ABSTRACT

EXPONENTIAL RANDOM GRAPH MODELS FOR MULTILEVEL NETWORKS

Dan Tattersall, M.S.
Division of Statistics
Northern Illinois University, 2014
Alan Polansky, Thesis Director

This master's thesis investigates the use of exponential random graph models for multilevel networks. It begins by describing some basic ideas in network analysis and then moves into the use of models to describe observed networks. After establishing modeling concepts for single-level networks, the discussion expands to modeling multilevel networks, which is a less common practice, and provides a brief multilevel modeling application.

Focus is given to ERGM theory basics and highlights potential problems that researchers may encounter when employing these methods. Ultimately, the reader leaves with a sense of how and why network complexity can be modeled and some of the challenges that face network research.

NORTHERN ILLINOIS UNIVERSITY

DEKALB, ILLINOIS

DECEMBER 2014

EXPONENTIAL RANDOM GRAPH MODELS FOR MULTILEVEL NETWORKS

BY

DANIEL TATTERSALL

A THESIS SUBMITTED TO THE GRADUATE SCHOOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE

MASTER OF SCIENCE

DIVISION OF STATISTICS

Thesis Director:

Alan Polansky

ACKNOWLEDGEMENTS

None of this would have been possible were it not for my wonderfully dense network of family, friends, and school colleagues who provided support, guidance, and have always showed me the importance of strong networks.

Thank you.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

CHAPTER 1

BACKGROUND

1.1 Brief History of Networks

Before networks can be modeled, it is important to consider how they are conceptualized and described. Network analysis is a method of analysis that is interdisciplinary with applications to social sciences, biology, computer science, statistics, organizational management, education, and more (Kolaczyk, 2010, pp. 3-10; Wasserman & Faust, 1994, p. 6). Some early advances in describing networks were done by sociologist Jacob Moreno in the 1930s using what he described as *sociograms*, or graphs of networks, and *sociomatrices*, or matrices describing networks. He and his colleagues, often working through their academic journal *Sociometry*, worked to standardize these descriptive tools. Following these early efforts, the next two decades were filled with an explosion of efforts and concepts to study networks including the introduction of graph theory, and eventually, as Wasserman and Faust note, "the line between sociometric and graph theoretic approaches to social network analysis began to become blurred" (Wasserman & Faust, 1994, p. 79).

Graph theory itself provided a natural extension to networks considering its early roots from Leonhard Euler, who, in 1735, was attempting to understand flow of traffic across

bridges that connected the different parts of Konigsberg, Prussia. His work laid the theoretical foundation for modern graph theory. Much of the notation from graph theory carried over into network analysis, and the vocabulary became freely used by non-mathematicians by calling network points, or individuals, *vertices* and the connections, or ties, between them *edges*. However, even today, 80 years after Moreno began studying networks, much of the vocabulary is non-standard because of the wide range of disciplines that use network analysis. Despite this disagreement of terms, consensus over many of the important tools and concepts for describing networks has emerged.

1.2 Tools to Record Network Data

The two most common tools used to illustrate a network are a network graph, or sociogram, and an adjacency matrix, or sociomatrix. The network graph represents a network with vertices connected by edges. It should be noted that distance between the vertices and the layout of the vertices are not important. For example, in the case of a friendship network between six people, individuals (vertices) are connected by friendships (edges) as shown in Figure 1.

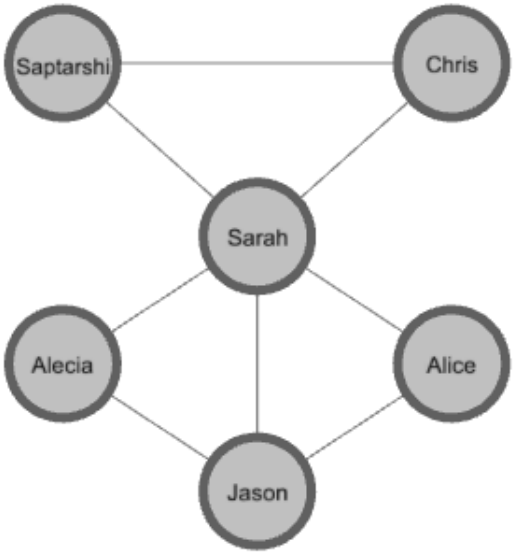

Figure 1: A small friendship network represented by a graph.

The adjacency matrix describes an identical relationship using a matrix, Y, in which each row corresponds to an individual. The columns have an identical representation to the rows. Inside of the matrix, each cell, y_{ij}, represents a possible connection, or edge, between the vertex in row i and the vertex in column j. For our purposes,

$$y_{ij} = \begin{cases} 1 \text{ when edge exists between vertex i and vertex j} \\ 0 \text{ when edge does } \textbf{not} \text{ exist between vertex i and j} \end{cases}$$

For example, the adjacency matrix of the friendship network for six people depicted in Figure 1 is given in Table 1.

Table 1: Adjacency Matrix of the Small Friendship Network Depicted in Figure 1.

	Sarah	Alice	Alecia	Jason	Saptarshi	Chris
Sarah	0	1	1	1	1	1
Alice	1	0	0	1	0	0
Alecia	1	0	0	1	0	0
Jason	1	1	1	0	0	0
Saptarshi	1	0	0	0	0	1
Chris	1	0	0	0	1	0

This specific network, shown through its adjacency matrix, has some important assumptions associated with it. These common assumptions will be used in this paper when expanding to more complicated networks:

1) The edges between vertices either occur (1 on the adjacency matrix) or they do not (0 on the adjacency matrix). This means all edges are equal, and none are weighted.

2) Multiple connections between vertices are not allowed.

3) The network is undirected, meaning that an edge between Sarah and Alice is no different than an edge between Alice and Sarah. If the network were a directed network, then there could be the case where Alice considered Sarah a friend ($y_{ij}=1$) but Sarah did not consider Alice a friend ($y_{ji}=0$), so there would be a difference between

an edge going from Alice to Sarah and an edge going from Sarah to Alice. The result of an undirected network is a symmetric adjacency matrix.

4) Vertices cannot connect to themselves, so the diagonal of an adjacency matrix is all zeroes.

In practice, an adjacency matrix can be created from a network graph, or vice versa, depending on the data.

1.3 Important Descriptive Statistics

Just as the tools to record networks have become more standardized through graphs and matrices, so have the important concepts to analyze them. Beyond the most basic measures of number of vertices, order, and number of edges, size, one of the most common concepts used to describe networks is the number of connections, or degrees, each vertex has. This simple measure can identify popular and unpopular vertices in the graph and, when observed across the entire network, can illustrate the degree distribution of a network, which has proved to be an important tool in comparing networks in order to understand if vertices have fairly uniform or widely varied degrees. For example, in the friendship network above, the degree distribution is given in Figure 2.

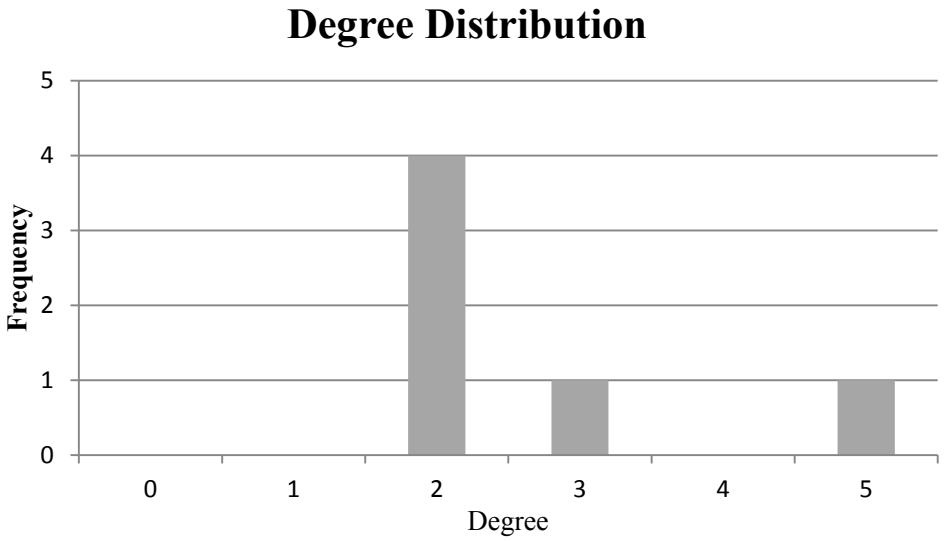

Figure 2: The degree distribution for the friendship network depicted in Figure 1.

Another common statistic to report when analyzing networks is the density of a network. The density of a network is simply a ratio of the number of possible edges ($N_v(N_v-1)/2$, where N_v is the number of vertices) to the number or observed edges, N_e, in the network. This is a simple but important descriptive network statistic because it gives a basic sense of the connectedness of the network. In terms of the simplest model described later, it gives the probability that an edge occurs in the network. For example, the friendship network in Figure 1 has 8 edges and 15 possible edges; therefore, the density is equal to 8/15 = 0.533.

