TOPICS IN EXPONENTIAL RANDOM GRAPH MODELING

A Dissertation in
Statistics
by
Rashmi P. Bomiriya

© 2014 Rashmi P. Bomiriya

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

August 2014
The dissertation of Rashmi P. Bomiriya was reviewed and approved* by the following:

David R. Hunter  
Professor of Statistics, Head of the Department of Statistics  
Dissertation Advisor, Chair of Committee

Murali Haran  
Associate Professor of Statistics

Le Bao  
Assistant Professor of Statistics

Rachel Smith  
Associate Professor of Communication Arts & Sciences and Human Development & Family Studies

Aleksandra Slavkovic  
Associate Professor and Associate Head for Graduate Studies of Department of Statistics

*Signatures are on file in the Graduate School.
Abstract

Exponential-family Random Graph Models (ERGMs) are a class of models that is frequently used for modeling social networks. ERGMs allow structural features as well as covariate information on networks in the models. There are many network statistics that can be used in an ERGM that are already incorporated in the R package ‘ergm’. However, for certain types of networks such as bipartite networks, some standard network statistics cannot be applied as they are.

The first portion—chapter 2—of this report introduces a few network statistics that can be used to measure the homophily effects in a bipartite network with nodal attributes and we have already added this new model terms to the R package ‘ergm’. We provide some applications of these statistics and some simulations that study the performance of the introduced measure. This is joint work with Prof. David R. Hunter and Shweta Bansal.

The second portion—chapter 3—of this thesis extend the idea of Bayesian inference for ERGMs to Curved-like ERGMs. Currently, the R package Bergm provides tools for Bayesian inference for ERGMs (Caimo and Friel, 2013). The extensions in this chapter have been carried out by modifying the algorithm in Bergm—which is a combination of the exchange algorithm and the parallel adaptive direction sampler—and will soon be available as a part of the same package.

Chapter 4 of this report also relates to ERGMs, but describes a very different application of ERGMs—i.e., studying the dynamics of epidemics via network analysis techniques. Groendyke et al. (2011a) study the properties of a disease outbreak and the network on which it spread given epidemic data—with the help of their package ‘epinet’—that is assumed to come from a disease spread across some contact network, which they describe by an ERGM. However, the standard models have drawbacks in modeling varying shapes of degree distributions. Hence, we introduce a semi-parametric Bayesian model and study the improvement in the inferences made. Also, we describe few other useful extensions we have made to the package ‘epinet’. This is joint work with Dr. Michael Schweinberger.

Finally, in chapter 5, we summarize the work of this dissertation and discuss a few paths of future developments—mostly on chapter 4. Here, we introduce some existing problems and possible directions for tackling them.
# Table of Contents

List of Figures viii  
List of Tables xii  

## Chapter 1  
**Modelling Social Networks**  

1.1 Introduction .................................................. 1  
1.2 Exponential-family Random Graph Models and Homophily .................. 3  
   1.2.1 Parameter Estimation and Pseudolikelihood ...................... 7  
   1.2.1.1 Derivation of the pseudo-loglikelihood ..................... 9  
   1.2.1.2 MCMC Algorithms for Sampling Networks .................. 10  
1.3 Analyzing Bipartite Networks .................................. 13  
   1.3.1 Bipartite Clustering Coefficient ............................ 15  
   1.3.2 Choice Statistics ........................................... 16  
   1.3.3 Event 2-stars and Actor 2-stars ............................. 17  
   1.3.4 Alternating-k-stars & Alternating-k-two-paths Statistics ...... 18
List of Figures

1.1 An example of an undirected network: red circles represent the nodes/individuals in the network and black lines represent the “friendships” (edges) between individuals. 2

1.2 An example of the matrix representation of a network and network statistics. Matrix in left corresponds to the adjacency matrix of the network in the middle. 4

1.3 An example for gender-based homophily statistics; female-homophily statistic is the number of circle-based edges in the box on the right. male-homophily statistic is the number of square-based edges shown in the same box. Overall gender-homophily (i.e., uniform homophily) statistic is given by the sum of female- and male-homophily statistics. 5

1.4 An example of the construction of $y_{ij}^+$ and $y_{ij}^-$ networks. Take the matrix representation of the observed network $y$ and force the value of element $y_{ij}$ to 1 (as in the left matrix) to obtain the network $y_{ij}^+$ and force the same element to zero (as in the right matrix) to obtain $y_{ij}^-$. 6

1.5 A sketch of the global expansion network: circles and squares correspond to companies and cities respectively. Ac.1, Ac.2,... stand for accounting companies, Ad.1, Ad.2,... stand for advertising companies, B.1, B.2,... stand for banking and finance companies, L.1, L.2,... stand for law companies whereas C.1, ... , C.55 represent the 55 cities. 14

1.6 $L_3$ and $C_4$ configurations of a bipartite network. Green circles correspond to b1-nodes and red squares correspond to b2-nodes. 15
1.7 Graphical representation of the basis of choice statistics with actor and event attributes (based on Agneessens et al., 2004). A represent any actor or event attribute, whereas $u$ represents a specific actor attribute and $w$ represents a specific event attribute. For example, $u-w$ represents a tie between an actor of type $u$ and an event of type $w$. ................................................................. 17

1.8 Graphical representation of event 2-stars with actor and event attributes (based on Agneessens et al., 2004) ................................................................. 18

2.1 Networks (b) and (c) depict the projections of the original two-mode network (a) onto mode 1 nodes (circles) and mode 2 nodes (squares), respectively. An edge between a pair of nodes exists in the projection if and only if there is at least one two-path between those two nodes in (a). ................................................................. 23

2.2 Mode-1 nodes are circles and mode-2 nodes are squares, while dashed and continuous outlines indicate different levels of a mode-1 nodal attribute. In (a), edge $i \leftrightarrow k$ is part of two distinct two-paths joining matching mode-1 nodes. In (b), we see the projection of (a) onto mode-1 nodes, with multiplicity indicated as appropriate. ................................................................. 28

2.3 Let circles represent firms with similar characteristics (say ‘Advertising’ firms) and squares represent cities. Nodelpair-wise counting is then counting the number of common cities for each pair of advertising firms, whereas edge-wise counting involves for each advertising firm $i$ located in city $k$, counting the number of other advertising firms located in city $k$. ................................................................. 30

2.4 Loglikelihood vs. discount parameter for different datasets. For each dataset, the best discount parameter is a $\beta$ value, indicated by a vertical dashed arrow. .... 35

2.5 MCMC diagnostic plots of ERGMs fitted for boards data with $\beta = 0.25$ (left) and for global expansion data with $\beta = 0$ (right). ................................................................. 37

2.6 Observed statistics (solid line) vs. simulated statistics (boxplots) for the degree distributions of mode-2 (firms) nodes and the dyadwise shared partner statistics for the global expansion dataset. ................................................................. 38

2.7 Loglikelihood vs. discount parameter for 10 simulated networks based on a true model with $\beta = 0.5$ and $\alpha = 0.5$ discount parameters respectively. .......... 39
3.1 Parallel Adaptive Directional Update (Example): The green circles denote the current values of the six chains and the red star denotes the proposed value for the next state of chain $h$. ................................................................. 49

3.2 MCMC diagnostic plots for a $\beta$-based ERGM fitted for global expansion data (Based on EX-ADS-NORM algorithm). $\theta_5$ stands for the fifth parameter in the model, $\beta$. Left most plot is the posterior density of $\beta$ followed by the time-series plot of the estimates and the right most plot is the autocorrelation plot of $\beta$ samples. The results are based on 10000 main iterations and 10 parallel Markov chains with thinning $= 10$. ................................................................. 57

3.3 MCMC diagnostic plots for parameter estimates of a $\beta$-based ERGM fitted for global expansion data (Based on EX-ADS-NORM algorithm). $\theta_1, \ldots, \theta_4$ stand for edge, b2nodematch-Advertising, b2nodematch-BankingFinance and b2nodematch-Law parameters in the model, respectively. ................................................................. 58

4.1 SEIR process for individual $i$ who was infected by individual $j$ ................................................................. 64

4.2 Data-generating process: (a) generate a contact network; (b) generate an epidemic conditional on the contact network; initially infected population member is in green and labeled by $\kappa$, other infected members are in red, susceptible population members are in black; black lines indicate contacts, blue arrows indicate transmissions. ................................................................. 64

4.3 Left plot (a): The number of individuals in the infectious state over time under the Erdős-Rényi model (Captured from Groendyke et al. (2012a)). The observed measles epidemic is given by the solid red line, whereas simulated data are shown in black, with boxplots summaries shown at each 5-day time increment. Right plot (b): The same plot obtained for the newly implemented semiparametric model with 2 blocks and without covariates (at each 5-day time increment). Here, the day axis has been centered at the mode of the distribution. ................................................................. 77

4.4 A timeline that displays some of the notations in this section. ................................................................. 78

4.5 Posterior distributions of a few removal times along with true removal times; dashed lines correspond to true removal times. ................................................................. 83

4.6 Credible intervals of the parameters $\eta_{E,1}$, $\eta_{E,2}$, $\eta_{I,1}$, $\eta_{I,2}$, $\beta$, $\gamma_1$, $\gamma_2$, and $\gamma_3$ based on the results by running a semi-parametric model with 3 blocks on 1000 simulated datasets. ................................................................. 85
4.7 Estimates of the block parameters $\gamma_1 = -2$, $\gamma_2 = -1$ and $\gamma_3 = 0$ based on the results of running a semi-parametric model with 3 blocks on 1000 simulated datasets with an MCMC sample of size 1000, for each dataset.

4.8 Posterior medians of the parameters $k_E$, $\theta_E$, $k_I$, $\theta_I$, $\beta$, $\gamma_1$, $\gamma_2$ and $\gamma_3$ based on the results by running a semi-parametric model with 3 blocks on 1000 simulated datasets.

4.9 RMSEs for $\beta$, $\gamma_1$, $\gamma_2$, $\gamma_3$ for varying sample sizes.

4.10 RMSEs for $k_I$, $k_E$, $\theta_I$, $\theta_E$ for varying sample sizes.

5.1 An example of a transmission tree update that should be possible with the new MCMC algorithm for an epidemic with two root nodes.
List of Tables

2.1 Boards Data: Parameter estimates and standard errors, along with `ergm` code used to obtain them. Statistical significance at the 0.0001 level is indicated by ***. 35

2.2 Global Expansion Data: Parameter estimates and standard errors, along with `ergm` code used to obtain them. Statistical significance at the 0.0001 level is indicated by ***. 36

2.3 Parameter estimates used in simulation study. 39

3.1 Global Expansion Data: Acceptance Rates and Time Taken for the algorithms EX-ADS-MIX (Mixture Proposal), EX-ADS-BETA (Dependent Beta Proposal) and EX-ADS-NORM (Multivariate Normal Proposal) each with 10000 main iterations, 20000 auxiliary iterations, and 500 burnin iterations. 56

3.2 Global Expansion Data: Parameter estimates; MCMC-MLE (with corresponding standard error) and posterior means (with corresponding standard deviations) for MLE-based and Bayesian methods, respectively. 57

3.3 Global Expansion Data: Posterior means along with 2.5% and 97.5% quantiles of the posterior for five networks simulated from the model `network ~ edges + b2nodematch('firm.type', diff = TRUE, keep = 2:4, β = 0)` (Based on the results from EX-ADS-NORM algorithm). First row shows the true values of the parameters. 59

4.1 Frequentist coverage properties of 95% posterior credible intervals: Number of times 95% posterior credible intervals covered data-generating values of parameters $β$, $k_E$, $θ_E$, $k_I$, $θ_I$, $γ_1$, $γ_2$ and $γ_3$, based on models with three blocks and five blocks, respectively. 85
Global Expansion Data: Parameter estimates (posterior means) based on 10000 MCMC iterations of the algorithms EX-ADS-MIX (Mixture Proposal), EX-ADS-BETA (Dependent Beta Proposal) and EX-ADS-NORM (Multivariate Normal Proposal).
Modelling Social Networks

1.1 Introduction

Social Network Analysis (SNA) has become a widely used technique for understanding and modeling relational data in a variety of fields such as sociology, biology, geography, etc. In SNA, we represent the underlying social structure by a network—a set of nodes (individuals/subjects) that are connected by edges (relationships between pairs of nodes) as shown in Figure 1.1. The type of relationships that we consider as edges might vary depending on the scenario. For example, in one network, an edge may represent friendship between two individuals, whereas in other networks it may represent relationships such as kinship, collaborations, citations, etc. More formally, a network is a duple $G = (V, E)$, where $V$ is a set of nodes and $E \subseteq V \times V$ is the set of edges. Often, we eliminate the pairs $(x, x)$ from the set of edges $E$, where $x \in V$—i.e. $(x, x) \notin E$—to avoid self edges in the network. If we let $(x, y) \in E \iff (y, x) \in E$, then we call it an undirected network—i.e. If A is a friend of B, then B is a friend of A and vice versa. If $(x, y) \in E$ does not guarantee that $(y, x) \in E$, we call it a directed network—i.e. even though A considers B as a
friend, B may not consider so.

Figure 1.1: An example of an undirected network: red circles represent the nodes/individuals in the network and black lines represent the "friendships" (edges) between individuals.

