An actor-oriented dynamic statistical network model. *Computational & Mathematical Organization Theory*, 5(2):167–192.
- van Duijn, M. A. J., Gile, K., and 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(1):52–62.
- van Duijn, M. A. J., Snijders, T. A. B., and Zijlstra, B. J. H. (2004). p2: a random effects model with covariates for directed graphs. *Statistica Neerlandica*, 58(2):234–254.
- Varin, C. and Vidoni, P. (2005). A note on composite likelihood inference and model selection. *Biometrika*, 92(3):519–528.
- Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. Cambridge University Press.
- Wasserman, S. and Pattison, P. (1996). Logit models and logistic regressions for social networks: I. an introduction to markov graphs and p*. *Psychometrika*, 61(3):401–425.
- Wasserman, S. and Robins, G. L. (2005). An introduction to random graphs, dependence graphs, and p*. In *Models and Methods in Social Network Analysis*, pages 148–161. New York: Cambridge University Press. Carington, P., Scott, J., Wasserman, S. (Eds.).
- Wasserman, S. S. (1980). Analyzing social networks as stochastic processes. *Journal of the American Statistical Association*, 75(370):280–294.
- Wasserman, S. S. and Iacobucci, D. (1988). Sequential social network data. *Psychometrika*, 53:261–282.
- Welling, M. (2009). Herding dynamic weights for partially observed random field models. In *UAI '09: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 599–606. Arlington, VA: AUAI Press.

- Younes, L. (1999). On the convergence of markovian stochastic algorithms with rapidly decreasing ergodicity rates. *Stochastics and Stochastic Reports*, 65(3):177–228.
- Yuille, A. (2005). The convergence of contrastive divergences. In *Advances in Neural Information Processing Systems (Proceedings of NIPS 2004)*, volume 17, pages 1593–1600. MIT Press. L.K. Saul, Y. Weiss and L. Bottou (Eds.).
- Zijlstra, B. J. H., van Duijn, M. A. J., and Snijders, T. A. B. (2006). The multilevel p2 model: A random effects model for the analysis of multiple social networks. *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*, 2(1):42–47.

Vita

Education

Ruth M. Hummel

Ph.D. Statistics	2011	Pennsylvania State University
B.S. Mathematics Education	2002	Taylor University, Upland, IN

Teaching Experience

Lecturer: Graduate Applied Statistics (STA 6166 / 6167) (Fall 2010)

Univ of Florida, Institute of Food and Ag Sciences, Statistics, Gainesville, FL

Instructor: Applied Intermediate Statistics (STAT 460) (Spring 2010)

Instructor: Statistical Consulting (STAT 470) (Spring 2008)

Instructor: Introductory Statistics (STAT 200) (Summer 2005)

Teaching Assistant: Introductory Statistics (STAT 200) (2004 – 2005)

Pennsylvania State University, Department of Statistics, University Park, PA

Research and Consulting Experience

Research Assistant, funded by Office of Naval Research (2008 – 2010)

Senior Staff Consultant, Statistical Consulting Center (2006 – 2007)

Student Consultant, Statistical Consulting Center (2005 – 2006)

Pennsylvania State University, Department of Statistics, University Park, PA

Research Intern, Research and Development (2005 – 2006)

Minitab, Inc., State College, PA

Papers

Hummel, R.M., Hancock, M.S., Hunter, D.R. “*Computational algorithms for fitting ERGMs*,” submitted.

Hummel, R.M., Banga, S., Hettmansperger, T.P. “*Better Confidence Intervals for the One-Sample Variance*,” Minitab Technical Report: October 2007.

Presentations

“**A Steplength Algorithm for Fitting ERGMS,**”

- The Joint Statistical Meetings (August 2009)

“**Computational Algorithms for fitting ERGMS,**”

- 5th Annual UK Social Network Analysis Conference (July 2009)

- MURI grant meeting: University of California, Irvine (April 2009)

- Working Group meeting: University of Washington (December 2007)

“**Better Confidence Intervals for the One-Sample Variance,**”

- Millersville University (February 2007)

- Hawaii Intl Conference of Statistics and Mathematics (January 2007)

Honors and Awards

ASA Computing/Graphics Section: Student Paper Competition Winner (2009)

Penn State University: Graduate Fellowship (2007-2008)

PSU Methodology Center: Grant for Social Networks Research (June-Aug 2007)

Millersville University: Invited talk (2007)