In addition to the simple measure of density, the idea of transitivity is important to capture as another measure of connectedness and clustering. Transitivity is the ratio of closed triads to total (closed and open) triads as shown in Equation 1. An open triad is a subgraph

with three vertices and two edges, which leaves one possible edge missing, and a closed triad is a subgraph with three vertices and three edges as shown in Figure 3.

$$\text{transitivity} = \frac{\sum_{v \in V}(\text{closed triads in v})}{\sum_{v \in V}(\text{closed} + \text{open triads in v})} \quad (1)$$

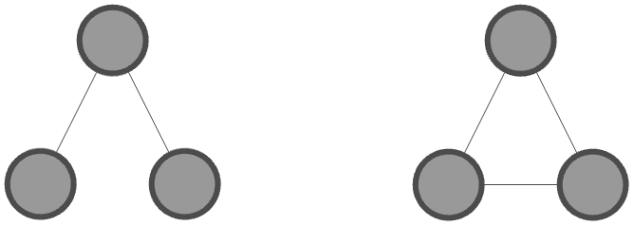

Figure 3: An open triad and a closed triad.

The structures of the open and closed triads are important ones to capture because they often show up in observed networks and have a meaningful interpretation in many situations. Simply put, a closed triad signifies that the friend of my friend is my friend. In our friendship network, there are 16 open triads. For example, an open triad occurs between Saptarshi, Sarah, and Alecia. There are 18 closed triads, one of which can be seen between Jason, Sarah, and Alice. Using Equation 1, the transitivity of this network is 0.529. These open and closed triads make up the building blocks of more complicated structures that can be counted or modeled.

These more complicated structures are often called motifs. A motif is a distinct subgraph with a set configuration and number of edges and vertices. These motifs are identified and counted in networks to identify common isomorphic, or structurally equivalent, subgraphs. For example, in an undirected network with four vertices, there are six connected and undirected motifs as shown in Figure 4.

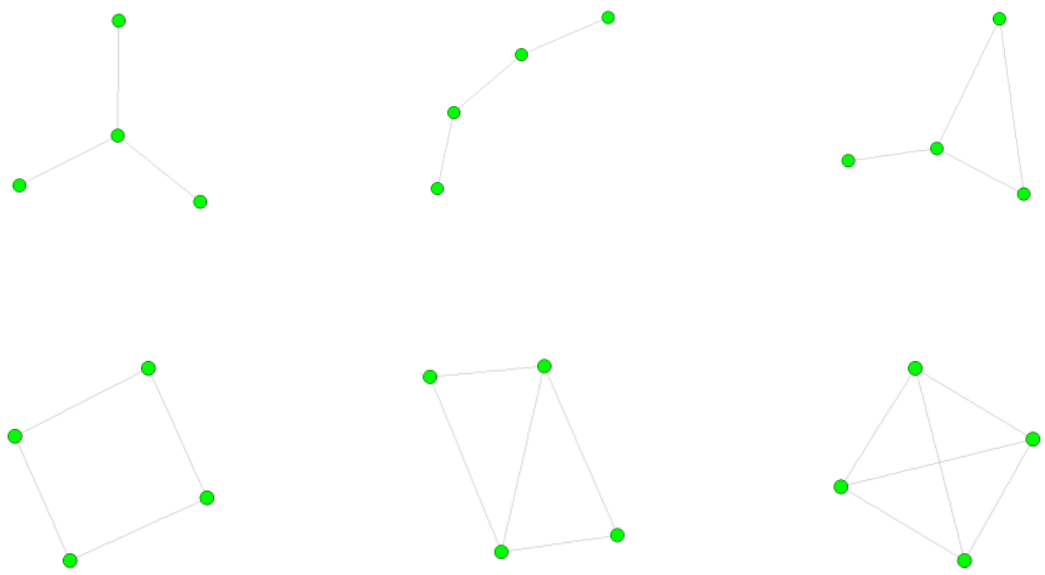

Figure 4: The six undirected motifs with four vertices.

Some of these motifs are more common than expected using some simple network models and, therefore, are important when considering modeling networks. Three particular structures are worth mentioning: stars, 2-paths, and triangles. These are used in modeling and

can be visualized explicitly through the first, fourth, and fifth motifs in the shape of a 3-star, two 2-paths, and a 2-triangle. These structures can grow beyond just 3-stars, two 2-paths, and 2-triangles as shown in Figure 5. In general, if k is an integer, we can consider k-stars, k 2-paths, and k-triangles. In the friendship network example, there is one 2-triangle involving Sarah, Jason, Alecia, and Alice.

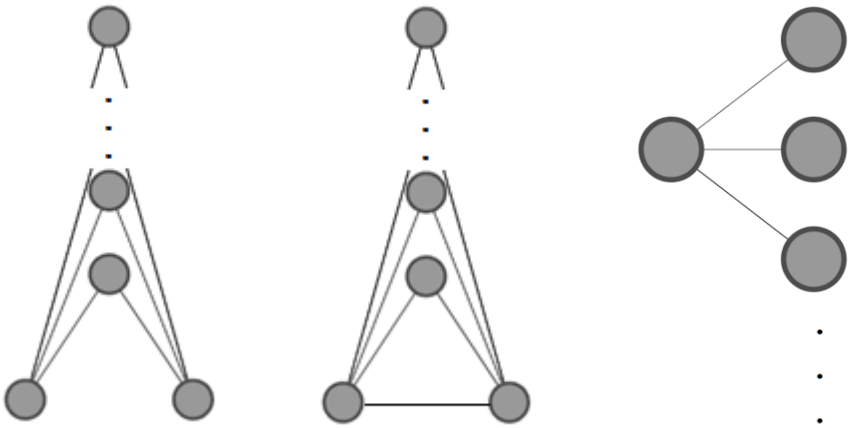

Figure 5: Structures built from k simple structures.

1.4 Transition to Models

While the previous network statistics mentioned do not constitute an exhaustive list, it highlights some of the important ones used in describing the properties of networks. Using these characterizations, observed networks can be compared with networks generated by

models for the purposes of evaluating how typical or extreme the observed network is to networks of similar order.

CHAPTER 2

MODELING NETWORKS

2.1 Simple Random Graph Models

2.1.1 Erdos-Renyi Model

One of the earliest attempts to describe networks in a statistical modeling framework was done by Erdos and Renyi in 1959 with a simple random graph framework (Erdos & Renyi, 1959). The idea was to model a network of similar order to an observed network and compare the model against the observed network. In order to do this, they considered a restricted subset, G, of all possible networks that have the same number of vertices and edges as the observed network. Then, with equal probability, networks from this subset must be chosen to compare – on some statistic, say transitivity – against the observed network. Through this comparison, one can judge whether the observed network has more transitivity than expected from similarly sized networks. It is important to note that, in practice, the observed network is not compared to the entire subset (G) of networks, or sample space, because it is much too computationally intensive to enumerate all of the networks in the subset as networks grow larger.

For example, the friendship network described earlier has six vertices and eight edges (15 possible edges), so the subset of networks with the same number of vertices and edges is $\binom{15}{8}$ = 6435 possible networks. Note that many of these networks are isomorphic if the vertices are not distinct. If some large number of those networks was chosen, say 1,000, with uniform probability from this subset and a histogram was created of their transitivity, the histogram would show variation in the transitivity of the different networks. We have done this below in Figure 6 for the friendship network. Furthermore, in comparison to the observed friendship network (shown with a dotted line), it can be seen that the observed network has a higher transitivity than only 62.1% of the networks with the same number of vertices and edges. Therefore, on this particular statistic, the Erdos-Renyi model may be a viable choice to describe this network on this statistic because it falls near the center of the transitivity distribution.

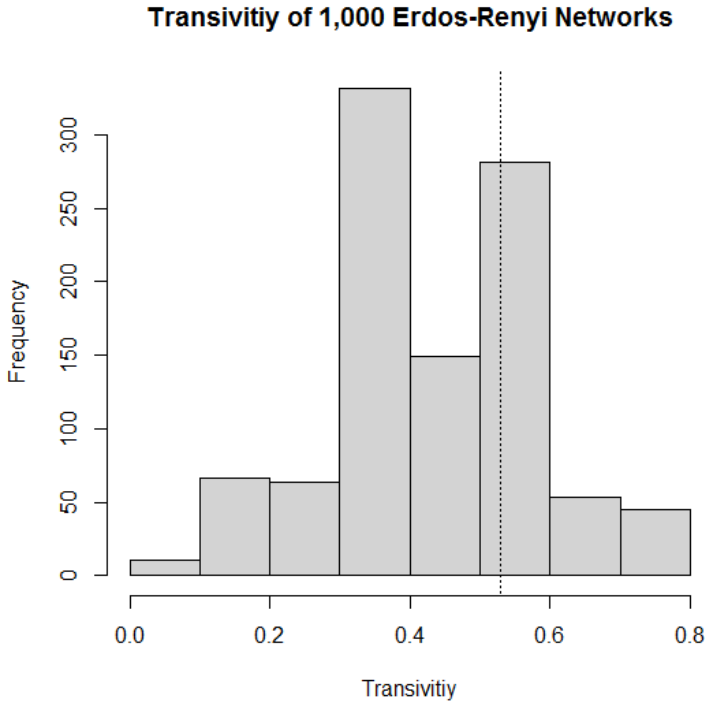

Figure 6: Transitivity histogram for 1,000 simulations from the friendship network.