We can think of an observed network (Say, with a fixed set of \( n \) nodes) as one realization of a larger set of possible networks with the same nodes. In other words, we can assume that the observed network is generated by some unknown stochastic process. In real social networks, certain sub-structures or features might occur more frequently than we would expect if they occur by chance (Robins et al., 2007a). Therefore, it is sensible to make use of statistical/probabilistic approaches to model social networks.

The purpose of the current chapter is to set the background for the remaining chapters, especially for Chapter 2, which deals with modelling homophily in bipartite networks. In Section 1.2, we introduce a class of statistical models called the Exponential-family Random Graph Models (ERGMs), which are often used in modelling social networks. In Section 1.3, we describe “bipartite networks”—a special form of networks—in detail and provide some notions of analyzing bipartite networks. In Section 2.1, we introduce the concept of measuring homophily in a standard unipartite network and also explain why it is not so straightforward to extend the existing unipar-
tite homophily measures into the bipartite setting. We provide a possible solution—our proposed method in Chapter 2. The content of Chapter 2 is based on a manuscript in progress and is joint work with Prof. David R. Hunter and Shweta Bansal.

Chapter 3 of this thesis extends the Bayesian inference idea for random graph models from ERGMs to Curved ERGMs and other ERGMs that include network statistics that depend on some parameters. This is an extension of the work of Caimo and Friel (2011) that is available to the public in the form of an R package named ‘Bergm’. We are currently collaborating with Alberto Caimo in order to include the extensions in this chapter as new features of the package ‘Bergm’.

Chapter 4 of this report discusses a few extensions to the work of Groendyke et al. (2012a) in studying the dynamics of epidemics and once again it involves ERGMs, but in a way that is different to Chapter 2. The content of this chapter is joint work with Dr. Michael Schweinberger. The tools developed by Groendyke et al. (2012a) are available in the R package ‘epinet’. Therefore, we have incorporated all the extensions we describe in this chapter in the same package.

Lastly, Chapter 5 describes how the main Chapters 2, 3 and 4 of this thesis relate to each other and discusses some avenues for future work. The existing problems and possible directions for resolving them are discussed here.

### 1.2 Exponential-family Random Graph Models and Homophily

If we allow the $n \times n$ matrix $Y$ to encode the status of all the edges of the network, where $Y_{ij}$ equals 0 or 1 according to whether the $(i, j)^{th}$ edge is absent or present (unless $Y_{ij} = Y_{ji}$, this is a
representation of a directed network), then the basic ERGM model may be written as

\[
P_\theta(Y = y) = \frac{\exp\{\sum_{i=1}^{p} \theta_i s_i(y)\}}{\kappa(\theta)},
\]

where \(s_1(y), \ldots, s_p(y)\) are user-defined statistics measured on the network \(y\) and we denote the vector of all network statistics by \(s(y)\) (See Figure 1.2). When covariates \(X\) are also needed to be included in the model, we may add \(X\) to the notation and write \(s(y, X)\) (Handcock et al., 2010), where we allow these statistics to depend on any available known covariates, \(\theta_1, \ldots, \theta_p\) are corresponding unknown coefficients to be estimated, and \(\kappa(\theta)\) is a normalizer necessary to ensure that Equation (2.1) defines a legitimate probability distribution. Social ties are more (or less) likely to form between individuals who share similar characteristics. This phenomenon is called homophily (anti-homophily). For example, individuals who are of the same gender may be more likely to become friends than who are of different genders. We can test such hypotheses by including suitable homophily statistics into an ERGM. If the coefficient \(\theta\) corresponding to the homophily statistic is significantly different from zero, then, depending on its sign we can detect homophily, if the sign is positive, or, detect anti-homophily, if the sign is negative. A simple homophiliy statistic that can be used in an ERGM in the above unipartite example, is the number of edges that are made up of individuals of the same gender. As depicted in Figure 1.3, if we consider the number of edges that are formed due to male-male and female-female combinations separately, then we refer

\begin{tabular}{|c|c|c|c|c|}
\hline
 & A & B & C & D & E \\
\hline
A & 0 & 1 & 1 & 1 & 1 \\
B & 1 & 0 & 0 & 0 & 0 \\
C & 1 & 0 & 0 & 1 & 0 \\
D & 1 & 0 & 1 & 0 & 0 \\
E & 1 & 0 & 0 & 0 & 0 \\
\hline
\end{tabular}

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{network_matrix.png}
\caption{An example of the matrix representation of a network and network statistics. Matrix in left corresponds to the adjacency matrix of the network in the middle.}
\end{figure}
to them as *differential-homophily* statistics, whereas, if we consider both combinations together, we call it *uniform-homophily*. The measures for homophily in non-unipartite networks, however, is no longer trivial. Refer Chapter 2 for more information, which includes a detailed discussion on homophily for a special form of networks known as *bipartite* networks.

![Figure 1.3](image.png)

**Figure 1.3:** An example for gender-based homophily statistics; female-homophily statistic is the number of circle-based edges in the box on the right. male-homophily statistic is the number of square-based edges shown in the same box. Overall gender-homophily (i.e., uniform homophily) statistic is given by the sum of female- and male-homophily statistics.

Let $y_{ij}^+$ and $y_{ij}^-$ represent the networks obtained by setting the $ij^{th}$ element $y_{ij} = 1$ and $y_{ij} = 0$ (See Figure 1.4), respectively, while holding all the other elements of the matrix $y$ as they are. Define $\delta_s(y)_{ij}$, the change statistic, as

$$
\delta_s(y)_{ij} = s(y_{ij}^+) - s(y_{ij}^-) 
$$

(Handcock et al., 2010). Here, $\delta_s(y)_{ij}$ represents the change in the value of the network statistic $s(y)$ that would occur if $y_{ij}$ was changed from 0 to 1 while the rest of the elements in $y$ remain unchanged. For example, if $s(y)$ statistic is the number of edges, then $\delta_s(y)_{ij}$ will always be one as the difference between the number of edges in the networks $y_{ij}^+$ and $y_{ij}^-$ is exactly one. The model in Equation (2.1) implies Equation (1.3), which corresponds to the distribution of the Bernoulli
Figure 1.4: An example of the construction of $y_{ij}^+$ and $y_{ij}^-$ networks. Take the matrix representation of the observed network $y$ and force the value of element $y_{ij}$ to 1 (as in the left matrix) to obtain the network $y_{ij}^+$ and force the same element to zero (as in the right matrix) to obtain $y_{ij}^-$. 

The interpretation of coefficients becomes much easier once the model is viewed in the form of Equation (1.3). Each component of $\theta$ would correspond to the change in conditional log-odds of creating an edge between nodes $i$ and $j$ (i.e. for e.g. $i^{th}$ and $j^{th}$ individuals becoming friends) with a one unit increase in the corresponding component of $s(y)$, while keeping the rest of the network statistics fixed as they are.
We discuss various applications of ERGM-based modeling in Chapters 2, 3, and 4.

1.2.1 Parameter Estimation and Pseudolikelihood

In this section we review the ERGM estimation procedure. One way to try estimating the ERGM parameters is to obtain the Maximum Likelihood Estimators (MLEs) by maximizing the true log-likelihood given by

\[ l(\theta) = \sum_{i=1}^{p} \theta_i s_i(y^{obs}) - \log \kappa(\theta) = \theta^T s(y^{obs}) - \log \kappa(\theta), \] (1.4)

where \( y^{obs} \) is the observed network and \( \kappa(\theta) \) is the sum of \( \exp\{\theta^T s(y)\} \) for all possible networks \( y \).

The MLE would be the same as the Method of Moment Estimate (MME) in this case due to exponential family.

We obtain the MMEs by solving the moment equation

\[ E_{\theta}[s(Y)] = s(y^{obs}), \] (1.5)

where \( s(y^{obs}) \) is the observed value of the sufficient statistic and \( s(Y) \) is a random value generated from the model with parameter \( \theta \). But as \( E_{\theta}[s(Y)] = \sum_{y \in Y} s(y) P_\theta(y) \), where \( Y \) is the set of all possible networks, the moment equation becomes

\[ \sum_{y \in Y} s(y) P_\theta(y) = s(y^{obs}). \] (1.6)

Also, recall that we obtain the MLEs by differentiating Equation 1.4 with respect to \( \theta \) and setting
it to 0. Thus, we have

$$\frac{\partial}{\partial \theta} \log P_\theta(y_{obs}) = s(y_{obs}) - \frac{\partial}{\partial \theta} \log \left\{ \sum_{y \in Y} e^{\theta^T s(y)} \right\} = s(y_{obs}) - \frac{\sum_{y \in Y} s(y) e^{\theta^T s(y)}}{\sum_{x \in Y} e^{\theta^T s(x)}} = s(y_{obs}) - \sum_{y \in Y} s(y) P_\theta(y) = 0,$$

(1.7)

which is the same as the moment equation. Hence, the MLE and MME are the same in this situation.

It is evident that \( \kappa(\theta) \) is not easily computable in general as there are \( 2^{(n^2)} \) possible undirected networks with \( n \) nodes and for some models \( \kappa(\theta) \) may only be evaluated as a sum with this number of terms. Therefore, the common practice is to approximate maximum likelihood via simulation methods that involve Markov Chain Monte Carlo (MCMC) techniques for simulating random networks (Hunter et al., 2008b).

For a moment, suppose that there is no dependence among the \( y_{ij} \) in Equation (1.3). Then, we can replace \( P_{\theta,Y}(Y_{ij} = 1|Y^c_{ij} = y^c_{ij}) \) by \( P_{\theta,Y}(Y_{ij} = 1) \), making Equation (1.3) look like an ordinary logistic regression model. In that case, we can obtain the MLE of the parameters by traditional logistic regression. In an ERGM, the independence of \( y_{ij} \) is not guaranteed. This is one of the major differences between the traditional logistic regression model and the ERGM. The estimation carried out by ignoring this dependence and maximizing Equation 1.11 even though it is not the true likelihood is called maximizing the “pseudolikelihood” (Say \( L_p \)) and the corresponding estimates are called the “maximum pseudolikelihood estimates” (MPLEs). The derivation of the explicit form of the pseudo-loglikelihood (See Equation 1.11) denoted by \( l_p(\theta, y) \) is given below. MPLEs are easier to compute, however, we only have limited knowledge on the quality of these estimates (van Duijn et al., 2009).
1.2.1.1 Derivation of the pseudo-loglikelihood

We begin by writing the standard form of the pseudo-likelihood, which is a special case of composite likelihood—only ignores the dependence in the sense that it multiplies the conditional probabilities.

\[
L_p(\theta, y) = \prod_{i,j} P(y_{ij} = 1 \mid Y_{ij}^c = y_{ij}^c)^{y_{ij}} P(y_{ij} = 0 \mid Y_{ij}^c = y_{ij}^c)^{(1-y_{ij})}.
\]  

(1.8)

Then, by taking log on both sides, we obtain the pseudo-loglikelihood,

\[
l_p(\theta, y) = \sum_{i,j} [y_{ij} \log \{ P(y_{ij} = 1 \mid Y_{ij}^c = y_{ij}^c) \}] + \sum_{i,j} [(1 - y_{ij}) \log \{ P(y_{ij} = 0 \mid Y_{ij}^c = y_{ij}^c) \}].
\]

(1.9)

By Equation (1.3), we can easily obtain the values of \( P(y_{ij} = 1 \mid Y_{ij}^c = y_{ij}^c) \) and \( P(y_{ij} = 0 \mid Y_{ij}^c = y_{ij}^c) \) in terms of the change statistics and substitute into Equation (1.9) to obtain

\[
l_p(\theta, y) = \theta^T \sum_{i,j} y_{ij} \delta_s(y)_{ij} - \sum_{i,j} [y_{ij} \log(1 + \exp\{\theta^T \delta_s(y)_{ij}\})] - \sum_{i,j} [(1 - y_{ij}) \log(1 + \exp\{\theta^T \delta_s(y)_{ij}\})].
\]

(1.10)

By simplifying Equation 1.10, we obtain the explicit form of the pseudo-loglikelihood as

\[
l_p(\theta, y) = \theta^T \sum_{i,j} y_{ij} \delta_s(y)_{ij} - \sum_{i,j} \log(1 + \exp\{\theta^T \delta_s(y)_{ij}\}).
\]

(1.11)

In the case of an undirected network with a dyadic independence ERGM, it is possible to replace \( P_{\theta,Y}(Y_{ij} = 1 \mid Y_{ij}^c = y_{ij}^c) \) by \( P_{\theta,Y}(Y_{ij} = 1) \). Hence, the MPLE is the true MLE in that case. A dyadic independence term is an ERGM term for which the corresponding network change statistic \( \delta_s(y)_{ij} \) can be obtained without any knowledge of \( y \) except for the value of \( y_{ji} \) (in the case
of a directed network). Therefore, the term dyadic independence ERGM above refers to an ERGM in which all the terms are dyadic independence terms (Handcock et al. (2010)). When the model for an undirected network does not satisfy dyadic independence it is preferable to use a stochastic MCMC algorithm such as the one described below to approximate the likelihood instead of using the MPLE.