2.1.2. Gilbert Model

At the same time and with a slight difference to the more well-known Erdos-Renyi model, Edgar Gilbert proposed a model which considers networks with the same number of vertices and independently assigns edges between all pairs of them with an equal probability (Gilbert, 1959). If the intent is to compare the modeled network to an observed one, then the edge probability will be approximately equal to the density of the observed network for adequate comparison. For example, consider a network that was observed with a density of

0.4. A Gilbert model would be created by flipping a biased coin (P(success)=0.4) for each pair of vertices to see whether an edge occurs between them. It's worth noting for comparison with later models that the probability of an edge occurring is not dependent on any vertex attributes or network characteristics; it is the same for every pair of vertices. As the number of vertices, N_v, increases towards infinity, the observed number of edges matches the Erdos-Renyi model and the observed network. And just like the Erdos-Renyi model, the observed network could be compared against a large collection of randomly generated networks from the Gilbert model on some network statistic in order to understand how the observed network differs from networks of the same density. An advantage to considering the Gilbert model is that thinking about edge formation as occurring with some probability provides a natural extension into exponential random graph models, which provide a much more complex way of modeling networks while considering the appearance of edges as probabilistic.

2.2 Exponential Random Graph Models for Single-Level Networks

2.2.1 Why Models Need Complexity

While the simple random graph models created by Erdos, Renyi, and Gilbert were a step forward, they usually fail to accurately model real-world networks. As Harris (2014) and others have summarized, when comparing real-world networks to random ones, there is often a skewed degree distribution, homophily (i.e. similar actors grouping together), and transitivity occurring more often than expected. Additionally, as the famous Watts and

Strogatz (1998) small-world paper explores, vertices in real-world networks are often separated by only a few degrees through short path lengths, which does not always happen with simple network models.

Additionally, simple random models only allow for comparison between the observed and randomly selected networks on a single statistic. Therefore, while the simple random model does provide some idea of where the observed network falls in comparison on a network statistic, such as transitivity, to all networks with the same number of vertices, "we only consider one effect at a time, and we ignore the nesting of configurations" which give us the nuances in observed networks described above (Robins & Lusher, 2013b, p. 31). In order to improve upon this model, it is necessary to include these complexities and nesting of configurations. Using exponential random graph models (ERGMs), network analysts can incorporate all of those network characteristics into the model, which allows the focus to shift from comparing an observed network to the distribution of simulated simple random networks on a specific statistic to trying to build a model that results in a sample space that is more consistent with the observed network. This process relegates the simple random networks, which have a uniform distribution over all networks of the same size, order, and a specific statistic, to serve as null models for the ERGM, which places varying probabilities on similar-order networks based on configurations in the network.

2.2.2 How Complexity Is Modeled

2.2.2.1 P_1 model

The framework for a general ERGM arose out of the p_1 model specification by Holland and Leinhardt (1981) for directed graphs and the development of Markov graphs by Frank and Strauss (1986).

The p_1 model from Holland and Leinhardt was important because it was the first to use the exponential family of distributions in its model and, thus, the use of a log-linear form familiar to modeling data with binary outcomes, such as the presence or absence of an edge (Harris, 2014, p. 22). The general goal of this form was to allow network analysts to incorporate more complex concepts in network models. For example, Holland and Leinhardt had an interest in accounting for vertex attributes. Kolaczyk (2010) gives the general form of this ERGM as

$$P(Y = y) = \frac{1}{\kappa(\theta)} \exp\left(\sum_{v \in V} \theta_H g(y_H)\right) \quad (2)$$

where Y is a random variable representing the entries of the adjacency matrix; y is a particular realization of the adjacency matrix as represented by Y; θ_H is the parameter for a graph

configuration represented by H, usually a subset structure such as stars or triangles; $g_H(y)$ is a function of cells in the adjacency matrix and is 1 if the configuration occurs in the network through those cells and 0 if it does not occur; κ is a normalization constant that allows the function to be a probability density function by summing to 1 over the sample space; and v is each vertex in the set of all vertices V.

While the general form may seem complicated at first, its simplicity can be viewed through a model analogous to the simple random graph models. That is, the Gilbert model can be written in terms of an ERGM of the form

$$P(Y = y) = \frac{1}{\kappa(\theta)} \exp\left(\sum_{i<j} \theta_{ij} y_{ij}\right) \quad (3)$$

where the only configuration is the presence or absence of an edge so that y_{ij} represents each possible edge in the network and θ_{ij} is the parameter for the edge term. A careful look at the previous equation will reveal that summing all of the y_{ij}'s is equivalent to summing up all of the cells in the adjacency matrix, which are either 0 or 1.

It is important to remember that in this simple random graph model, the formation of an edge is independent from all other vertices and edges around it. This is often referred to as a Bernoulli independence model because the entries of Y are essentially N_e independent

Bernoulli draws. In the friendship network, this means that an edge between Jason and Alecia is independent of an edge between Alecia and Sarah.

2.2.2.2 Markov Dependence Model

Markov Assumptions

By using the exponential family of distributions for modeling and introducing the Markov assumption, Frank and Strauss (1986) further improved the ability to easily model complicated network structures. This is achieved by adding the assumption that the probability distribution of an edge depends only on the presence or absence of edges for vertices also incident to that edge. In the previous example, that would mean that the probability of observing an edge between Jason and Alecia and an edge between Alecia and Sarah would be dependent on each other because they have a vertex, Alecia, in common. This assumption is a substantive departure from the Bernoulli independence model and is important because it has been shown in practice that edges often do not form independently of each other.

Markov Model: Overview

In order to incorporate this dependence concept into a quantitative form, Frank and Strauss (1986) employed the earlier log-linear ERGM introduced by Holland and Leinhardt

(1981). Through some investigation on the ramifications of this dependence assumption and use of the Hammersly-Clifford theroem, Frank and Strauss (1986) show that a network graph will be a Markov network graph if and only if it can be specified using the form

$$P(Y = y) = \frac{1}{\kappa(\theta)} \exp\left\{ \sum_{k=1}^{N_v - 1} [\theta_k S_k(y)] + \theta_T T(y) \right\} \quad (4)$$

where $S_k(y)$ represents the number of k-stars in the network ($S_1(y)$:edges, $S_2(y)$:2-stars, $S_3(y)$:3-stars, . . . ,$S_{Nv-1}(y)$:N_v-1 stars); θ_k is the parameter for the k^{th} star; T(y) represents the number of triangles; and θ_t is the parameter for the triangles. This model has many more terms than the ERGM form of the simple random graph model, which just considered an edge forming between two vertices, through the addition of terms for 2-stars (i.e. an open triad) through k-stars and a term for the triangle structure.

Markov Model: Nesting Structures: As shown earlier, the interpretation for the parameter, θ, associated with each term relates to the log-odds, which is akin to the interpretation of parameter coefficients in logistic regression, except that these terms are all nested inside one another so they cannot be considered separately. For example, when looking at the triangle term, one must consider the 2-star term because there are three 2-stars in a triangle and the edge term because there are three edges inside of each triangle. Therefore, in order to determine the probability of an edge occurring, it must be determined which

structures that edge creates. If the edge creates a 2-star and a triangle (e.g. the edge between Chris and Saptarshi in the friendship network) then in order to calculate the probability of that edge occurring, the 2-star and triangle terms would need to be included in the calculation of the edge probability because the probability is calculated conditionally on the rest of the network. In general, the probability of an edge can be found as

$$logit\left[\Pr\left(X_{ij}=1\middle|X_{-ij}=x_{-ij},\theta\right)\right]=log\left(\frac{\Pr\left(X_{ij}=1\middle|X_{-ij}=x_{-ij},\theta\right)}{1-\Pr\left(X_{ij}=1\middle|X_{-ij}=x_{-ij},\theta\right)}\right) \quad (5)$$

$$=log\left(\frac{\Pr\left(X_{ij}=1\middle|X_{-ij}=x_{-ij},\theta\right)}{\Pr\left(X_{ij}=0\middle|X_{-ij}=x_{-ij},\theta\right)}\right)$$

$$=\theta_1\delta_{ij,1}^{+}+\theta_2\delta_{ij,2}^{+}+\cdots+\theta_p\delta_{ij,p}^{+}$$

where there is a change statistic, δ^{+}, for each structure in the model which represents the increase in the number of that structure by adding an edge between vertex i and j. Therefore, for the previous example involving Chris and Saptarshi, the conditional log-odds is $\theta_1+2*\theta_2+\theta_T$ and the edge will occur with a probability of $\frac{\exp(\theta_1+2*\theta_2+\theta_T)}{1+\exp(\theta_1+2*\theta_2+\theta_T)}$. If this edge were to create structures other than an edge, two 2-stars, and triangle, it would be necessary to include those parameters in the calculation as well. Because the probability is calculated conditionally on the rest of the network, the calculation can become quite complicated as the network grows in order and has more complex structures.

Estimating the parameters of an ERGM based on an observed network can be computationally difficult and usually requires Markov chain Monte Carlo (MCMC) methods as will be explained in Chapter 3. This is particularly true as networks grow in complexity, which renders maximum likelihood estimation (MLE) unfeasible. Often, because of this complexity in estimating the parameters and the trouble interpreting those parameters due to nesting structures, stars over size 3 are not commonly estimated (Kolaczyk, 2010, p. 183).

Markov Model: Stars and Degree Distribution**.** One important reason to include stars of at least k=2 is that they can model the variance of the degree distribution (Koskinen & Daraganova, 2013, p. 62). The mean of the degree distribution is already modeled through the edge parameter, and the variance is shown by

$$\frac{1}{n}\sum_{i=1}^{n}(x_{i+}-\bar{x})^2 \qquad (6)$$

where $x_{i+}^2 = \sum_{i=1}^{n}\left(\sum_i \sum_j x_{ij}x_{jh}\right)$ is the sum of the 2-stars in the network and $\bar{x}$ is $\frac{N_e}{\binom{N_v}{2}}$. The variance is important to model because simple random network models do not adequately describe the variance and skew of the distribution.

Markov Model: Modifications. Whereas the problem from the simple random graph models was a lack of ability to address skewed degree distributions, homophily of vertex attributes, and transitivity, the Markov graph model addresses all of these issues by inducing possible dependence between the appearance of edges. However, Markov graph models have issues as well.

One of the biggest issues with Markov graph models is that they often have trouble correctly modeling real-world networks. Part of this problem can be attributed to phase transitions in the graph (Koskinen & Daraganova, 2013; Snijders et. al., 2006). Phase transitions occur when a graph changes from low density to high density, and, with Markov graphs, this transition can occur with only a very small change in some of the parameters. The problem arises because of the interdependence of the terms. This results in a model that struggles to describe a graph that has neither high nor low density. Because of this problem, Snijders et al. (2006) proposed adding an alternating k-star term and an alternating k-triangle term.

The alternating k-star term works by combining all of the star statistics into one term and alternates the sign on each of them while also down weighting the effect of each. That is,

$$AKS(y) = S_2(y) - \frac{S_3(y)}{\lambda} + \cdots + (-1)^{N_v-1}\frac{S_{N_v-1}(y)}{\lambda^{N_v-3}} = \sum_{k=2}^{N_v-1}(-1)^k\frac{S_k(y)}{\lambda^{k-2}} \quad (7)$$

where λ can be set or estimated or be set to a certain value as recommended by Snijders et al. (2006) in their article that introduced the alternating terms. If it is estimated, the probability function is no longer from the exponential family of distributions but becomes a curved exponential family function. Hunter and Handcock (2006) estimate λ for the same application performed by Snijders et al. (2006) and find that their model outperforms the original on the

likelihood ratio test criteria. They discuss the difficulties in estimating parameters for a curved exponential family and detail their approach which combines computationally intensive methods with biased ones in order to find an MCMC MLE (Hunter & Handcock, 2006, p. 571).

Koskinen and Daraganova (2013) give some basic guidelines for interpreting the AKS term and its parameter in the ERGM, θ_s (pp. 66-67). They explain that when θ_s is positive, the network tends to be more centralized, whereas when it is negative, the degree distribution is more even. Also, as λ increases, the centralization is focused on higher degree vertices.

In addition to the AKS term, Snijders et al. (2006) proposed a similar term but for triangles, the alternating k-triangle term which models the prevalence of the k-triangle structures shown in Chapter 1. Including this term in the model satisfies a dependence structure that is in addition to Markov dependence. Such an addition has been proposed by a number of network analysts and is also referred to as partial conditional dependence (Pattison and Robins, 2002; Snijders et al., 2006). The effect of the term is that it allows for clustering of triangle structures whereas the original Markov model does not account for this tendency. The specification of this term is similar to the AKS term in that it has a down weighting variable, λ, which decreases the effect of higher order triangles and has alternating signs on each triangle. The form of this term is given by

$$ATS(y) = 3T_1(y) - \frac{T_2(y)}{\lambda} + \cdots + (-1)^{N_v-3}\frac{T_{N_v-2}(y)}{\lambda^{N_v-3}} \quad (8)$$

$$= 3T_1(y) + \sum_{k=2}^{N_v-2}(-1)^{k+1}\frac{T_k(y)}{\lambda^{k-1}}$$

A positive parameter for the AKT term, θ_{AT}, in the ERGM model promotes clustering. A small value for λ, however, limits this clustering effect while a large value of λ encourages the clustering effect. This occurs because simple 1-triangles (3 possible edges, 3 actual edges) are more clustered than larger k-triangles, such as 4-triangles (15 possible edges, 9 actual edges), and a larger λ gives more influence to smaller k-triangles

While network analysts have suggested that other terms should be included with the original Markov network model, AKS and AKT terms have gained the most acceptance because of their ability to best improve the model with regard to representing observed networks. This method provides a better way to model degree distribution and a way to model clustering.

Markov Model: Frozen Graphs and Overspecification. Because the ERGM framework can allow for many more terms and increasingly complicated models, the issue of model selection becomes an issue. With all of these complex models, it is important to remember the original goal: find a subset, *G*, of all possible networks that describe the observed network so that comparisons between model and observed network can be made to either understand how the observed network differs from expectation or how to best model the observed network. As Kolaczyk astutely notes, “An important practical issue is determining which *G* to use”

because it forms the basis for all of the model analysis (Kolaczyk, 2009, p. 163). Problems will arise if the incorrect model is chosen and that can happen in the following situations:

1. The model is overspecified and too perfectly describes the observed network data.
2. The parameter values are too extreme.
3. Homogeneity assumptions are not made about parameters.

Model selection becomes more of an issue because, when developing more complex ERGMs through the addition of more terms, network analysts effectively shrink the subset *G*. For example, if an ERGM was created that just modeled the number of edges in a network as was done in the simple random graphs and represented in Equation 3, then it would limit the subset *G* to networks with the same number of vertices and similar densities. If a network analyst were to further refine that model to include terms for 2-stars, 3-stars, and triangles, it would further limit the subset *G* to networks with the same number of vertices, similar density, and similar number of 2-stars, 3-stars, and triangles. If more complex structures were increasingly added to the model to account for every structure in the observed network, then *G* would become so small that there would be very few networks in the subset. The result of this overspecification of the model is a lack of variation in the subset, thereby limiting the subset to only a few networks. This is problematic because the goal is to create a statistical, not purely mathematical, model that considers edge formation occurring with some probability depending on the vertices and structures involved, so there should naturally be variation in the networks created by the model. This overparameterization problem, which

includes choosing terms that represent structures that do not appear in the network, is called model degeneracy or a frozen graph problem (Pattison & Snijders, 2013, p. 289).

Model degeneracy not only occurs because the model has too many terms and describes the observed network too perfectly, but problems can also occur when the parameter values for the terms are so extreme (high or low) that they encourage or discourage a specific structure too strongly. This also causes the subset *G* to be too small and lack variability. In this scenario specifically, the network is either nearly empty or nearly completely connected.

In addition to including too many structures or the wrong structures in the model and having too extreme of parameter estimates, a homogeneity assumption is usually made about the parameters for each network structure. This assumption is made because otherwise it would be necessary to estimate a parameter for each vertex pair on a certain structure which would create an over-parameterized model. Homogeneity implies that, for example, in Equation 3, the θ_{ij} parameter would be the same for all edges, and hence only one parameter would need to be estimated.

Furthermore, network estimation suffers from a problem of parameter estimation converging using MCMC methods. When parameter estimates fail to converge, "the estimates are meaningless," and it is necessary to respecify the model (Robins & Lusher, 2013a, p. 178). A lot of literature has suggestions on how to proceed with model selection including Robins and Lusher (2013a), Robins et al. (2007), Hunter et al. (2008), Goodreau et al. (2008) and more. Much of the advice is practical and based in model-fitting experience, such as to include at least 2- and 3-star terms or include at least alternating terms.

As mentioned in Lusher, Koskinen, and Robins (2013), one common thread through most of the literature is that network researchers should incorporate existing theories about connections to avoid incorrectly choosing structural parameters. This is true especially because there is so much dependency between the terms in the model, and therefore terms should only be included when there is good rationale to do so. By failing to heed this advice, we may encounter problems which will lead to uninterpretable results at best and model degeneracy and frozen graphs at worst. The practical aspects of these problems will be dealt with through the application of the multilevel framework.

2.3 Conclusion

Whereas the first chapter focused on describing characteristics of networks through precise numerical summaries (e.g. the density is 0.2 and the transitivity is 0.3), this second chapter has explored describing networks using statistical models, which, at their heart, involve beliefs about how networks evolve and why edges exist between vertices. Having established these descriptions and models, Chapters 3 and 4 will apply them towards a more complex scenario: networks with more than one level and, therefore, vertices of different types.

CHAPTER 3

MODELING MULTILEVEL NETWORKS

3.1 Introduction to Multilevel Networks

3.1.1 Defining a Multilevel Network

A multilevel network consists of a network with at least two distinct sets, or modes, of vertices. In this way, it is similar to the more commonly analyzed affiliation network. However, *multilevel network* usually denotes a network that has edges within each mode and between modes, whereas affiliation networks only have edges within only one of the modes (usually people) and between modes (usually people and events). While the multilevel framework can consider these affiliation networks and other networks with two modes of vertices, it excels at analysis of networks with edges at all levels.

The basic concept of the multilevel network is well-described by Wang, Robins, Pattison, and Lazega (2013) where they limit their research to a two-mode network. They describe a macro level, or more generally an A network, and a micro level, or more generally a B network, as the two modes, with each one separately being similar to a single-level network. Additionally, there are connections between these two networks in the meso level, or

more generally the X network. When considered alone, the meso network belongs to a class of networks called bipartite networks.

The vertices within these networks often have some hierarchical feature such as schools and students. Often, when the network has a hierarchical structure, the higher level vertices, say schools, are denoted as the macro-level network. Alternatively, the lower level vertices, say students that go to a school in the macro level, are denoted as the micro-level network. Edges in the macro network between schools could represent a shared school district while edges in the micro level between students could represent friendships, and edges in the meso level between students and schools could represent students attending that school. The multilevel network is perhaps best explained through the visual in Figure 7, which is similar to the one provided by Wang et al. (2013).