As direct maximization of the true likelihood is impossible, we use an indirect approach in which the first step is to write down the log-likelihood ratio based on some fixed parameter vector $\theta_0$:

$$ l(\theta) - l(\theta_0) = (\theta - \theta_0)^T s(y^{obs}) - \log E_{\theta_0}[\exp\{(\theta - \theta_0)^T s(Y)\}] $$

(1.12)

Next, we randomly sample networks $Y_1, ..., Y_m$ ($m$ is a sufficiently large integer) from $P_{\theta_0}$ and approximate the expectation term by its sample average making use of the law of large numbers, as follows:

$$ l(\theta) - l(\theta_0) \approx (\theta - \theta_0)^T s(y^{obs}) - \log \left[ \frac{1}{m} \sum_{i=1}^{m} \exp\{(\theta - \theta_0)^T s(Y)\} \right] $$

(1.13)

Lastly, we maximize the approximate log-likelihood ratio in Equation (1.13) in order to obtain the parameter estimates of $\theta$ (Hummel et al. (2012)).

### 1.2.1.2 MCMC Algorithms for Sampling Networks

We obtain a sample of networks via MCMC. The idea is to simulate a discrete-time Markov chain whose stationary distribution is $P_{\theta_0}$. We will discuss two ways in which the MCMC can be run. The first way is to use Gibbs sampling whereas the second approach is to use a Metropolis algorithm.

---

**Gibbs Sampling:**

- First, select a pair of nodes $(i, j)$ uniformly at random from all possible pairs.
- By Equation 1.3, it can be easily seen that
\[ P_{\theta_0, \gamma}(Y_{ij} = 1|Y_{ij}^c = y_{ij}^c) = \frac{\exp \{\theta_0^T \delta_s(y)_{ij}\}}{1 + \exp \{\theta_0^T \delta_s(y)_{ij}\}}. \]

- Then, based on this conditional distribution of \( Y_{ij} \) given the rest of the network \( Y_{ij}^c \), make a decision whether to set \( Y_{ij} = 0 \) or \( Y_{ij} = 1 \)—i.e, with probability \( P_{\theta_0, \gamma}(Y_{ij} = 1|Y_{ij}^c = y_{ij}^c) \), set \( Y_{ij} = 1 \), otherwise, set \( Y_{ij} = 0 \).

- Repeat this process until you obtain the desired number of networks (Hunter et al., 2008b).

---Metropolis Algorithm:

- Once again, select a pair of nodes \((i, j)\) uniformly at random from all possible pairs.

- Then, calculate the acceptance ratio \( \pi \) as

\[ \pi = \frac{P(Y_{ij} \text{ changes} \mid Y_{ij}^c = y_{ij}^c)}{P(Y_{ij} \text{ stays the same} \mid Y_{ij}^c = y_{ij}^c)} = \exp\{\pm \theta_0^T \delta_s(y)_{ij}\} \]

  i.e. if the current state of \( Y_{ij} = 0 \), then

\[ \pi = \frac{P(Y_{ij} = 1 \mid Y_{ij}^c = y_{ij}^c)}{P(Y_{ij} = 0 \mid Y_{ij}^c = y_{ij}^c)} = \exp\{\theta_0^T \delta_s(y)_{ij}\}, \]

  whereas if the current state of \( Y_{ij} = 1 \), then

\[ \pi = \frac{P(Y_{ij} = 0 \mid Y_{ij}^c = y_{ij}^c)}{P(Y_{ij} = 1 \mid Y_{ij}^c = y_{ij}^c)} = \exp\{-\theta_0^T \delta_s(y)_{ij}\} \]

- At the next step, accept the change of \( Y_{ij} \) with probability \( \min\{1, \pi\} \).

- Repeat this process until you obtain the desired number of networks (Hunter et al., 2008b).

In order to run the MCMC for sampling networks, it is not required to know the individual values of \( s(y_{ij}^+) \) or \( s(y_{ij}^-) \) and knowing only the corresponding change statistics \( \delta_s(y)_{ij} \) is sufficient.

In both of the above approaches, we select a dyad uniformly at random to be toggled. However, when the underlying network is sparse—i.e. links exist only between a small proportion of all possible dyads—choosing a pair of nodes uniformly at random frequently suggests to switch-on a link rather than taking an existing one away. Due to the low density in sparse networks, this will lead the Markov chains to stuck in the same state for many iterations. The TNT (tie/no
tie) sampler (Morris et al., 2008) given below is one way to remedy this issue by choosing pairs of nodes with links more often than they would be, if chosen uniformly at random. Thus, TNT sampler will move around the whole space of possible networks more effectively than the other two methods above.

—TNT (Tie/No Tie) Sampler:

TNT sampler is a Metropolis-Hastings algorithm and the steps given below are for the version of the TNT sampler that has been implemented in the R package, \texttt{ergm}. Denote the total number of edges in the network at current state by $n_{edges}$ and the number of all possible dyads in the network by $ndyads$.

- Choose a connected dyad, i.e., a tie, with probability $w$ (say $w = 0.5$) to toggle, otherwise select any dyad, either connected or disconnected, uniformly at random.

- Based on the current state of the selected dyad $(i, j)$, calculate the Hastings-factor $\rho$.

  - If an edge is chosen, i.e., if $y_{ij} = 1$,
    \[ \rho = \begin{cases} 
    \frac{1}{w * ndyads + (1 - w)} & \text{if } n_{edges} = 1; \\
    1 + \left( \frac{w}{1 - w} \right) \left[ \frac{ndyads}{n_{edges} + 1} \right] & \text{if } n_{edges} > 1. 
    \end{cases} \]

  - If a non-edge is chosen, i.e., if $y_{ij} = 0$,
    \[ \rho = \begin{cases} 
    w * ndyads + (1 - w) & \text{if } n_{edges} = 0; \\
    1 + \left( \frac{w}{1 - w} \right) \left[ \frac{ndyads}{n_{edges} + 1} \right] & \text{if } n_{edges} > 0. 
    \end{cases} \]

- Calculate the acceptance probability $\pi$ as

  \[ \pi = \begin{cases} 
    \exp\{-\theta_0^T \delta_s(y)_{ij}\} * \rho & \text{if } y_{ij} = 1; \\
    \exp\{\theta_0^T \delta_s(y)_{ij}\} * \rho & \text{if } y_{ij} = 0. 
    \end{cases} \]
• Accept the change of $Y_{ij}$ with probability $\min\{1, \pi\}$.

• Repeat this process until you obtain the desired number of networks.

### 1.3 Analyzing Bipartite Networks

Mathematically, a bipartite network can be represented as a triple $G = (V_1, V_2, E)$ where $V_1$ and $V_2$ are two disjoint sets of nodes and $E \subseteq V_1 \times V_2$ is the set of edges. The disjoints sets $V_1$ and $V_2$ are often called first- and second-mode nodes and a tie may occur only between a first-mode node and a second-mode node. Bipartite networks are also known as affiliation or two-mode networks, and some examples for networks of this nature includes citation networks, collaboration networks, actor-movie networks etc. Some bipartite networks are composed of modes of the same type of nodes, e.g., heterosexual contact networks, and in such networks, homophily may be handled just as in the standard, non-bipartite case; however, we often find applications in which the two modes of nodes are entirely different types of entities. Homophily among entities within the first- or the second-mode nodes here is no longer obvious. The first- and second-mode nodes are sometimes called as “actors” and “events”, though in some situations, this terminology is not so meaningful. Therefore, we frequently refer to them as mode-1 and mode-2 nodes in this thesis.

An example of bipartite network dataset is given below. We refer to this network as the “global expansion” dataset and it consists of $n_1 = 55$ cities around the world and $n_2 = 46$ global firms, defined as firms that have offices in at least 15 cities worldwide (Taylor and Walker, 2001). An edge in this network represents the presence of a given firm in a given city and we find 1342 edges connecting these cities and firms. We have information on the type of each of these firms, which is one of “Accountancy,” “Advertising,” “Banking & Finance,” or “Law.” The original dataset has a service level value for each edge, ranging from 0 (absence) to 3 (major presence). For the purposes of this paper, we have recoded any positive (1 to 3) service level value as 1, so that we look at only the presence or absence of a global firm in any given city. A sketch of this global expansion...
network is shown in Figure 1.5. This is the primary dataset that we use throughout Chapter 2 and Chapter 3 for illustration purposes and for simulation studies.

![Figure 1.5: A sketch of the global expansion network: circles and squares correspond to companies and cities respectively. Ac.1, Ac.2,... stand for accounting companies, Ad.1, Ad.2,... stand for advertising companies, B.1, B.2,... stand for banking and finance companies, L.1, L.2,... stand for law companies whereas C.1, ... , C.55 represent the 55 cities.](image)

Borgatti and Everett (1997a) review the basic notions of analyzing bipartite networks, and explain how most unipartite network techniques and measurements such as visualization of networks, nodal degree, network density, betweenness and centrality can be extended to the bipartite case. More recently, Latapy et al. (2008) propose extensions to a few common notions such as clustering to two-mode networks, as well as introduce novel statistics like the “redundancy coefficient” or expand on others such as the bipartite clustering coefficient of Robins and Alexander (2004) (See Section 1.3.1). Various references to bipartite network applications are provided in Latapy et al. (2008) and here, the notion of projection is discussed at length, which leads to possibly the simplest method for dealing with homophily, the idea we describe briefly in Section 1.2, and it is discussed at length once more in Chapter 2. The idea of one-mode projection of two-mode networks is illustrated in Chapter 2. However, as Latapy et al. (2008) point out, we may lose
important information on the original network due to projection and it may also alter certain measures as clustering coefficients inappropriately, making the projection approach less ideal for analyzing bipartite networks.

We will now look at some of the existing notions of analyzing bipartite networks in detail.

### 1.3.1 Bipartite Clustering Coefficient

The bipartite clustering coefficient introduced by Robins and Alexander (2004) is motivated by the concept of 1-mode clustering coefficient—calculated as three times the ratio of number of triangles to number of 2-stars. In the bipartite case, the clustering coefficient still remains a ratio of two network statistics but with triangles and 2-stars replaced by the number of $C_4$ configurations and the number of $L_3$ configurations in the network that we describe below. An $L_3$ configuration, as shown in the left hand side of Figure 1.6, is a possible 3-path in a bipartite network. A $C_4$ configuration on the other hand is the simplest form of a cycle that can occur in a bipartite network. Note that every $C_4$ configuration includes four $L_3$ configurations. The bipartite clustering coefficient is then defined as the ratio $4 \times \text{Number of } C_4 \text{ Configurations}/\text{Number of } L_3 \text{ Configurations}$ (Robins and Alexander, 2004).

**Figure 1.6:** $L_3$ and $C_4$ configurations of a bipartite network. Green circles correspond to b1-nodes and red squares correspond to b2-nodes

If we represent b1-nodes as company directors and b2-nodes as the companies that they are
affiliated with as in Robins and Alexander (2004), a high clustering coefficient indicates that there are more frequent personal connections between the pairs of directors. In addition, it means that many of the $L_3$ configurations in this network are redundant, i.e., they do not provide a lot of room for new paths of connectivity across the underlying bipartite network (Robins and Alexander, 2004).

As we can see, the clustering coefficients provide some useful information about bipartite networks in terms of the level of connectivity between the pairs of nodes. However, clustering coefficient is not a measure that captures the homophily effects due to specific actor or event attributes present in the data.

### 1.3.2 Choice Statistics

Consider a bipartite network that consists of a set of actors and events. Suppose that an actor-attribute with $U$ categories and an event-attribute with $W$ categories are observed (i.e. we have $UW$ actor-event specific attribute combinations). Figure 1.7 graphically represents the possible combinations of these actor-event ties. A generic actor or event category is denoted by $A$ in this figure. For example, the $u - A$ tie in the lower left corner of Figure 1.7 represents a tie between an actor with the $u$-attribute type and any event (irrespective of the event type).

Agneessens et al. (2004) defines four types of choice statistics based on the different types of ties shown in Figure 1.7. $C_{A,A}$, $C_{A,w}$, $C_{u,A}$ and $C_{u,w}$ denote the statistics obtained by counting the all ties in the network of the types $A-A$, $A-w$, $u-A$, and, $u-w$, respectively. For example, $C_{A,A}$ statistic equals the total number of edges in the network. Let $\theta_{C_{A,A}}$, $\theta_{C_{A,w}}$, $\theta_{C_{u,A}}$ and $\theta_{C_{u,w}}$ denote the corresponding ERGM parameters of these statistics once you include them in an ERGM. Then, (i) $\theta_{C_{A,A}}$ is a general choice parameter which refers to the logs odds of a tie irrespective of the actor and event attributes, (ii) $\theta_{C_{A,w}}$ is a choice parameter for the $w^{th}$ event category irrespective of the category for the actor attribute, (iii) $\theta_{C_{u,A}}$ is a choice parameter for the $u^{th}$ actor category irrespective of the category for the event attribute and (iv) $\theta_{C_{u,w}}$ is a choice parameter for the $u^{th}$
Figure 1.7: Graphical representation of the basis of choice statistics with actor and event attributes (based on Agneessens et al., 2004). \( A \) represent any actor or event attribute, whereas \( u \) represents a specific actor attribute and \( w \) represents a specific event attribute. For example, \( u-w \) represents a tie between an actor of type \( u \) and an event of type \( w \).

actor category and \( w^{th} \) event category. Here, \( w \in \{1, ..., W\} \) and \( u \in \{1, ..., U\} \).

1.3.3 Event 2-stars and Actor 2-stars

The choice statistics discussed in the previous section focus on ties between one actor and one event. By contrast, Agneessens et al. (2004) define an event 2-star as a pair of actors who have ties with the same event. On the other hand, an actor 2-star is a pair of events connected to the same actor. As before, we consider an actor attribute with \( U \) categories and an event attribute with \( W \) categories.