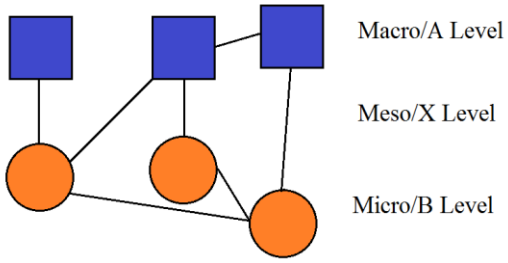

Figure 7: A multilevel network.

As Wang et al. (2013) establish in their concluding remarks, this multilevel framework can also be applied to networks that are not nested but merely consist of two distinct groups, such as males and females or the Jets and the Sharks. In this case, the A, B, and X network descriptions fit more adequately than do the macro, micro, and meso connotations.

3.1.2 Examples of Multilevel Networks

Many networks can be described as multilevel networks. For example, Wang et al. (2013) studied a network of elite cancer researchers connected by advice-giving and -seeking relationships at the micro level and research laboratories connected by collaboration relationships at the macro level. The macro research laboratory network is connected to the micro elite cancer researcher network through an employee-employer relationship which provides the edges for the meso level.

Currently, a major limitation to studying multilevel networks is that many network datasets are not created with multilevel networks analysis in mind, so edges at all levels are not reported with initial data. For example, a popular affiliation dataset looks at Southern women and the social events they attended (Davis, Gardner, & Gardner, 1941). This dataset has a micro level (Southern women), a macro level (social events), and a meso level (Southern women attending social events). However, the only edges that were recorded were friendships between women and the events they attended, so the macro-level network has no edges in it. While the multilevel ERGM framework can work here, potential insights are lost by not defining edges between the social events in the macro level. If these edges are inferred later on, such as placing an edge between two events that have common attendees, artificial connections are created between macro-level vertices and does not offer additional insight or new information.

3.2 Expansion of ERGMs to Multilevel Networks

3.2.1. A Simple Model

While a simple network for the single-level network was one in which only edge structures were modeled, the simple network for the multilevel network, which is also used as the null model, is one where the macro (A) and micro (B) networks are modeled without the edges from the meso (X) level network as part of the model. This model does not consider the network levels connected and would identify important structures in each network separately, such as Markov and social circuit structures (i.e. alternating structures).

3.2.2. Introducing Complexity

By introducing connections between levels into the network model, the connections within each network may be better understood as a piece of a larger multilevel structure. The multilevel structure could lead to additional, useful interpretations of the network. For example, say there is a vertex in network B that is highly connected to the other vertices in B. The interpretation of the structure may be aided by knowing how network B is connected to nodes in network A (perhaps that B node is popular in network A as well).

In their investigation of French cancer researchers, Wang et al. (2013) discovered that a multilevel analysis of network connections is useful in the interpretation of the entire network. When each network is studied alone, features of the bigger picture are lost. For

example, looking at the researcher advice network alone, Wang et al. (2013) saw a tendency towards clustering in researchers. However, by looking at the entire multilevel structure, they discovered that some of this clustering is attributable to researchers at the same laboratory seeking advice from each other through interaction effects and also through other clustering effects due to researchers at collaborating laboratories seeking advice from each other through cross-level effects. Additionally, the inclusion of cross-level effects in the model "greatly simplified the previous specifications" of the researcher-level model because "complicated features of the within-level network structure are explained solely by the cross-level interactions" (Wang et al., 2013, p. 111). Therefore, by introducing a more complex framework through the multilevel method of analysis, Wang et al. (2013) were actually able to obtain a simpler and more complete model to describe the observed network.

3.2.3. How Complexity Is Modeled

The multilevel network model as specified by Wang et al. (2013) takes the form

$$P(A = a, X = x, B = b) \tag{9}$$

$$= \frac{1}{\kappa(\theta)} exp\left(\sum_{H} \{\theta_H z_H(a) + \theta_H z_H(b) + \theta_H z_H(x) + \theta_H z_H(a, x) + \theta_H z_H(b, x) + \theta_H z_H(a, b, x)\}\right)$$

where (1) A, X, and B are random variables representing adjacency matrices for networks A, X, and B; (2) a, x, and b are specific realizations of those adjacency matrices; (3) H, just as in Chapter 2, is a network configuration such as an edge or a star; (4) $z_H(a)$, $z_H(b)$, and $z_H(x)$ represent the network statistics for the macro-, micro-, and meso-level networks respectively - for example, using for the simple random graph model for the macro network, $z_H(a)= \sum_{i<j} y_{ij}$, which is just the sum of all of the edges in the macro network; (5) $z_H(a,x)$ and $z_H(b,x)$ represent the network statistics for the interaction structures which have edges from either the macro or micro network and the meso-level network; and (6) $z_H(a,x,b)$ represents the network statistics for the cross-level structures which have edges from all three networks.

To connect this back to Chapter 2, this means that all of the terms for a single network, say the macro level, will fit inside of $\theta_Q z_Q(a)$ term of this model. Imagine, then each term in the multilevel network model representing an entire network of specified structures or the interactions and crossing of those structures, and it becomes easy to see how this model can quickly become complex.

The following discussion of this model will describe each term in the model. Also, as the terms and structures become more complex, the shorthand notation used by Wang et al. (2013) will be introduced and employed.

3.2.3.1 Meso-Level Term: $\theta_Q z_Q(x)$

In the multilevel framework, the structures modeled within the macro and micro levels are exactly the same as for a single network. However, the structures modeled within the meso level are slightly different because the meso level is a bipartite network. Bipartite networks are a type of network that consist of two subsets of vertices and of edges that have an endpoint in each subset, which means none of the vertices within a subset are connected. In this respect, bipartite networks clearly differ from normal networks because of the restriction imposed on which vertices can have edges between them. The result of this restriction is that certain structures, such as triangles, cannot occur, which becomes important in choosing structures to model. Additionally, whereas having more closed than open triads indicates higher clustering in a non-bipartite network, having more 4-cycles than 3-paths indicates a higher clustering bipartite network because a 4-cycle is the smallest closure possible in a bipartite network. The 3-path and 4-cycle can be seen in Figure 8.

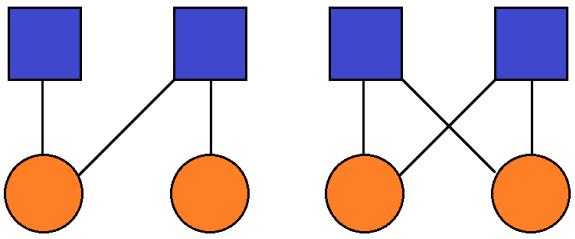

Figure 8: Bipartite 3-path and 4-cycle.

These structures are relevant to modeling a bipartite network insofar as trying to have the model have similar characteristics to the observed one as seen in Chapters 1 and 2.

In addition to including 3-paths and 4-cylces into the model, Wang (2013) recommends using star terms, including alternating stars, and alternating 2-paths (pp. 122-124).

3.2.3.2 Interaction Terms: $\theta_Q z_Q(a,x)$, $\theta_Q z_Q(b,x)$

The modeling of the interaction terms in a multilevel network begins to introduce much more complexity into the model. This complexity can come in the form of a simple structure having complex implications when prevalent in a large network. Two examples are the Star2AX and TXAX structures seen in Figure 9. The Star2AX structure creates hubs in the A network where A vertices and B vertices are connected through popular A vertices. The TXAX structure involving two A vertices and a B vertex is easy enough to conceptualize, and the interpretation of this structure could be a number of simple concepts, such as two A vertices which connect to a common B vertex sharing an edge, which is very similar to the homophily-based triangle structure from the single-level networks.

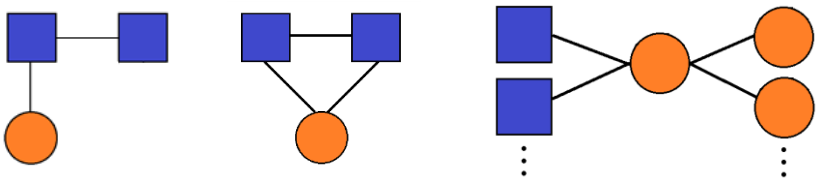

Figure 9: Star2AX, TXAX, and AAAXS.

However, the effect of the TXAX structure on a large network is difficult to imagine. Even through simulation of a network with 71 vertices and a fixed density of 0.17 in each network, the effects are a little unclear, as can be seen below in Figure 10 and Figure 11.