Based on the eight possible types of ties in Figure 1.8, Agneessens et al. (2004) defines a set of event 2-star statistics: (i) general event 2-star statistic \( S_{AA,A} \) that ignores the attribute categories of both actors and the event in consideration, (ii) \( S_{uA,A} \) or \( S_{Av,A} \)—a statistic that takes into account just one actor attribute and ignores the other actor attribute and the event attribute, (iii) \( S_{uv,A} \)—a statistic that focusses on both actor attributes \( u \) and \( v \) but ignores the event attribute, and so on.
Similarly, the actor 2-star statistics are defined by $S_{AA,A}$, $S_{A,wA}$, $S_{A,Ay}$, $S_{A,wy}$, $S_{u,AA}$, $S_{u,wA}$, $S_{u,Ay}$ and $S_{u,wy}$. These statistics could be described just like the event 2-star statistics by replacing the two actors with attributes $u$ and $v$ by two events with attributes $w$ and $y$ and the event with attribute $w$ by an actor with an attribute $u$. Then, the idea is to include all these terms in a probability model to account for homophily effects.

### 1.3.4 Alternating-k-stars & Alternating-k-two-paths Statistics

The direct counting of two stars or two paths may lead to the so-called model degeneracy problem discussed in Robins et al. (2007b) in great detail. In this section, we describe two statistics that can be used to overcome this issue in the bipartite case, introduced by Wang et al. (2009) by extending the idea of alternating-$k$-stars and alternating-$k$-two-paths ideas for unipartite networks of Hunter (2007a). However, these statistics are not meant for measuring homophily in bipartite networks though one may extend them to do so. We briefly discuss how one may extend these statistics to homophily-based statistics in Chapter 2.
1.3.4.1 Alternating-k-star Statistics

Consider a bipartite network \( y \) with \( n \) mode-1 nodes and \( m \) mode-2 nodes.

Denote the mode-1-\( k \)-star and mode-2-\( k \)-star statistics by \( Z_{1k} \) and \( Z_{2k} \), respectively. Letting \( d_{1i} \) and \( d_{2j} \) denote the degree of mode-1 node \( i \) and mode-2 node \( j \), respectively, we have

\[
Z_{1k}(y) = \sum_{k=1}^{n} \binom{d_{1i}}{k} \quad \text{and} \quad Z_{2k}(y) = \sum_{k=1}^{m} \binom{d_{2j}}{k}.
\] (1.14)

Wang et al. (2009) define mode-1-alternating-\( k \)-stars as

\[
Z_{\text{alt.}k.\text{star.}1}(\lambda, y) = \sum_{k=2}^{m} (-1)^k \frac{Z_{1k}(y)}{\lambda^{k-2}}
\] (1.15)

and mode-2-alternating-\( k \)-stars as

\[
Z_{\text{alt.}k.\text{star.}2}(\lambda, y) = \sum_{k=2}^{n} (-1)^k \frac{Z_{2k}(y)}{\lambda^{k-2}},
\] (1.16)

where \( \lambda \) is a weighting parameter that satisfies \( \lambda \geq 1 \). When \( \lambda = 1 \), Equations (1.15) & (1.16) become

\[
Z_{\text{alt.}k.\text{star.}1}(\lambda = 1, y) = 2 \ast Z(y) - n + \sum_{i=1}^{n} I\{d_{1i} = 0\},
\] (1.17)

\[
Z_{\text{alt.}k.\text{star.}2}(\lambda = 1, y) = 2 \ast Z(y) - m + \sum_{i=1}^{m} I\{d_{2j} = 0\},
\] (1.18)

where \( Z(y) \) denotes the total number of edges in the network and \( I\{d_{1i} = 0\} \) and \( I\{d_{2j} = 0\} \) are indicator functions.
1.3.4.2 Alternating-k-twopath Statistics

Denote the number of two-paths between mode-2 nodes $i$ and $l$ by $L_{1il}$. Then Wang et al. (2009) express the number of mode-1-$k$-two-paths (i.e., $k$-two-paths centered at mode-1 nodes) as

$$Z^*_1(y, k) = \begin{cases} 
\sum_{i=1}^{l-1} \sum_{l=2}^m \binom{L_{2il}}{k} & \text{if } k > 2; \\
\frac{1}{2} \sum_{i=1}^{l-1} \sum_{l=2}^m \binom{L_{2il}}{k} & \text{if } k = 2 \text{ (due to symmetry)}. 
\end{cases}$$

Furthermore, they define mode-1-alternating-$k$-twopaths (say $Z_{\text{alt.path.1}.k}$) by

$$Z_{\text{alt.path.1}.k}(\lambda, y) = Z^*_1(y, 1) - \frac{2Z^*_1(y, 2)}{\lambda} + \sum_{i=3}^{m-2} \frac{(-1)^{i-1}Z^*_1(y, i)}{\lambda} = \left[ \lambda \sum_{i=1}^{l-1} \sum_{l=2}^m \{1 - (1 - \frac{1}{\lambda})^{L_{2il}}\} \right],$$

where $\lambda$ is once again a weighting parameter that satisfies the condition $\lambda \geq 1$. Mode-2-alternating-$k$-twopath statistics are defined similarly. In the special case of $\lambda = 1$, Equation (1.19) reduces to the number of mode-2 dyads that are indirectly connected by at least one two-path,

$$Z_{\text{alt.path.1}.k}(\lambda = 1, y) = \sum_{i=1}^{l-1} \sum_{l=2}^m I\{L_{2il} > 0\}$$

(1.20)
Modelling Homophily in ERGMs for Bipartite Networks

This chapter is based on a manuscript that was recently submitted to the *Network Science* journal. Thus, some information from Chapter 1 is repeated in this chapter.

For a network in which nodes represent individuals and edges represent, say, social ties, the basic idea of homophily is that ties are more likely to form between individuals who share one or more attributes in common than would otherwise be expected. In other words, homophily leads individuals to form homogeneous groups. The importance of homophily in directing social relationships is well-established in the sociology literature; for instance, see McPherson *et al.* (2001).

Furthermore, it is often straightforward to incorporate homophily into statistical models for social networks. A commonly-used modeling framework used to accommodate and estimate the effect of homophily in forming social networks is the exponential-family random graph model (ERGM), or p-star, framework. For an in-depth introduction to ERGMs, see Robins *et al.* (2007a). In this framework, which we discuss in greater detail in Section 2.1, homophily related to variable X may be woven into the model by ensuring that the sufficient statistics that define the model
include the count of all ties between two individuals who match on X. For instance, we may model
the propensity of children in school to forms friendship ties preferentially with members of their
class by including the count of all friendship ties between children in the same class as one of
the model statistics. This modeling practice is not only straightforward but, as Hunter et al.
(2008b) point out, including homophily statistics in an ERGM in this way does not destroy a
particularly appealing mathematical characteristic of the resulting models—dyadic independence.
Dyadic independence is helpful because it enables ease of estimation of the statistical parameters
(the regression coefficients) associated with each of the statistics of the model. In brief, this ease
is due to the fact that ERGMs exhibiting dyadic independence are equivalent to standard logistic
regression models.

Yet dealing with homophily is no longer straightforward for bipartite networks, also known as
affiliation or two-mode networks. Technically, a bipartite network is a triple \( G = (V_1, V_2, E) \) where
\( V_1 \) and \( V_2 \) are two disjoint sets of nodes, referred to here as the two modes, and \( E \subseteq V_1 \times V_2 \)
is the set of edges. More colloquially, bipartite networks are those in which nodes are split into
two subgroups, or modes, and an edge may only connect a first-mode node and a second-mode
node. Some bipartite networks are composed of modes of the same node type, and in these cases
homophily may be considered as in the standard, non-bipartite case. For instance, in a heterosexual
contact network, in which partnerships are only allowed between individuals of the opposite sex, it
is meaningful to study homophily within a single partnership. However, quite often the two modes
of nodes are comprised of entirely different types of entities, and homophily is only meaningful
among entities of the same type. It is these latter cases in which the current article is relevant.
Occasionally, first- and second-mode nodes are referred to as “actors” and “events,” though this
terminology may not be appropriate in all contexts and so in this article we simply use “mode 1”
and “mode 2” to refer to the two modes.

Basic notions of analyzing bipartite networks are reviewed by Borgatti and Everett (1997b), who
explain how many familiar network techniques and measurements—e.g., visualization of networks,
nodal degree, network density, betweenness and centrality—may be extended to the bipartite case.
More recently, Latapy et al. (2008) similarly extend several common notions such as clustering to two-mode networks, introducing novel statistics like the “redundancy coefficient” or expanding on others such as the bipartite clustering coefficient of Robins and Alexander (2004). Latapy et al. (2008) also include numerous references to applications of bipartite networks and discuss at length the notion of projection, which gives rise to perhaps the simplest method for dealing with homophily. To wit, any two-mode network gives rise to two separate one-mode projections, as illustrated by Figure 2.1, and each projection may use a standard model such as an ERGM that takes homophily into account. Yet as Latapy et al. (2008) point out, the projection approach is not ideal for several reasons: In brief, the projection loses much information about the original bipartite structure of the network, and it may even distort certain types of information such as clustering coefficients.

![Figure 2.1](image)

**Figure 2.1:** Networks (b) and (c) depict the projections of the original two-mode network (a) onto mode 1 nodes (circles) and mode 2 nodes (squares), respectively. An edge between a pair of nodes exists in the projection if and only if there is at least one two-path between those two nodes in (a).

A method for dealing with homophily that retains all relevant bipartite information is therefore desirable, and we wish to focus on the particular ERGM paradigm in this article. Authors such as Faust et al. (2002) and Agneessens et al. (2004) incorporate homophily effects in ERGMs by considering two-stars classified according to the categories of their endpoints. However, ERGMs that use two-star statistics are known to give rise to the peculiar problem of degeneracy, an issue we discuss in Section 2.1, which can make model-fitting and simulation of random networks from a fitted model problematic. This article introduces a modification of the two-star statistics designed to avoid some of these issues in ERGMs that incorporate homophily effects.
As will become evident in Section 2.1, it is possible to combine many different effects into a single ERGM for bipartite network data. For this purpose, we refer interested readers to existing surveys of ERGM techniques specifically tailored for bipartite networks, such as the articles by Wang et al. (2009) and Wang (2013). In addition, Wang et al. (2013) covers certain ERGM ideas in a context that generalizes bipartite networks in a sense—indeed, this article discusses homophily as it applies in this broader context, though it takes an approach more similar in spirit to that of Agneessens et al. (2004) than the one we propose here. Yet broader modeling questions of how one might combine homophily effects with other structural effects are beyond the scope of the present article; here, we keep the focus squarely on homophily effects.

Consider the example of a dataset consisting of 55 cities around the world and 46 global firms, defined as firms that have offices in at least 15 cities worldwide (Taylor and Walker, 2001). These cities and firms are connected via 1342 edges, where an edge in this network represents the presence of a given firm in a given city. The attribute of interest for matching is the firm type, which is one of “Accountancy,” “Advertising,” “Banking & Finance,” or “Law.” The original dataset has a service level value for each edge, ranging from 0 (absence) to 3 (major presence). However, for the purpose of this paper we only consider the presence or absence of a global firm in any given city: We have recoded any positive (1 to 3) service level value as 1. We wish to know whether edge formation in this network exhibits homophilous behavior: For example, do firms of the same type tend to congregate in the same cities more or less frequently than we would expect by chance? We consider this question in Section 2.4.

The rest of this article, in which we introduce a novel way to model homophily in ERGMs for bipartite networks and then demonstrate the proposed method on real world applications, is organized as follows. Section 2.1 explains the ERGM framework and discusses why bipartite networks pose a problem where homophily is concerned. Section 2.2 introduces a new modification of the usual ERGM bipartite statistics that helps to ameliorate this problem, and Section 2.3 shows that this modification may be viewed as a special case within the general class of so-called curved exponential-family models. Section 2.4 demonstrates the potential effectiveness of
the new modification by applying it to two real datasets: The global firms and cities dataset described above, and another dataset involving movies and individuals who have positively rated these movies. Section 2.5 provides a simulation study to illustrate the efficacy of our estimation procedure and Section 2.6 offers some concluding remarks.

2.1 Exponential-family Random Graph Models

If we allow the $n \times n$ matrix $Y$ to encode the status of all the edges of the network, where $Y_{ij}$ equals 0 or 1 according to whether the $(i,j)$th edge is absent or present, then the basic ERGM may be written

$$P_\theta(Y = y) = \frac{\exp\{\sum_{i=1}^p \theta_i s_i(y)\}}{\kappa(\theta)}, \quad y \in \mathcal{Y},$$

(2.1)

where $s_1(y), \ldots, s_p(y)$ are user-defined statistics measured on the network $y$ and we denote the vector of all network statistics by $s(y)$. When covariates $X$ should be included in the model, we may add $X$ to the notation and write $s(y, X)$, where we allow these statistics to depend on any available known covariates (Hunter et al., 2008b). The parameters $\theta_1, \ldots, \theta_p$ are the corresponding unknown coefficients to be estimated, $\mathcal{Y}$ is the set of all allowable networks, and $\kappa(\theta)$ is merely a normalizer necessary to ensure that Equation (2.1) defines a legitimate probability distribution.