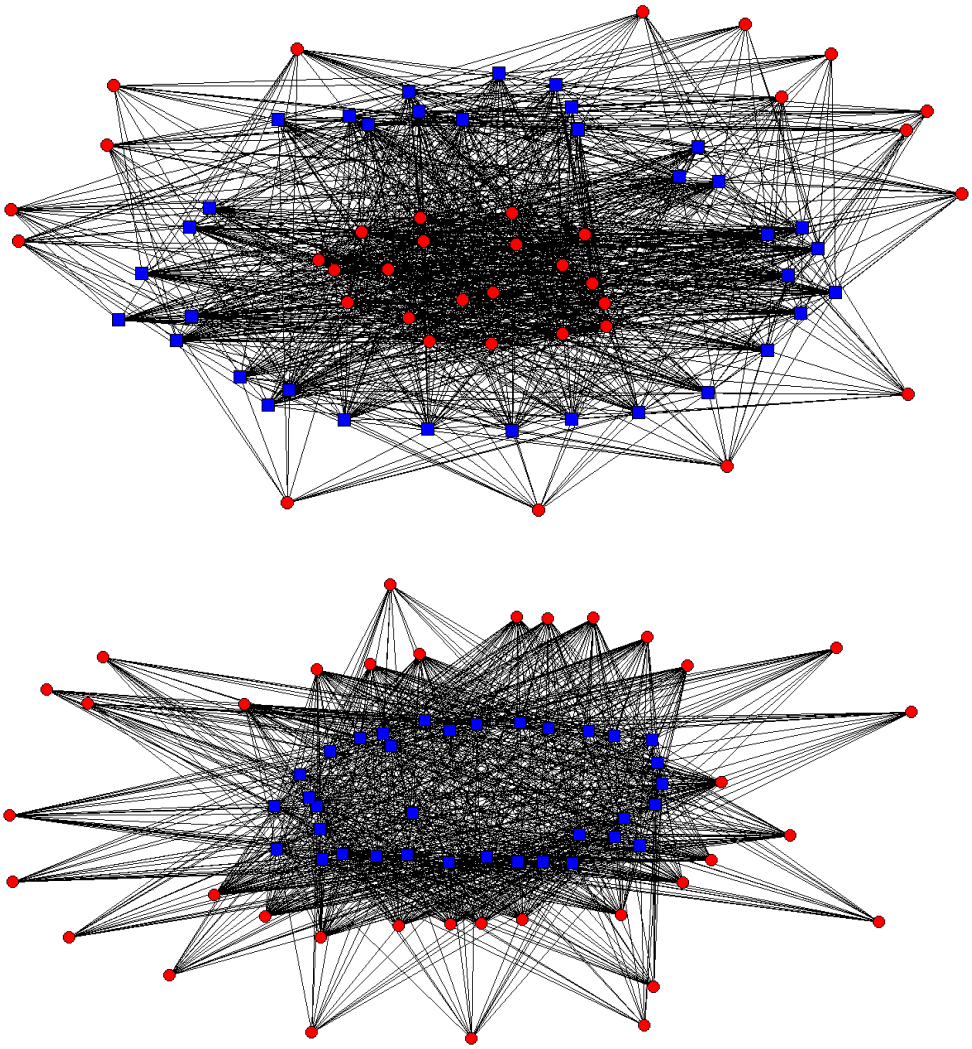

Figure 10: Simulated network with high prevalence of TXAX. Top: Whole network. Bottom: X level.

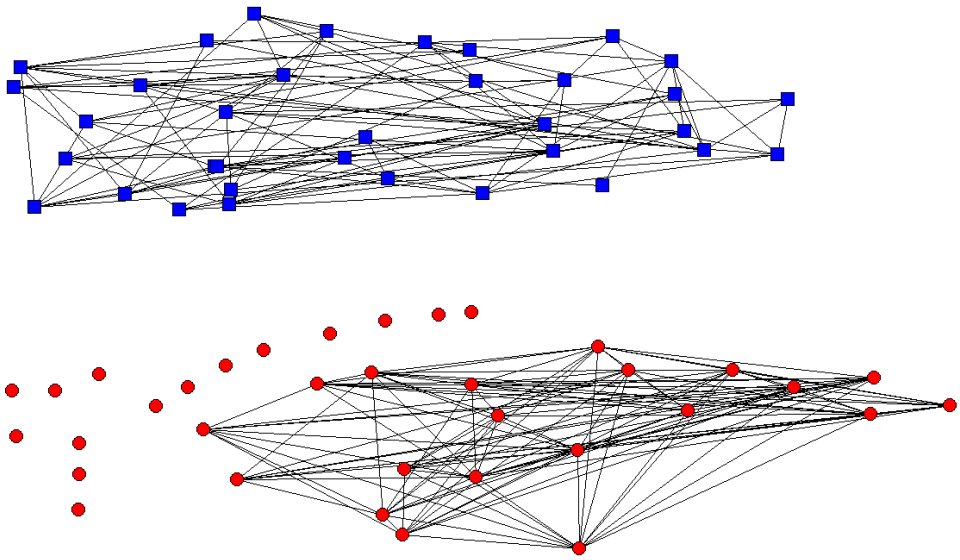

Figure 11: Simulated network with high prevalence of TXAX. Top: A level. Bottom: B level.

Additionally, this complexity can be seen purely in the structures themselves before their effect on a large network is even considered. For example, an alternating-A/alternating-X star, or AAAXS, is complex in itself, which can be seen in Figure 9, and has various possible explanations, such as one vertex being a broker between networks.

3.2.3.3 Cross-Level Term: $\theta_Q z_Q(a,x,b)$

A cross-level network effect is one that includes edges from all three – macro, meso, micro – networks. This means there are two types of cross-level structures that can occur: a 3-path and a 4-cycle.

The 3-path describes a structure where well-connected vertices in the macro network are connected through the meso network to well-connected vertices in the micro network. The L3AXB structure is shown as an example in Figure 12. However, it is important to notice that, unlike the 4-cycle which will be described next, only one of the vertices in the macro network and one of the vertices in the B network are connected. If this term were highly prevalent in the overall network, this would not mean that only one A vertex and only one B vertex would be connected for the entire network, but that this 3-path structure would occur frequently, so many A and B vertices would not connect directly but instead connect to the other network through popular vertices within their own network. This creates what Wang et al. (2013) call a hub where the connected vertices in each network use the hub vertex in order to connect through the meso network.

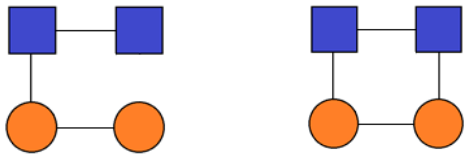

Figure 12: Cross-level 3-path and 4-cycle.

The 4-cycle is identical to the 3-path except for an edge connecting both the A and B vertices. The interpretation of this structure is that "members of connected groups are themselves connected" (Wang et al., 2013, p. 105). In the larger network, this structure creates networks in which cliques in one level are connected to cliques in another level.

3.2.4 Problems

Two major problems plague the fitting of models to multilevel networks: degeneracy (frozen graphs) and convergence. Just as in the single-level network example, model degeneracy can occur when a model is incorrectly parameterized. Because of the complexity of the multilevel network structures, this is easy to do without even realizing it since so many of the structures overlap and influence each other. In order to avoid this, Wang (2013) recommends starting with standard edge, Markov, and social circuit specifications for the within-network effects and just the basic edge effects for the between-network effects (p. 120). After successfully fitting those, more complicated interaction and cross-level effects can be fitted. However, it is important to note that, just as with the single-level networks, if a network structure effect is included in the model that does not appear in the observed network, this will also lead to degeneracy.

In addition to model degeneracy, convergence of parameter estimates is especially problematic in multilevel networks.

3.2.4.1 Convergence

With small networks and uncomplicated models, the estimate of the parameters for each effect is done through maximum likelihood estimation wherein the parameters are estimated as the most likely based on the observed data.

For example, in a network with three vertices and one edge modeled by an ERGM modeling just the edges, the maximum likelihood estimate (MLE) could be calculated by first finding the likelihood function and then maximizing the log of that function with respect to the edge parameter as shown in Equations 2 and 3. Because the log function is strictly increasing over (0,∞), the likelihood and log likelihood share the same maxima (Casella & Berger, 2002, p. 317).

$$L(\theta) = \frac{\exp(\theta \sum \sum_{i<j} y_{ij})}{k(\theta)} \tag{10}$$

$$where\ k(\theta) = 1 + 3\exp(\theta) + 3\exp(2\theta) + \exp(3\theta)$$

$$d(\log(L(\theta))) = d\left(\theta \sum \sum_{i<j} y_{ij} - \log(1 + 3\exp(\theta) + 3\exp(2\theta) + \exp(3\theta))\right) \tag{11}$$

Upon solving this equation, it can be seen that the MLE of the edge parameter θ for the 3-vertex network with one edge is –ln(2).

Using the discussion of edge probabilities from Chapter 2 and this parameter estimate, we can say that the probability of an edge occurring in this network conditional on the rest of the graph is 1/3, as shown in Equation 12.

$$\frac{\exp(-\ln(2))}{1+\exp(-\ln(2))} = 1/3 \quad (12)$$

However, as mentioned previously, when the network grows large and the model becomes more complicated, maximizing the log-likelihood function becomes much too intensive and impractical because the normalization constant becomes large and unruly. Because of this issue with finding the MLE of the parameter analytically in ERGMs, network analysts have adopted a few different approaches to parameter estimation including pseudo-maximum likelihood estimation (PMLE) and MCMC simulation methods of estimation.

MCMC methods described below are the most popular because PMLEs are often not accurate for many sets of data because the dependent observations create bias in the parameter estimates and inaccurate standard errors (Robins, Pattison, Kalish, & Lusher, 2007, pp. 186-187).

It can be shown that, for exponential family distributions, the value of the parameters that maximize the likelihood of the observed network occurring is both the MLE and method of moments estimator for parameters (Lehmann, 1983). The method of moments estimator is found by determining the values of the parameters for which $E(z(X)) = z(x_{obs})$ where x_{obs} is the observed network and $z(x_{obs})$ is a vector of statistics on the observed network. In other words, the best estimate of the parameters is the one whose expected value makes the set of network statistics equal to the observed network statistics (Koskinen & Snijders, 2013, p. 147). Therefore, if we were to simulate a plethora of networks using a set of parameters, we could expect that the observed network statistics would fall in the center of the simulation distribution for the parameter values if they were close to the method of moments estimator

or, equivalently, the MLE. If the first set of parameters chosen does not have the observed network's statistics in the center of the simulation distribution, then another set of parameters must be chosen to better accomplish that goal. This procedure needs to be repeated until $E(z(X)) = z(x_{obs})$, but "because the estimation procedure is numerical, we do not expect the quality to hold true exactly" (Koskinen & Snijders, 2013, p. 153). Note that simulations must be done because the sampling distribution of most networks is too large to enumerate.