A simple illustrative model for mutual friendships among a group of people that assumes all undirected edges form independently with some probability $p_1$ between people of the opposite sex and with some different probability $p_2$ between people of the same sex can be written as

$$P(Y = y) \propto \exp \left\{ \theta_1 \sum_{i<j} y_{ij} + \theta_2 \sum_{i<j} y_{ij} I\{i \text{ and } j \text{ are of the same sex} \} \right\}.$$

(2.2)

One may show without much difficulty that under model (2.2), $p_1$ and $p_2$ are given by $e^{\theta_1}/(1 + e^{\theta_1})$ and $e^{\theta_1 + \theta_2}/(1 + e^{\theta_1 + \theta_2})$, respectively. The second statistic in Equation (2.2)—the multiplier of the $\theta_2$ coefficient—is actually a count of the number of edges that connect nodes of the same sex, and
it is this term that assures that homophily (if $\theta_2 > 0$) or heterophily (if $\theta_2 < 0$) is taken into account in the model. In the \texttt{ergm} package, this statistic may be incorporated into a model by adding \texttt{nodematch("sex")} to the model specification. Fellows and Handcock (2012) show that this statistic can lead to problems even when the network is not bipartite in the case when covariates are considered to be random variables. However, in the context of this article, nodal covariates are assumed to be fixed and observed, and the model of Equation (2.2) is straightforward.

By contrast, an analogous method of measuring homophily in the context of bipartite graphs would include a statistic counting two-paths connecting nodes of the same category, since it is impossible that such nodes are connected directly. For example, homophily in the global firms network would correspond to two firms located in the same city having the same firm type attribute, such as “Advertising.” Every two-path is also a two-star—i.e., a central node connected to two neighbors—and yet it may not be desirable to count every two-star with matching endpoints, for two reasons.

First is a type of law of diminishing returns: If two nodes are already connected via a two-path, every additional two-path connecting them might be less important than the previous one. A method to model this type of diminishing effect is via the alternating $k$-twopath or alternating $k$-star statistics of Robins \textit{et al.} (2007b). These statistics are reformulated as geometrically weighted degree and geometrically weighted dyadic shared partner statistics, respectively, by Hunter (2007b); later we will see a relationship between the methods we propose and the latter of these two statistics.

The second reason is a well-studied feature of certain ERGMs called \textit{degeneracy}. The basic issue of degeneracy is described by Handcock (2003) and Schweinberger (2011). As these articles point out, the basic problem is that it is possible, by adding a single edge to a network in a particular configuration, to increase the number of two-stars (say) by an arbitrarily large number, up to the number of first- or second-mode nodes. Since other statistics cannot compensate for this large increase in such situations, degenerate models often put inappropriately high probability mass on networks with very large or very small numbers of edges, depending on whether the
coefficient of the two-star term is positive or negative, even with coefficient values that are in some sense optimal. We do not delve into the technical details of this issue, instead referring interested readers to the two papers cited above.

The approach we outline in the next section involves a pair of homophily-based statistics that may be easily incorporated into an ERGM. Our new statistics introduce two sliding scales, at one end of which we find the full two-star statistic that can sometimes prove problematic. The other end of the scale counts only the first two-star formed by each pair of matching nodes, or the first two-star (connecting matching nodes) that contains each edge, depending on the interpretation of homophily we choose.

2.2 Modeling homophily

We let $n = n_1 + n_2$ denote the number of nodes in the bipartite network, where nodes 1 through $n_1$ are of mode 1 and nodes $n_1 + 1$ through $n$ are of mode 2. Suppose we want to model the homophily effect of a particular categorical nodal attribute measured on the mode 1 nodes. For example, gender may be considered as such a measure, if mode 1 nodes represent people. We assume here that the variable of interest is applicable only to the mode 1 nodes; however, a similar argument would apply in the case where we wished instead to model a homophily effect on mode 2 nodes.

As a starting point for measuring the degree of homophily, let us consider the analogue of the homophily statistic in Equation (2.2), namely, the total number of two-paths that link one mode-1 node with another mode-1 node of the same category. This statistic may be obtained by summing $y_{ik}y_{jk}$ for all matching mode-1 nodes $i$ and $j$ and all mode-2 nodes $k$. Thus, we write the basic homophily statistic as

$$b1\text{nodematch}(y) = \frac{1}{2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_1} \sum_{k=n_1+1}^{n} y_{ik}y_{jk} \mathbb{I}\{i \text{ matches } j, \ i \neq j\}, \quad (2.3)$$
where the “b1” in b1nodematch stands for bipartite of mode 1 and \( I\{\cdot\} \) is the indicator function taking either a zero or one depending on the falsity or truth of the enclosed condition. We divide by 2 because the summation double-counts every relevant two-path.

A simple reformulation of Equation (2.3) gives

\[
b1\text{nodematch}(y) = \frac{1}{2} \sum_{i=1}^{n_1} \sum_{k=n_1+1}^{n} y_{ik} \left[ \sum_{j \neq i} y_{jk} I\{i \text{ matches } j\} \right].
\]  

(2.4)

Thus, for every edge \( i \leftrightarrow k \), we count the number of two-stars containing this edge, sum, and finally divide by two. We illustrate this edge-centered view of the homophily statistic in Figure 2.2(a). There are two mode 1 nodes that match node \( i \) (the categories are denoted by dashed and continuous lines in the figure) that are also connected to \( i \) through \( k \). Thus, the value in square brackets in Equation (2.4) is 2.

---

**Figure 2.2:** Mode-1 nodes are circles and mode-2 nodes are squares, while dashed and continuous outlines indicate different levels of a mode-1 nodal attribute. In (a), edge \( i \leftrightarrow k \) is part of two distinct two-paths joining matching mode-1 nodes. In (b), we see the projection of (a) onto mode-1 nodes, with multiplicity indicated as appropriate.

By contrast, a different reformulation of Equation (2.3) gives

\[
b1\text{nodematch}(y) = \frac{1}{2} \sum_{i \neq j} \sum_{i \text{ matches } j} \left[ \# \text{ of two-paths from } i \text{ to } j \right],
\]  

(2.5)

where the sum is taken over all pairs \( i \neq j \) of matching mode 1 nodes. We refer to Equation (2.5) as the nodepair-centered view (See Figure 2.3), because it counts the number of two-paths that
connect each matching pair of nodes. We illustrate this view in Figure 2.2(b), where we see that the value in square brackets in Equation (2.5) is 1 when \( j = b \), 2 when \( j = a \).

Our modification to the two-star count involves modifying either Equation (2.4) or Equation (2.5) by adding an exponent in the range \([0, 1]\) (where we use the convention that \(0^0 = 0\)). In the nodepair-centered view (2.5), we use the exponent \( \alpha \), which results in

\[
\text{b1nodematch}(y; \alpha) = \frac{1}{2} \sum_{i \neq j} \sum_{i \text{ matches } j} |\text{# of two-paths from } i \text{ to } j|^\alpha \quad \text{with } 0 \leq \alpha \leq 1. \tag{2.6}
\]

On the other hand, if we take the edge-centered view (2.4), we use the exponent \( \beta \), which gives

\[
\text{b1nodematch}(y; \beta) = \frac{1}{2} \sum_{i=1}^{n_1} \sum_{k=n_1+1}^{n_1+n_2} y_{ik} \left[ \sum_{j \neq i} y_{jk} \right]^{\beta} \quad \text{with } 0 \leq \beta \leq 1. \tag{2.7}
\]

From Figure 2.2 we see that only the nodepair-based count above is preserved in the one-mode projection of a bipartite network.

To understand how statistics (2.6) and (2.7) operate, it is helpful to examine the extreme cases. When \( \alpha = 1 \) or \( \beta = 1 \), the two statistics coincide with each other and with the full count of two-stars with matching endpoints. On the other end of the spectrum, when \( \alpha = 0 \), statistic (2.6) counts the number of matching node pairs connected by at least one two-path. When \( \beta = 0 \), statistic (2.7) counts the number of edges involved in at least one matching two-star (remember that we use the convention \(0^0 = 0\)). If the alternating \( k \)-twopath statistic of Wang et al. (2009, Equation (6.13)) is extended to a matching-attribute-based alternating \( k \)-twopath statistic, then statistic (2.6) with \( \alpha = 0 \) is equivalent to its special case of \( \lambda_t = 1 \). However, we are not aware of any other direct correspondences between existing ERGM statistics and either of Equations (2.6) or (2.7). Each of the b1nodematch statistics discussed above may be decomposed, if desired, into distinct sub-statistics, one for each of the distinct values of the variable that defines matching. For instance, in the case of gender homophily, we might consider two separate statistics, one for males and one for females; alternatively, we might sum these statistics together. When using the
Figure 2.3: Let circles represent firms with similar characteristics (say ‘Advertising’ firms) and squares represent cities. Nodepair-wise counting is then counting the number of common cities for each pair of advertising firms, whereas edge-wise counting involves for each advertising firm $i$ located in city $k$, counting the number of other advertising firms located in city $k$.

$b1nodematch$ term in the $\text{ergm}$ package, the former case, called differential homophily, is invoked using $\text{diff=TRUE}$, while the latter, uniform homophily, uses $\text{diff=FALSE}$. The default value, if no value for $\text{diff}$ is specified, is uniform homophily. Alternatively, we might wish to examine how the coefficient for the $b1nodematch$ ($b2nodematch$) statistic changes across the levels of some mode 2 attribute (mode 1 attribute). This feature has been incorporated into the $\text{ergm}$ package. For instance, in the global expansion network described in Section 1.3, suppose we want to examine the interaction of homophily on the mode-2 attribute $\text{firm.type}$ with the mode-1 attribute $\text{region}$ that is defined on the cities, we may use $\text{b2nodematch("firm.type", b1 = "region")}$ in R. We illustrate some of these features in Section 2.4.

2.3 Curved exponential-family models

Hunter and Handcock (2006) demonstrated the utility of applying the statistical concept of curved exponential-family models (Efron, 1975, 1978) to the modeling of networks. In a curved ERGM,
the standard ERGM of Equation (2.1) is modified by assuming that the linear combination of the \( s_i(y) \) statistics is not defined by the \( \theta \) parameters directly, but rather by some function \( \eta(\theta) \). Thus, Equation (2.1) is replaced by

\[
P(Y = y) = \exp\left\{ \sum_{i=1}^{p} \eta_i(\theta) s_i(y) \right\} \kappa[\eta(\theta)],
\]

(2.8)

where \( s(y) \) is a \( p \)-dimensional vector of network statistics on \( y \), as before, but now \( \theta \), the parameter vector of interest is \( q \)-dimensional for some \( q < p \). The two vectors are related by the \( p \)-dimensional vector \( \eta \), which is assumed to be a function of \( \theta \). As usual, statistical estimation focuses on the maximum likelihood estimator of \( \theta \),

\[
\hat{\theta} = \arg \max_{\theta \in \mathbb{R}^q} \frac{\exp\left\{ \sum_{i=1}^{p} \eta_i(\theta) s_i(y) \right\}}{\kappa[\eta(\theta)]}.
\]

Here, we demonstrate that the statistics defined by Equations (2.6) and (2.7) may be rewritten in curved exponential-family form, where \( \alpha \) or \( \beta \) plays the role of the \( \theta \) parameter of interest seen in Equation (2.8).

Recall that \( n_2 \) is the number of mode 2 nodes in the network. Let us define, for \( 0 \leq i \leq n_2 \), the mode-1 matching dyadwise shared partner statistic \( \text{b1MDSP}_i(y) \) to equal the number of matching pairs of mode-1 nodes that have exactly \( i \) common (mode-2) neighbors. Then we may rewrite Equation (2.6) as

\[
\text{b1nodematch}(y, \alpha) = \sum_{i=1}^{n_2} (i^\alpha) * \text{b1MDSP}_i(y).
\]

(2.9)

Similarly, for any \( 0 \leq i \leq n_1 - 1 \), if \( \text{b1MESP}_i(y) \) is the number of edges in \( y \) that form a “mode-1 matching two-path” with exactly \( i \) other edges—i.e., that are contained in exactly \( i \) two-paths connecting two mode-1 nodes that match on the attribute of interest—then we may rewrite Equation (2.7) as

\[
\text{b1nodematch}(y, \beta) = \frac{1}{2} \sum_{i=1}^{n_1-1} (i^\beta) * \text{b1MESP}_i(y).
\]

(2.10)
The b1MDSP and b1MESP statistics may be viewed as roughly analogous to the DP and EP statistics that give rise to the geometrically weighted dyadwise shared partner (GWDSP) and edgewise shared partner (GWESP) statistics explained in Hunter (2007b). In fact, the b1MDSP statistics are precisely a homophily-based version of the DP (dyadic shared partner) statistics in Hunter (2007b, Equation (26)), which demonstrates that when $\beta = 0$, our Equation (2.10) is a homophily-based special case of the alternating $k$-twopath statistic of Robins et al. (2007b).

However, the analogy between b1MESP and EP statistics is not quite as direct. Indeed, the EP statistics of Hunter (2007b)—and thus the alternating $k$-triangle statistic of Robins et al. (2007b)—are meaningless in a bipartite context since triangles are impossible.

If we view $\alpha$ as a fixed constant in Equation (2.9), then the coefficients $i^\alpha$—which are the parameters $\eta_i(\theta)$ seen in Equation (2.8)—are fixed. Therefore the “curvature” is lost, and the model is a standard (non-curved) ERGM. On the other hand, if we view $\alpha$ as an unknown parameter to be estimated, then the complications of curved exponential families, as explained in Hunter and Handcock (2006), arise. The same arguments are true of the $\beta$ in Equation (2.10). Since the capability of dealing with these complications automatically is not currently implemented in the ergm package, we use an alternative technique in the numerical examples of Sections 2.4 and 2.5: We fit separate models for each value of $\alpha$ or $\beta$ on a grid of points from 0 to 1.