There are different algorithms that perform this operation, including the Geyer-Thompson algorithm used by the popular statnet R package and the Robbins-Monro algorithm used in the popular PNET software (Koskinen & Snijders, 2013, pp. 149, 151).

Because the moment equality of expected and observed network statistics is not exact, it is simply important that the expected value of the statistics be very close to the observed statistics. Once it is very close, it is said to have converged to the MLE. In the context of the Robbins-Monro algorithm, very close is defined as being when the statistic in Equation 13,

$$\frac{\bar{z_\theta} - z_{obs}}{SD_\theta\left(\{z_k(x^{(m)})\}_{m=1,\ldots M_3}\right)} \qquad (13)$$

is close to zero where *m* is a graph from one simulation of the model with parameter θ^m (Koskinen & Snijders, 2013, p. 152). More specifically, it is considered close to zero when it is between -0.1 and 0.1. Beyond those values, it is said that the estimate for the parameters is too far from the MLE (Koskinen & Snijders, 2013, p. 154).

Sometimes parameters do not converge through this process. The process can be repeated a number of times by starting at the values last suggested by this simulation method.

However, even after many attempts, the parameters may not converge. One reason this can happen is that the model is degenerate for the reasons mentioned earlier.

When models do not converge, making sure to properly specify them can help achieve convergence. For example, in an application done by Zhao and Rank on workplace relations, the addition of Markov star effects helped improve the convergence of the model even though those specific effects were not significant in the model (Zhao & Rank, 2013, p. 219). Also, Koskinen and Snijders suggest trying conditional estimation by keeping the density fixed (Koskinen & Snijders, 2013, p. 154).

CHAPTER 4

APPLICATION OF MULTILEVEL ERGM

4.1 Introduction to Lawyer Network

In order to apply this multilevel network framework, we will be looking at a network of lawyers who work at a law firm in the Northeast, which was originally presented by Lazega and Pattison (1999). Data was collected on whom the lawyers look to for advice, who they consider friends, and with whom they work. Furthermore, some attribute data was recorded including gender, law school, seniority, age, type of practice, status (partner or associate), and office location.

By focusing on one of the dichotomous variables, status, we hope to discover whether the multilevel framework can shed more light on the structure of the advice network since "any network data involving binary categorization of nodes can be seen as a 'two-level' network" (Wang et al., 2013, p. 111). The status variable defines a lawyer as either a partner or associate within the law firm. For our purposes, the 36 partners will populate the A network while the B network will be comprised of 35 associates.

The overall network can be seen in Figure 14 where Network B has a density of 0.1798, Network X has a density of 0.0278, and Network A has a much higher density of

0.3794. Unsurprisingly, Network A also has a much higher average degree than Network B (6.6389 vs. 3.0571). Network X, which connects the partners and the associates, has relatively few edges, with an average degree of 0.9722 with A vertices and 1.000 with B vertices.

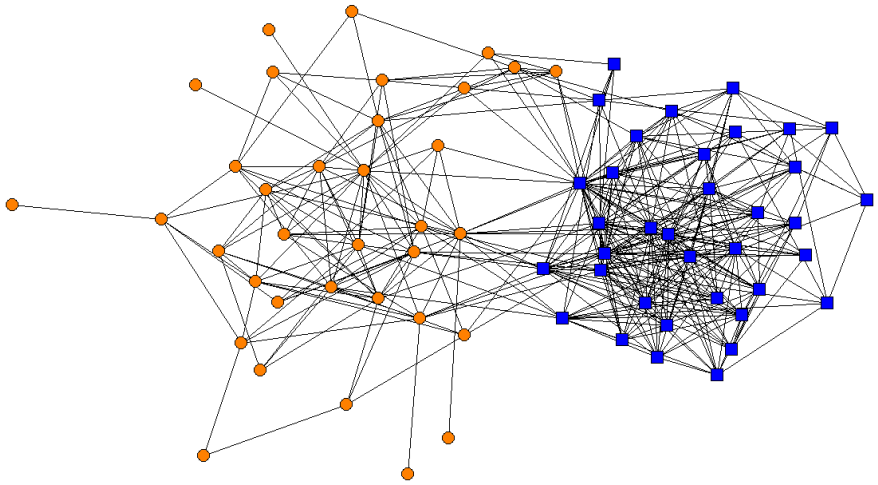

Figure 143: Lawyer network (blue – partners; orange – associates).

4.2 Model Selection

As mentioned in Chapter 3, it is recommended to first fit a model to each network level separately and then begin including interaction and cross-level terms into the model. Therefore, after several attempts to model both the partner (A) network and the associate (B) network using a variety of terms and refining each model by means of the MPNet software, a model with the basic Markov terms was chosen for each (Wang, Robins, & Pattison, 2009). Models with higher level terms such as alternating stars and alternating triangles would not converge. The models for each network can be found in Table 2.

Table 2: Model for Networks A and B Separately

Network A					
Effects	**Lambda**	**Parameter**	**StdErr**	**t-ratio**	**SACF**
EdgeA	2.0	-2.1888	0.388	0.066	0.845
Star2A	2.0	0.0647	0.027	-0.076	0.857
Star3A	2.0	-0.0065	0.003	-0.078	0.851
TriangleA	2.0	0.2113	0.081	-0.078	0.851
Network B					
EdgeB	2.0	-3.2070	0.485	-0.90	0.685
Star2B	2.0	0.1701	0.073	-0.074	0.673
Star3B	2.0	-0.0163	0.011	-0.061	0.644
TriangleB	2.0	0.1767	0.173	-0.061	0.635

When considering these models, it is of primary importance that they are able to converge in estimation because, as Robins and Lusher explain, "all models will produce numerical output . . . but the numbers for the estimates are meaningless unless all parameters have converged" (Robins & Lusher, 2013, p. 178). In practical terms, this means finding structures that appear in the network and have a real-world interpretation, estimating them using network software, and increasing the multiplication factor in the estimation until the model converges – if it is going to. The multiplication factor allows the network software to

spend more time tweaking the parameter estimates in order to find a set that will converge (Harrington, 2007).

While a number of network effects are modeled for these two networks, it is worth considering a few important ones that are not explicitly modeled such as degree distribution and clustering. In order to check how these models compare against the observed networks, goodness of fit statistics were run on each model in which 5,000 networks were simulated using the parameter estimates in order to ensure the models describe the data. The results can be seen in Table 3.

Table 3: GOF Statistics for Networks Modeled Separately

Network A				
Effects	**Observed**	**Mean**	**StdErr**	**t-ratio**
EdgeA	239.0	235.9836	65.798	0.046
Star2A	3486.0	3361.7338	1875.617	0.066
Star3A	18318.0	16972.3726	14162.716	0.095
TriangleA	579.0	538.1962	436.136	0.094
Stddev_degreeA	11.2029	7.7054	1.820	1.922
Skew_degreeA	1.0782	1.2426	0.091	-1.803
clusteringA	0.6186	0.4239	0.104	1.868
Network B				
EdgeB	107.0	110.1150	39.112	-0.080
Star2B	790.0	826.8168	543.246	-0.068
Star3B	2174.0	2282.3018	2100.713	-0.052
TriangleB	75.0	80.6488	72.199	-0.078
Stddev_degreeB	4.8889	4.3867	1.202	0.418
Skew_degreeB	1.6002	1.4341	0.160	1.038
clusteringB	0.2848	0.2470	0.078	0.487

Having established models for the A- and B- level networks, we turned to creating a model for the entire multilevel network. After introducing just the X-level edge term and achieving convergence, we moved to a slightly more complicated network which included Star2AX and Star2BX terms. Through many rounds of estimation, a model converged, and its results are seen in Table 4 and GOF statistics in Table 5. When efforts were made to try to improve this model by including triangle interaction terms (TXAX and TXBX), the model quickly became degenerate. Upon further investigation, it was revealed that the network had very few interaction triangles compared with other statistics in the model, as can be seen in Table 6. Once the TXAX and TXBX terms were removed from the model, it was no longer degenerate, as suggested in Chapters 2 and 3.

Our results indicate the prevalence of Star2AX structures, which occur when A vertices that are popular with other A vertices are also popular with other B vertices. In the context of this network where A vertices represent partners in the law firm and B vertices represent associates in the law firm, this suggests that partners who are popular in terms of advice sharing within their own network are also popular with associates in the B-level network. This interpretation is enhanced by the comparatively small number of Star2BX structures because this would entail associates who are popular advice sharers with their peers being popular advice sharers with partners. This scenario seems more unlikely considering partners would likely be more willing to seek advice from other partners.

Table 4: Multilevel Network Model Parameters

Effects	**Lambda**	**Parameter**	**StdErr**	**t-ratio**	**SACF**
EdgeA	2	-2.4767	0.394	0.024	0.233
Star2A	2	0.0874	0.031	0.025	0.267
Star3A	2	-0.0095	0.003	0.026	0.285
TriangleA	2	0.212	0.087	0.026	0.284
EdgeB	2	-3.2784	0.447	-0.019	0.195
Star2B	2	0.1797	0.07	-0.013	0.232
Star3B	2	-0.0187	0.011	-0.007	0.253
TriangleB	2	0.1822	0.168	-0.006	0.25
XEdge	2	-5.7426	0.506	0.027	0.298
Star2AX	2	0.1297	0.031	0.03	0.299
Star2BX	2	0.0269	0.035	0.024	0.308

Table 5: GOF Statistics for Multilevel Model

Effect	Observed	Mean	StdDev	t-ratio
EdgeA	239	240.1448	72.903	0.016
Star2A	3486	3522.227	2321.045	0.016
Star3A	18318	18630.0986	20009.512	0.016
TriangleA	579	589.6763	611.227	-0.017
EdgeB	107	104.9556	40.72	0.05
Star2B	790	757.4529	588.786	0.055
Star3B	2174	2042.2267	2400.73	0.055
TriangleB	75	70.7507	80.118	0.053
XEdge	35	35.5855	29.677	-0.02
Star2AX	632	646.207	837.749	-0.017
Star2BX	273	273.2115	389.896	-0.001
stddev_degreeA	12.1393	7.8225	1.959#	2.204
skew_degreeA	1.1013	1.2298	0.094	-1.366
clusteringA	0.6819	0.4314	0.111#	2.255
stddev_degreeX_A	2.2039	1.0248	0.438#	2.694
skew_degreeX_A	0.8183	1.124	0.521	-0.587
stddev_degreeX_B	1.5213	0.9285	0.325	1.824
skew_degreeX_B	0.3757	0.8846	0.476	-1.07
clusteringX	0.0778	0.0245	0.058	0.918

(continued on following page)

Table 5 (continued)

stddev_degreeB	5.9381	4.1698	1.18	1.499
skew_degreeB	1.2743	1.4427	0.164	-1.027
clusteringB	0.3291	0.2294	0.075	1.335

Table 6: Counts of Structures in Lawyer Network

Structure	**# Observed**
Edge A	239
Star2A	3486
Star3A	18318
TriA	579
Edge B	107
Star2B	790
Star3B	2174
TriB	75
Edge X	35
Star2AX	632
TXAX	25
Star2BX	273
TXBX	14

4.3 Concluding Remarks

While the multilevel framework illuminates some features that would not have been seen through single-level analysis, the process of model selection and refinement for multilevel ERGMs is as much of an art as it is a science. While this can be said of most model selection, the complex dependencies of network statistics makes the process much less direct. The choice of terms and simulation factors creates opportunities for many errors and many discussions in both journals and at conferences about the best way to proceed in building ERGMs. Even when a model is discovered that is not degenerate and does converge, there is still room for debate about whether it is the best model.

Beyond model selection, degeneracy, and convergence, a recent article by Shalizi and Rinaldo (2013) provides some evidence that ERGM parameters are not projectible from a sub-graph to a global network for any model other than the simple edge model. The theory surrounding this assertion is complex, but the implications are simple and concerning for the use of ERGMs:

1) ERGM parameter estimates for simple random graph models gained from a sample network can be projected to the population network, but that model does not adequately represent the observed network.
2) ERGM parameter estimates for more complicated model gained from a sample network cannot be used to describe the population network.

However, if a sample network is not being analyzed, then, as Robins et al. (2007) note, the interactions between terms is "an important model feature that assists with interpretation" (Robins, Pattison, Kalish, & Lusher, 2007, p. 184).

More specifically for multilevel ERGMs, a major barrier to analysis is lack of software options. MPNet works well, but its output and GOF leaves something to be desired even when compared to its single-level predecessor PNet (Wang et al., 2009). An additional barrier is the lack of network data that is prepared for multilevel analysis. Because multilevel analysis is not as popular as similar methods like analysis of affiliation networks, one of the networks usually lacks within-level ties, which hinders the multilevel framework from outshining other methods.

While there are concerns about the application of these models, there are also many opportunities to further clarify the procedures for proper modeling and investigate the underlying features of exponential random graph models.

REFERENCES

Casella, G., & Berger, R. L. (2002). *Statistical Inference.* Pacific Grove, CA: Duxbury.

Davis, A., Gardner, B. B., & Gardner, M. R. (1941). *Deep South: A Social Anthropological Study of Caste and Class.* Chicago: University of Chicago Press.

Erdos, P., & Renyi, A. (1959). On Random Graphs I. *Publicationes Mathematicae, 6*, 290-297.

Frank, O., & Strauss, D. (1986, Sep.). Markov Graphs. *Journal of the American Statistical Association, 81*(395), 832-842.

Gilbert, E. N. (1959). Random Graphs. *The Annals of Mathematical Statistics, 30*, 1141-1144.

Goodreau, S., Handcock, M., Hunter, D., Butts, C., & Morris, M. (2008, Feb.). A Statnet Tutorial. *Journal of Statistical Software, 24*(9). Retrieved from http://www.jstatsoft.org

Harrington, N. (2007). *PNet for Dummies.* Retrieved from http://www.mysmu.edu/faculty/nharrigan/PNetForDummies.pdf

Harris, J. (2014). *An Introduction to Exponential Random Graph Modeling.* Thousand Oaks: Sage Publications, Inc.

Holland, P. W., & Leinhardt, S. (1981, March). An Exponential Family of Probability Distributions for Directed Graphs. *Journal of the American Statistical Association, 76*(373), 33-50.

Hunter, D. R., & Handcock, M. S. (2006). Inference in Curved Exponential Family Models for Networks. *Journal of Computational and Graphical Statistics, 15*(3), 565-583.

Hunter, D., Handcock, M., Butts, C., Goodreau, S., & Morris, M. (2008, Feb.). ERGM: A Pacakage to Fit, Simulate and Diagnose Exponential-Family Models for Networks. *Journal of Statistical Software, 24*(3). Retrieved from http://www.jsstatsoft.org

Kolaczyk, E. D. (2010). *Statistical Analysis of Network Data.* New York: Springer Science+Business Media, LLC.

Koskinen, J., & Daraganova, G. (2013). Exponential Random Graph Model Fundamentals. In D. Lusher, J. Koskinen, & G. Robins (Eds.), *Exponential Random Graph Models for Social Networks* (pp. 49-76). New York: Cambridge University Press.

Koskinen, J., & Snijders, T. (2013). Simulation, Estimation, and Goodness of Fit. In D. Lusher, J. Koskinen, & G. Robins (Eds.), *Exponential Random Graph Models for Social Networks* (pp. 141-166). New York: Cambridge University Press.

Lazega, E., & Pattison, P. E. (1999). Multiplexity, Generalized Exchange and Cooperation in Organizations: A Case Study. *Social Networks, 21*, 67-90.

Lehmann, E. L. (1983). *Theory of Point Estimation.* New York: Wiley.

Lusher, D., Koskinen, J., & Robins, G. (2013). *Exponential Random Graph Models for Social Networks.* New York: Cambridge University Press.

Pattison, P., & Robins, G. (2002). Neighborhood-Based Models for Social Networks. *Sociological Methodology, 32*(1), 301-337.

Pattison, P., & Snijders, T. (2013). Modeling Social Networks: Next Steps. In D. Lusher, J. Koskinen, & G. Robins (Eds.), *Exponential Random Graph Models for Social Networks* (pp. 287-301). New York: Cambridge University Press.

Robins, G., & Lusher, D. (2013a). Illustrations: Simulation, Estimation, and Goodness of Fit. In D. Lusher, J. Koskinen, & G. Robins (Eds.), *Exponential Random Graph Models for Social Networks* (pp. 167-185). New York: Cambridge University Press.

Robins, G., & Lusher, D. (2013b). Simplified Account of an Exponential Random Graph Model as a Statistical Model. In D. Lusher, J. Koskinen, & G. Robins (Eds.), *Exponential Random Graph Models for Social Networks* (pp. 29-36). New York: Cambridge University Press.

Robins, G., Pattison, P., Kalish, Y., & Lusher, D. (2007). An Introduction to Exponential Random Graph (p*) Models for Social Networks. *Social Networks, 29*, 173-191.

Shalizi, C. R., & Rinaldo, A. (2013). Consistency Under Sampling of Exponential Random Graph Models. *The Annals of Statistics, 41*(2), 508-535.

Snijders, T., Pattison, P., Robins, G., & Handcock, M. (2006). New Specifications for Exponentional Random Graph Models. *Soociological Methodology, 36*(1), 99-153.

Wang, P. (2013). Exponential Random Graph Model Extensions: Models for Multiple Networks and Bipartite Networks. In D. Lusher, J. Koskinen, & G. Robins (Eds.), *Exponential Random Graph Models for Social Networks* (pp. 115-129). New York: Cambridge University Press.

Wang, P., Robins, G., & Pattison, P. (2009). MPNet. University of Melbourne. Retrieved from http://sna.unimelb.edu.au/PNet

Wang, P., Robins, G., Pattison, P., & Lazega, E. (2013). Exponential Random Graph Models for Multilevel Networks. *Social Networks, 35*(1), 96-115.

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

Watts, D., & Strogatz, S. (1998). Collective Dynamics of 'Small-World' Networks. *Nature, 393*(6684), 440-442.

Zhao, Y., & Rank, O. (2013). Interdependencies Between Working Relations: Multivariate ERGMs for Advice and Satisfaction. In D. Lusher, J. Koskinen, & G. Robins (Eds.), *Exponential Random Graph Models for Social Networks* (pp. 213-225). New York: Cambridge University Press.