2.4 Applications

In this section, we demonstrate that the proposed bipartite homophily statistics give rise to meaningful and interpretable results for real datasets. Also, we show how widely the optimal values of the exponents $\alpha$ and $\beta$, which we call discount parameters, may vary depending on the application. However, we will not consider a full range of potential ERGM statistics that could potentially be used together with these new homophily statistics. Instead, we refer interested readers to review articles devoted to the modeling of bipartite networks using ERGMs, such as those of Wang et al. (2009) or Wang (2013), for a fuller treatment of the broad range of statistics that might be used.
in conjunction with the statistics we introduce here.

On the other hand, we shall include one or two additional statistics along with the homophily statistics in this section. First, we will always use the total number of edges in the network as an ERGM statistic. This term controls the overall density of the network, which is useful in much the same way that it is typically useful to include an intercept term in any logistic regression model unless there is a good reason not to do so.

The second term we will sometimes use attempts to correct for a certain type of confounding. To understand this confounding, it is helpful to consider the modeling of homophily in the case of a unipartite network. In the model of Equation (2.2), the edges term (which is the sum multiplied by $\theta_1$ in that model) is included not only because it is the intercept, but because without this term in the model, the $\theta_2$ coefficient would not be directly interpretable as the homophily effect—instead, this effect would be confounded with the overall density effect, and the two effects could not be measured separately. In other words, interpretation of the homophily effect $\theta_2$ requires us to correct for the overall density using the edge statistic. Similarly, when using the new homophily statistic in an ERGM, we will sometimes include another statistic that is a duplicate of the homophily statistic except for one detail: It assumes that all nodes match one another. In this way, the homophily term will capture only the additional effect due to the matching.

In practice, we can create this additional statistic by creating a “dummy” nodal covariate that takes the same value for every node. We may then add a b1nodematch (or b2nodematch) term to the model based on this dummy covariate with the same value of the $\alpha$ or $\beta$ discount parameter used for the homophily statistic. In the case when $\alpha = 1$ (or equivalently, $\beta = 1$), this non-matching two-star statistic is simply the two-star statistic, which may lead the model to degeneracy. However, we often find that the best-fitting model uses a discount parameter that is less than 1, in which case we do not encounter the problem stated above.

A problem that arises in some applications when using the dummy statistic is that it may be highly correlated with the edges term (indeed, it is occasionally perfectly correlated for the range
of networks that are likely to occur in a reasonable simulation). In such cases, we may only use one of the two terms, typically the edges term due to its simplicity. We will see an example of this phenomenon in the examples of this section.

The two datasets we consider are the global expansion dataset described in Section ?? and another social network due to Seierstad and Opsahl (2011) that we call the “boards” dataset, in which people are connected to the companies that they serve as members of the board of directors; thus, we may interpret an edge between director $k$ and company $i$ of this network as “individual $k$ is a director of company $i$”. We consider the most recent dataset, dated 2011–08–01, from the many available on the website maintained by Seierstad and Opsahl (2011). This dataset consists of $n_1 = 357$ companies, $n_2 = 1437$ directors, and 1767 directorships. The Gender attribute is present for the directors and this attribute will be used in studying homophily. For the Global Expansion dataset, we consider (differential) homophily for the “firm type” attribute of the mode-2 nodes.

To accomplish these analyses in R, we add the terms $\text{b1nodematch("Gender")}$ and $\text{b2nodematch("firm.type", diff = TRUE)}$, respectively, to the corresponding ERGM formulas. The default value of $\alpha$ (or $\beta$) equals 1, and any other value, say $x$, is coded by inserting $\text{alpha = x}$ or $\text{beta = x}$ as an additional argument inside the $\text{b1nodematch}$ or $\text{b2nodematch}$ parentheses.

Figure 2.4 shows the behavior of the maximized loglikelihoods when these ERGMs are fitted using a sequence of discount parameter (i.e., $\alpha$ or $\beta$) values that range from 0 to 1 for (i) the boards dataset, and (ii) the global expansion dataset. Tables 2.1 and 3.2 provide parameter estimates for the ERGMs fitted for these two datasets at their optimal discount parameters—i.e. the values of the discount parameters that maximize the loglikelihood.

In Figure 2.4, we see that the optimal value of the discount parameter is approximately $\alpha = 0$ for the Boards data, whereas the optimal value for the Global Expansion dataset is approximately $\beta = 0$. The behavior of the loglikelihood as a function of $\alpha$ and $\beta$ is quite different for these two datasets; we discuss these discount parameters in greater detail in Section 2.6.

From Table 2.1, we see that females in this network are less likely than males to serve on the
Boards: Loglikelihood Vs. Discount Factor

Discount Factor
Loglikelihood

Global Expansion: Loglikelihood Vs. Discount Factor

Discount Factor
Loglikelihood

Figure 2.4: Loglikelihood vs. discount parameter for different datasets. For each dataset, the best discount parameter is a $\beta$ value, indicated by a vertical dashed arrow.

Table 2.1: Boards Data: Parameter estimates and standard errors, along with `ergm` code used to obtain them. Statistical significance at the 0.0001 level is indicated by ***.

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Parameter estimate (S.E.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>edges</td>
<td>$-4.5261 (0.1470)$***</td>
</tr>
<tr>
<td>b2factor.Gender.Female</td>
<td>$-0.5038 (0.1113)$***</td>
</tr>
<tr>
<td>b2nodematch.dummy</td>
<td>$0.0302 (0.0446)$</td>
</tr>
<tr>
<td>b2nodematch.Gender</td>
<td>$-0.4605 (0.0661)$***</td>
</tr>
</tbody>
</table>

Code in `ergm`:
```
ergm(boards ~ edges + b2factor("Gender", alpha = 0) + b2nodematch("Dummy", alpha = 0) + b2nodematch("Gender", alpha = 0))
```

boards of the companies in this network (NB: It is important to interpret these results only in the context of the size and makeup of the current network): Being female is associated with a decrease in log-odds of an edge equal to 0.5038, which means that the odds of an edge are smaller by a multiplicative factor of $\exp\{-0.5038\} = 0.604$, holding all other factors equal. Of greater interest from our point of view, the `binodematch` statistic with $\alpha = 0$ counts the number of pairs of nodes of the same gender that are members of the same board of directors. Furthermore, the presence of the corresponding statistic using the dummy nodal covariate (defined to have the same value for every node) means that the model corrects for any baseline tendency for individuals to serve on the same boards—that is, the coefficient of $-0.4605$ isolates the homophily effect, suggesting that
pairs of individuals are less likely to serve on the same board if they happen to be of the same gender.

**Table 2.2:** Global Expansion Data: Parameter estimates and standard errors, along with `ergm` code used to obtain them. Statistical significance at the 0.0001 level is indicated by ***.

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Parameter estimate (S.E.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>edges</td>
<td>4.1738 (0.3728)***</td>
</tr>
<tr>
<td><code>b2nodematch.firm.type.Advertising</code></td>
<td>−7.2070 (0.7458)***</td>
</tr>
<tr>
<td><code>b2nodematch.firm.type.BankingFinance</code></td>
<td>−7.5435 (0.7589)***</td>
</tr>
<tr>
<td><code>b2nodematch.firm.type.Law</code></td>
<td>−9.3817 (0.6532)***</td>
</tr>
</tbody>
</table>

Code in `ergm`:

```
ergm(global ~ edges + b2nodematch ("firm.type", beta = 0, diff = TRUE,
keep = 2:4))
```

Table 3.2 summarizes the results of the model that uses differential “firm type” homophily statistics at the optimal value of $\beta = 0$, which means we look at the matching nodes of each firm type separately. There are actually four values of firm type in the dataset: Accountancy, Advertising, Banking & Finance, and Law. However, as accountancy firms are located in all cities in the network, it is not necessary to include a separate `b2nodematch` term for the accountancy firm type. Eliminating this first statistic, i.e., keeping only statistics 2 through 4, is accomplished using `keep=2:4` as shown in Table 3.2.

With $\beta = 0$, the `b2nodematch` statistic counts one-half the number of edges that are part of at least one two-path joining two firms of the same type. In other words, it is half the number of firm-city combinations with at least one other firm of the same type in the same city. In this dataset, if we include the `b2nodematch` term with a dummy covariate, the corresponding statistic counts one-half the number of edges that are part of at least one two-path joining two firms of any type. But since each city hosts multiple firms in this dataset, all edges are part of such a two-path; furthermore, any networks even vaguely resembling the observed network have this same property. Thus, for this particular dataset, the `edges` term captures nearly exactly the same effect as the `b2nodematch` term with a dummy covariate and $\beta = 0$ would capture, so we omit the latter
term from the model.

The striking feature of the coefficients in Table 3.2 is the strong presence of anti-homophily. Our fitted model predicts, for instance, that the odds of an advertising firm locating in a particular large city are either $e^{4.1738} = 64.9618$—a virtual certainty—if no other advertising firm is already located there, or $\exp\{4.1738 - 7.2070 \times 1/2\} = 1.7688$ is another firm is already located there. By design, our model predicts a different strength of effect for the three types of firms considered.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure2_5.png}
\caption{MCMC diagnostic plots of ERGMs fitted for boards data with $\beta = 0.25$ (left) and for global expansion data with $\beta = 0$ (right).}
\end{figure}

Figure 2.5 shows Markov chain Monte Carlo (MCMC) diagnostic plots obtained for randomly generated networks from the fitted models. The trace plots (left panels) show that the values of the statistics obtained from repeatedly simulating networks using MCMC lack any evident pattern over time, while the histograms of these values (right panels) show roughly normal-shaped distributions centered at the observed values of the corresponding statistics. Though plots such as these do not prove that the model-fitting routine has performed well, they are consistent with good performance and they do not indicate any obvious problems.

The diagnostics of Figure 2.5 are a special case of the goodness-of-fit idea of Hunter et al. (2008a) in which we compare particular observed network statistics with the corresponding statistics that come from a set of networks simulated from the fitted model. These statistics might be among those used in the model, or they might not. If a particular observed statistic seems unrealistic
to obtain via the simulated networks, then we suspect possible model degeneracy if it was a statistic included in the model, or a lack of fit relative to that statistic otherwise. Figure 2.6 compares our observed network to 1000 different networks simulated from the model using our maximum likelihood estimates based on two sets of statistics: The degree distributions of mode-2 nodes (firms) and the so-called dyadwise shared partner (DSP) distribution, in which DSP_k equals the number of all node pairs having exactly k shared partners (i.e., nodes to which they are both connected). The plots indicate some discrepancies, which may not be surprising given the simplicity of the models we used here, but overall the simulated networks appear to be reasonable at capturing certain features of the observed network. One might consider modifications to the model based on the discrepancies in these plots, but this topic is beyond the scope of the current article.

Figure 2.6: Observed statistics (solid line) vs. simulated statistics (boxplots) for the degree distributions of mode-2 (firms) nodes and the dyadwise shared partner statistics for the global expansion dataset.

2.5 Simulations

In this section, we present a set of simulations based on the nodes and covariate values of the global expansion dataset to assess how well the estimation algorithm performs. We consider two models, each using the same statistics as in Table 3.2, that is, the edges statistic along with b2nodematch statistics for advertising, banking & finance, and law. In one case, we fix $\beta = 0.5$, and in the other
Parameter Estimates

<table>
<thead>
<tr>
<th>Statistic</th>
<th>$\beta = 0.5$</th>
<th>$\alpha = 0.5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>edges</td>
<td>1.024</td>
<td>0.326</td>
</tr>
<tr>
<td>b2node.match.firm.type.Advertising</td>
<td>-0.186</td>
<td>0.526</td>
</tr>
<tr>
<td>b2node.match.firm.type.BankingFinance</td>
<td>-0.248</td>
<td>0.192</td>
</tr>
<tr>
<td>b2node.match.firm.type.Law</td>
<td>-1.184</td>
<td>-0.907</td>
</tr>
</tbody>
</table>

Table 2.3: Parameter estimates used in simulation study.

In each case, we fix $\alpha = 0.5$. In each case, we estimate parameters for these models using the observed network, resulting in the values shown in Table 2.3. We then use these values to simulate ten random networks, on the same set of nodes with the same nodal covariate values, from each fitted model. Finally, we treat each of the twenty resulting networks as in Section 2.4, varying the two discount parameters to see which gives the best maximized log likelihood. Ideally, we will be able to recover approximately the true discount parameter in all cases.

Figure 2.7: Loglikelihood vs. discount parameter for 10 simulated networks based on a true model with $\beta = 0.5$ and $\alpha = 0.5$ discount parameters respectively.

For the series of ERGMs fitted for each of the ten simulated networks by varying the discount parameters in the $\beta = 0.5$ case, the model that maximized the loglikelihood was observed to be the one having a $\beta$ discount parameter that is approximately equal to 0.5, fulfilling our expectations. Similarly, in the $\alpha = 0.5$ case, the model with a discount parameter $\alpha = 0.5$ appeared as the one that maximizes the loglikelihood. The results are depicted in Figure 2.7. These results are
reassuring since they suggest that we may recover the correct discount parameter when the true model is known.

2.6 Discussion

The bipartite homophily statistics introduced in this article address the degeneracy that may occur due to direct counting of two-stars with matching endpoints in an exponential-family random graph model, or ERGM. We introduce two types of attenuation parameter, one based on a nodepair-centered view of homophily and the other based on an edge-centered view, that prevent the two-star counts from becoming too highly inflated by the addition of a single edge to the network. The strength of this attenuation effect is controlled by an exponent, or discount parameter, which we denote by $\alpha$ in the nodepair-centered case and $\beta$ in the edge-centered case: The attenuation is strongest when the discount parameters are close to zero, and it disappears when the parameters equal one. The new statistics are available in the \texttt{ergm} package for \texttt{R}, and they may be used in conjunction with a wide range of other statistics. The specific statistics included in an ERGM will depend on the modeling aims particular to the situation and, as always, the interpretation of coefficients from a fitted model must be done in the presence of all other model terms. For more on the types of statistics that might be considered, we recommend that interested readers consult survey papers available about ERGMs, such as the general treatment by Robins \textit{et al.} (2007a) or the bipartite-specific articles by Wang \textit{et al.} (2009) and Wang \textit{et al.} (2013).

We demonstrate that if $\alpha$ and $\beta$ are not fixed, the new statistics may be viewed as belonging to a curved exponential family model. However, as a practical matter, we demonstrate that evaluation of the approximate log-likelihood for several fixed values of $\alpha$ or $\beta$ in a grid in the range from 0 to 1 should suffice to choose a near-optimal value of the discount factor. Although existing curved exponential family models elsewhere in the network modeling literature (e.g., Hunter and Handcock, 2006; Robins \textit{et al.}, 2007b; Hunter, 2007b) do not take homophily into account, in Section 2.3 we point out some limited analogies between our statistics and other well-known curved...
exponential family statistics.

A natural question is when it is more appropriate to use the nodepair-centered ($\alpha$) model than the edge-centered ($\beta$) model. Although we have selected one example of each for Section 2.4, our preliminary investigations on other datasets suggest that the $\beta$ exponent often leads to higher likelihoods than the $\alpha$ exponent. Why this is true is not immediately clear, but this fact may point to a broader question about the true sociological nature of homophily in bipartite network contexts.

For example, when a company $i$ wishes to enter a new market $k$, the existing competition in that market plays a major role in the decision of entering or not entering the market. Comparatively, the competitors of $i$ having branches in other cities where $i$ also has branches wouldn’t matter so much in the decision of $i$ entering to city $k$. This might be the reason why we see models with edge-based homophily statistics which look at the competition within a specific market lead to better fitting models than the nodepair-based models in the case of the global expansion dataset. If we look at the background of boards dataset, we see that the directors of companies are appointed by invitations and often different companies prefer to have board members in common with other companies so that all involved companies fall under the the same control. This will enhance the power of companies as a whole or a group and the companies can work collectively towards shared goals. That means the board memberships may be highly influenced by the shared multiple board memberships of directors, which will be captured by the proposed nodepair-based homophily statistics. These are some intuitive ideas as to why a $\beta$ discount factor may be preferred in one situation and an $\alpha$ discount factor may be preferred in another.

Although we do not demonstrate it in this article, the \texttt{b1nodematch} (or \texttt{b2nodematch}) terms have the flexibility to allow for separate homophily effects for each level of a particular mode-2 (or mode-1) attribute. For example, in the global expansion dataset, if we know to which geographical region each city belongs, then we may study the homophily effect separately for each region by using, say, \texttt{b2nodematch("firm.type", beta=0, byb1attr="region"). This \texttt{byb1attr} argument may be used with or without the \texttt{diff = TRUE} option, which decomposes each b2nodematch
statistic into its levels as illustrated in Table 3.2.

We close by making explicit an implicit assumption used throughout this article, namely, that the nodal covariates defining homophily are fixed and observed for every node. In the case where we allow some of the nodal covariates to be random, however, even the unipartite case is not simple. To handle this case in the context of ERGMs, Fellows and Handcock (2012) propose a new concept and related statistic, called “regularized homophily.” It may be possible to extend regularized homophily to the case of bipartite networks, as we have extended standard homophily here; however, this topic is beyond the scope of the present article.

2.7 Acknowledgments

This work is supported by NIH grant R01-GM083603-01. The global firms data were created by P.J. Taylor and D.R.F. Walker as part of their project “World City Network: Data Matrix Construction and Analysis” and are based on primary data collected by J.V. Beaverstock, R.G. Smith and P.J. Taylor (ESRC project “The Geographical Scope of London as a World City” (R000222050)). They constitute Data Set 6 of the Globalization and World Cities (GaWC) Research Network (http://www.lboro.ac.uk/gawc/) publication of inter-city data.

2.8 Appendix: Change Statistics

As discussed in Hunter et al. (2008b), for a particular choice \( s(y) \) of network statistics in an ERGM and a particular pair \((i, j)\) of nodes, we may define the vector of change statistics as

\[
\delta_s(y)_{ij} = s(y_{ij}^+) - s(y_{ij}^-),
\]
where \( y^+_{ij} \) and \( y^-_{ij} \) represent the networks obtained by fixing \( y_{ij} = 1 \) or \( y_{ij} = 0 \), respectively, and keeping all other entries the same as in \( y \) itself. These change statistics are important because the conditional log-odds that \( Y_{ij} = 1 \), conditional on all other entries in \( Y \) (denoted by \( Y^c_{ij} \)), may be expressed very simply by the expression

\[
\log \frac{P_\theta(Y_{ij} = 1 \mid Y^c_{ij} = y^c_{ij})}{P_\theta(Y_{ij} = 0 \mid Y^c_{ij} = y^c_{ij})} = \theta^T \delta_s(y)_{ij}.
\] (2.11)

Equation (2.11), which arises directly from Equation (2.1) via simple algebra, does not involve the troublesome factor \( \kappa(\theta) \), which makes it particularly simple for, say, simulating random networks whose distribution is approximately \( P_\theta(\cdot) \) via MCMC. Simulation of this nature is discussed in Section 6 of Hunter et al. (2008b).

In addition, examination of change statistics may sometimes identify possible degeneracy problems with an ERGM: If a change statistic can grow exceedingly large in the positive or negative direction, then a corresponding \( \theta \) coefficient of the same or opposite sign can virtually assure the presence or absence of an edge, and this type of behavior can become self-reinforcing. We do not delve into details of the degeneracy issue here, referring interested readers instead to Handcock (2003) and Schweinberger (2011).

This section presents change statistics for both the nodepair-centered (\( \alpha \)-based) and edge-centered (\( \beta \)-based) homophily statistics introduced in this article. First, fix a node pair \((i,k)\), where \( i \) belongs to mode 1 and \( k \) belong to mode 2; we will focus on the \texttt{b1nodematch} change statistics, though the arguments below may easily be adapted to the \texttt{b2nodematch} case.

The corresponding change statistics are obtained by calculating the difference between the contributions to the statistic with \( y_{ik} = 1 \) and \( y_{ik} = 0 \). For the nodepair (\( \alpha \)-based) version of the homophily statistic, Equation (2.6) leads to

\[
\delta^{[\alpha]}(y)_{ik} = \frac{1}{2} \sum_{j \neq i} I\{i \text{ matches } j\} \left[ \sum_{k' \neq k} y_{ik'}y_{jk'} + y_{jk} \right] \alpha
\]
\[-\frac{1}{2} \sum_{j \neq i} I\{i \text{ matches } j\} \left[ \sum_{k' \neq k} y_{ik'} y_{jk'} \right]^\alpha,\]

where the outer sums are taken over mode-1 nodes \( j \neq i \) and the inner sums are taken over mode-2 nodes \( k' \neq k \). This expression may be rewritten as

\[\delta^{[\alpha]}(y)_{ik} = \frac{1}{2} \sum_{j \neq i} I\{i \text{ matches } j\} \left[ (m(i, j, k; y) + y_{jk})^\alpha - m(i, j, k; y)^\alpha \right], \tag{2.12}\]

where \( m(i, j, k; y) \) is the number of two-paths from \( i \) to \( j \) not passing via \( k \). Similarly, for the edge (\( \beta \)-based) version of the homophily statistic, Equation (2.7) leads to

\[\delta^{[\beta]}(y)_{ik} = \frac{1}{2} \left[ \sum_{j \neq i} I\{i \text{ matches } j\} y_{jk} \right]^\beta. \tag{2.13}\]

Considering the extreme case in which \( \alpha \) or \( \beta \) equals 0, we see from Equation (2.13) that when \( \beta = 0 \), the change statistic cannot be larger than \( 1/2 \), which means that degeneracy behavior is unlikely. When \( \alpha = 0 \), on the other hand, it takes a bit of thought to see that the quantity in square brackets in Equation (2.12) is zero unless both \( m(i, j, k; y) = 0 \) and \( y_{jk} = 1 \). Thus, the value of \( \delta^{[\alpha]}(y)_{ik} \) could theoretically be as large as one half the number of mode-1 nodes minus one, yet such a large value would not occur repeatedly as the number of edges increases toward a complete network or decreases toward an empty network. Thus, we do not expect degeneracy behavior for the nodepair-based statistic either.
Bayesian Inference for Curved ERGMs

3.1 Introduction

The estimation of parameters of a curved ERGM, such as the $\alpha$ or $\beta$ discount parameters we discussed in Chapter 2, via an MLE-based approach is very complicated and the necessary tools may not always be available in `ergm`. In this chapter, we take a Bayesian approach to estimation, which enables us to simultaneously estimate all the parameters including the curved/discount parameters. Bayesian inference tools for non-curved ERGMs are already available in the form of an R package called `Bergm` (Caimo and Friel, 2011). However, when the network statistics depend on unknown parameters like $\alpha$ (or $\beta$), we can no longer use `Bergm` tools to estimate these parameters unless we assume their values are fixed. We propose a modification to the algorithm in `Bergm` that can accommodate scenarios with non-fixed parameters such as the ones mentioned above.
3.2 The General Model

The general model that we consider throughout this chapter takes the form

\[
P(Y = y) = \frac{\exp\{\sum_{i=1}^{k} \eta_i s_i(y) + \sum_{j=k+1}^{p} \eta_j \gamma_{j-k}(y)\}}{\kappa(\eta)},
\]

(3.1)

where \( p \) is the total number of network statistics that are used in the model, of which \( k \) do not involve any curved/discount parameters and \( (p - k) \) do. Also, we take \( m = p - k \) and use the notation \( \eta = (\eta_1, \eta_2, \ldots, \eta_p) \) and \( \gamma = (\gamma_1, \ldots, \gamma_m) \) throughout this chapter. Therefore, the curved/discount parameters are the \( \gamma \) parameters.

3.3 Bayesian Inference

The usual approach for conducting Bayesian inference for ERGM parameters is to use a Metropolis-Hastings (M-H) algorithm which draws samples directly from the approximate posterior distribution of interest. That is, if we let \( \theta = (\eta, \gamma) \), then the M-H algorithm constructs a Markov chain whose stationary distribution is \( \pi(\theta|y) \). But in our scenario, the acceptance ratio for this algorithm requires the calculation of a parameter-dependent normalizing constant \( \kappa(\theta) \), which is very hard to compute. This is a situation where we have a doubly-intractable distribution where the posterior distribution involves two normalizations, one with regard to the marginal distribution \( \pi(y) \) and the other with regard to \( \pi(y|\theta) \) which involves a parameter dependent normalizing constant. Møller et al. (2006) and Murray et al. (2006) describe the exchange algorithm, which allows us to get away from computing these intractable normalizing constants.
3.3.1 Exchange Algorithm

The idea behind the exchange algorithm is to get rid of the intractable normalizing constants from the acceptance ratios of M-H updates. This is achieved by sampling from an augmented distribution $\pi(x, \theta | y)$ instead of $\pi(\theta | y)$, the original posterior distribution of interest, where the auxiliary variable $x$ is defined on the state space of $y$ (Møller et al., 2006). This augmented distribution satisfies

$$\pi(x, \theta | y) \propto \pi(\theta, x, y) = \pi(x | \theta, y) \pi(y | \theta) \pi(\theta),$$

(3.2)

where $\pi(x | \theta, y)$ is assumed not to depend on $y$ and to coincide with the distribution $\pi(y | \theta)$. When running the MCMC algorithm, we first draw $\theta'$ from $\epsilon(\theta' | \theta)$, a proposal distribution for $\theta'$ given the current state $\theta$, which is often considered to be a symmetric distribution centered at the current value in order to make the calculations easier, and then draw $x$ from $\pi(x | \theta')$, which is the same distribution on which data $y$ is defined. Then, we update $\theta \rightarrow \theta'$ or retain $\theta$ based on the Hastings-ratio as in step four shown below. Even though we wish to estimate $\eta$ and $\gamma$ simultaneously, these two types of parameters do not necessarily have the same support. In other words, we need to be careful when choosing proposal distributions for these two types of parameters. To resolve these issues easily, one may draw $\eta$ parameters from one proposal distribution (generally, from a multivariate normal distribution) and draw $\gamma$ parameters from a different proposal distribution independently from the $\eta$ parameters. However, this simplest approach may result in poor mixing in the Markov Chains.

The steps of the exchange algorithm for the scenario in which $\eta$ and $\gamma$ are drawn independently of each other can be summarized as follows.

For $i = 1, \ldots, N$

1. generate $\eta'$ from some proposal distribution $\epsilon(., \eta) \equiv N(\eta, \Sigma)$

2. generate $\gamma'$ from some proposal distribution $h(., \gamma)$

3. simulate $x$ from $\pi(., | \eta', \gamma')$ by MCMC methods as described in Section 1.2.1.2.
4. update $\theta \equiv (\eta, \gamma) \rightarrow \theta' \equiv (\eta', \gamma')$ with probability $p$ where

$$
\log p = \min\{0, \eta^T [S_\gamma(x) - S_\gamma(y)] - \eta'^T [S_{\gamma'}(x) - S_{\gamma'}(y)] + \log\left(\frac{\pi(\theta')}{\pi(\theta)}\right) + \log\left(\frac{h(\gamma|\gamma')}{h(\gamma'|\gamma)}\right)\},
$$

and otherwise keep $\theta$ unchanged.

3.3.2 Parallel Adaptive Direction Sampler

The Parallel Adaptive Direction Sampler (parallel ADS) is a special case of the General Adaptive Direction Sampler, a technique that is used to obtain samples from all-over the support of the target density more efficiently, by automatically adjusting the sampling direction in each iteration (Gilks et al., 1994). Parallel ADS involves multiple chains that interact with each other. The next state (say $\theta_{h+1}$) of chain $h$ depends on its own current state $\theta_h$ and the current states (Say $\theta_{h_1}$ and $\theta_{h_2}$) of two randomly selected chains from the rest of the chains. The term parallel comes into play as the next state of chain $h$ is obtained by adding a small amount of noise to a point on a line that passes through $\theta_h$ in the direction that is parallel to the line that goes through $\theta_{h_1}$ and $\theta_{h_2}$. This idea is illustrated in Figure (3.1).
3.3.3 Exchange Algorithm + Parallel Adaptive Direction Sampler

In order to improve mixing, we take a similar approach to Caimo and Friel (2013), that is, using a combination of the modified exchange algorithm in Section (3.3.1) and the parallel adaptive direction sampler described in Section (3.3.2). The steps given below are for the case where $\eta$ and $\gamma$ are drawn from separate proposals.

for $i = 1, \ldots, N$

for $h = 1, \ldots, H$

1. generate $h_1$ and $h_2$ such that $h_1 \neq h_2 \neq h$

2. generate $\eta'_h$ from $\rho(\eta_{h_1} - \eta_{h_2}) + \epsilon_{|\eta_h}$

3. generate $\gamma'_h$ from $\rho(\gamma_{h_1} - \gamma_{h_2}) + h_{|\gamma_h}$

4. simulate $y'$ from $\pi(., |\eta'_h, \gamma'_h) \propto \exp\{\eta'_h^T s_{\gamma'_h}(x, y)\}$

5. update $\theta_h \rightarrow \theta'_h$ with probability $p$ where

Figure 3.1: Parallel Adaptive Directional Update (Example): The green circles denote the current values of the six chains and the red star denotes the proposed value for the next state of chain $h$.
\[ \log p = \min\{0, \eta_{\text{h}}^T [S_{\gamma_{\text{h}}}(y') - S_{\gamma_{\text{h}}}(y)] - \eta_{\text{h}}^T [S_{\gamma'_{\text{h}}}(y') - S_{\gamma'_{\text{h}}}(y)] + \log \left( \frac{\pi(\eta_{\text{h}})}{\pi(\eta_{\text{h}}')} \right) + \log \left( \frac{h(\gamma_{\text{h}}|\gamma'_{\text{h}})}{h(\gamma_{\text{h}}'|\gamma_{\text{h}})} \right) \} \]

and otherwise keep \( \theta_{\text{h}} \) unchanged.

### 3.3.4 Proposal Distributions

An important consideration in applying the above algorithm to our problems is selecting an appropriate proposal distribution for the curved and non-curved parameters. The support of curved/discount parameters is often different from the non-curved parameters. As discussed earlier, one way in which we can tackle this issue is to use completely independent proposals for curved and non-curved parameters. Poor mixing and slow convergence of the Markov chains are the main issues associated with this method. Another way to approach this issue is to generate proposals for both curved and non-curved parameters using a single multivariate normal distribution and transform them appropriately so that all of them fall within their respective supports. This will add an additional step to the process, yet will significantly improve mixing.

#### 3.3.4.1 Separate Proposals for Curved & Non-Curved Parameters

Consider an ERGM with an edges term and a `b1nodematch` term with a \( \beta \) discount parameter introduced in Chapter 2. Thus, we have two non-curved parameters (Say \( \eta_1 \) and \( \eta_2 \)) and one curved parameter (\( \beta \)) to be estimated. The parameter space for both \( \eta_1 \) and \( \eta_2 \) is \((-\infty, +\infty)\) while that of \( \beta \) is \([0, 1]\). To deal with the discrepancies between the parameter space of \( \eta \) and \( \beta \), one can simply use a proposal \( h(\eta, \beta|\eta_{\text{curr}}, \beta_{\text{curr}}) \) such that

\[
h(\eta, \beta|\eta_{\text{curr}}, \beta_{\text{curr}}) = h_1(\eta|\eta_{\text{curr}})h_2(\beta|\beta_{\text{curr}})
\]

where \( h_1 \) and \( h_2 \) propose values from the support of \( \eta \) and \( \beta \), respectively. We illustrate a similar scenario via a real-world dataset in Section 3.4, where \( h_1(\eta) \) is a multivariate normal distribution and \( h_2(\beta) \) is a beta distribution or a mixture distribution such as a Beta\((a, b)\) distribution with point masses at 0 and 1. One may either let (i) \( h_2(\beta|\beta_{\text{curr}}) = h_2(\beta) \) (proposed value is independent...
of the current value) or, (ii) let the parameters of the proposal distribution \( h_2(\beta|\beta_{\text{curr}}) \) depend on the current value in some manner. For example, if we use a Beta\((a, b)\) distribution as the proposal, then one may use \( \beta_{\text{curr}} \) as its mean, i.e., \( a/(a + b) = \beta_{\text{curr}} \) where we fix \( a + b = k \), a constant so that the variance equals \( \beta_{\text{curr}}(1 - \beta_{\text{curr}})/(k + 1) \). This will result in \( a = k\beta_{\text{curr}} \) and \( b = k(1 - \beta_{\text{curr}}) \), hence the Beta\((a, b)\) proposal distribution is dependent on the current state \( \beta_{\text{curr}} \) of \( \beta \). To avoid problems with \( \beta_{\text{curr}} \) becoming too close to 0 or 1, we set \( a = 0.05k \) and \( b = 0.95k \) if \( \beta_{\text{curr}} < 0.05 \) and set \( a = 0.95k \) and \( b = 0.05k \) if \( \beta_{\text{curr}} > 0.95k \). Different choices for \( k \) as well as the cutoffs 0.05\( k \) and 0.95\( k \) yield algorithms with different mixing characteristics. In Section 3.4, the algorithm in which the proposal distribution for \( \beta \) is a mixture of a beta distribution and a point mass at 0 that does not depend on the current value of \( \beta \) is denoted by EX-ADS-MIX while the algorithm with a current-state-dependent beta distribution as the proposal is denoted by EX-ADS-BETA.

### 3.3.4.2 A Multivariate Normal Proposal for Curved and Non-Curved Parameters

If we can generate proposals for all parameters from one known multi-dimensional distribution as in Caimo and Friel (2011), we can simplify the modifications required for the algorithm in Section 3.3.3 to accommodate the curved/discount parameters. We do this as follows. First, we generate a proposal for both curved and non-curved parameters from a single multivariate normal distribution. Next, via the inverse of a bijective function that converts the support from its original space to \((-\infty, +\infty)\), we find a valid proposal for the curved parameters. Then, we use these proposals to simulate a network and also to calculate the acceptance probability. For example, in the case where we aim to estimate the \( \beta \) discount parameter (discussed in Chapter 2) along with other parameters, we can use the inverse logit transformation to convert the proposal back to the appropriate parameter space \([0, 1]\) of \( \beta \). The algorithm that uses a multivariate normal proposal for both curved parameters (or discount parameters) and non-curved parameters is denoted by EX-ADS-NORM in Section 3.4.

In this case, the algorithm can be modified as follows.
Let \( g(\beta) = \gamma \)

Let \( \theta = (\eta, \gamma)^T \)

for \( i = 1, \ldots, N \)

for \( h = 1, \ldots, H \)

1. generate \( h_1 \) and \( h_2 \) such that \( h_1 \neq h_2 \neq h \) by sampling two numbers from 
   \( \{1, \ldots, h-1, h+1, \ldots, H\} \) without replacement

2. generate \( \theta'_h \) from \( \rho(\theta_{h1} - \theta_{h2}) + \epsilon(.|\theta_h) \)

3. set \( \beta'_h = g^{-1}(\gamma'_h) \)

4. simulate \( y' \) from \( \pi(.|\eta'_h, \beta'_h) \propto \exp\{\eta'_h^T S_{\beta'_h}(x, y)\} \)

5. update \( \theta_h \rightarrow \theta'_h \) with probability \( p \) where

\[
\log p = \min\{0, \eta_h^T [S_{\beta_h}(y') - S_{\beta_h}(y)] - \eta'_h^T [S_{\beta'_h}(y') - S_{\beta'_h}(y)] + \log(\frac{\pi(\theta'_h)}{\pi(\theta_h)})\} \text{ and otherwise keep } \theta \text{ unchanged.}
\]

### 3.4 Application and Results

In this section, we test our algorithms with real data using the global expansion dataset we used in Chapter 2. Recall that this is a bipartite network with 55 cities around the world and 46 global firms (Taylor and Walker, 2001), in which the ties denote the presence or absence of a given firm in a given city. We fit an ERGM to these data with an edges term and homophily terms based on firm type, which is one of Advertising, Banking Finance or Law.

The study was conducted in three main ways. First, the \( \eta \) parameters were estimated via MCMC-MLE, while fixing the value of \( \beta \) to one in a sequence of values that range from 0 to 1 with
0.05 increments. The best model (the one which had the highest loglikelihood/AIC) appeared to be the one with $\beta = 0$. Secondly, the existing R package Bergm was used to fit the $\eta$ parameters via Bayesian method by fixing $\beta$ to its best value 0 obtained from the MCMC-MLE approach. Bergm is a tool for running Bayesian estimation for an ERGM when no unknown parameters such as $\alpha$ (or $\beta$) exist in the vector of network statistics. Bergm estimates for the $\beta = 0$ case were obtained to assess how MLE and Bayesian estimations may tally or differ when $\beta$ is considered to be fixed at the same value. Thirdly, Bayesian inference for both $\eta$ and $\beta$ was carried out via the two algorithms EX-ADS-MIX and EX-ADS-BETA in Section (3.3.4.1) and the EX-ADS-NORM algorithm in Section 3.3.4.2. The inputs used for these algorithms are shown in Section 3.4.1. The coefficients obtained from each algorithm are summarized in Table 1 and the results obtained from the MCMC-MLE approach and the original Bergm tools by fixing $\beta$ discount factor to 0 are summarized in Table 3.2, while the overall acceptance rates and time taken for the latter three algorithms are summarized in Table 3.1. In terms of the accuracy of the estimates and the mixing of the Markov chains, the algorithm EX-ADS-NORM appears to be superior to EX-ADS-MIX and EX-ADS-BETA.

### 3.4.1 Settings

The following prior proposal distributions and other tuning parameters were used in the application described above.

1. $\epsilon(,|\eta_h) \equiv N(\eta_h, \Sigma)$, where $\Sigma = \text{Diag}(0.0025)$

2. Proposal distributions used for $\beta$:

   - **EX-ADS-MIX**: 
     
     $h(\beta'_h|\beta_h) \equiv h(\beta) = w \delta(\beta) + (1-w)\text{Beta}(1, 20)$, where $\delta(\beta)$ is a point mass at $\beta = 0$ and $w = 0.3$
• **EX-ADS-BETA:**
  \[ \text{Beta}(a, b) \text{ with } a = 4\beta_{\text{curr}}, b = 4 \times (1 - \beta_{\text{curr}}) \]

• **EX-ADS-NORM:**
  \[ g(\beta) = \gamma \text{ was proposed along with } \eta \text{ via a single multivariate normal proposal, } \]
  \[ \epsilon(.|\theta_h) \equiv N(\theta_h, \Sigma), \text{ where } \Sigma = \text{Diag}(0.0025) \text{ and } \theta = (\eta, \gamma)^T \]

3. \[ \pi(\theta) \equiv \pi(\eta)\pi(\beta) \]

4. \[ \pi(\beta) \equiv \text{Beta}(1, 10) \]

5. \[ \pi(\eta) \equiv N(0, \Sigma^*) \text{, where } \Sigma^* = \text{Diag}(400) \]

6. \[ N = 10000 \text{ (main iterations)}, H = 10 \text{ (Number of parallel Markov chains)}, \rho = 0.5, \]
  \[ \text{burnin} = 500, \text{ thinning} = 1, \text{ auxiliary iterations} = 20000 \]

### 3.4.2 R Code

All these algorithms are implemented in R and part of the code—the R code for EX-ADS-NORM—is provided in the Appendix (A.1). The R commands used to run the EX-ADS-NORM algorithm by calling the `bcergm` function in Appendix (A.1) for the above scenario are given below.

```r
#------------------------------------------------------------------------
# FUNCTION TO GET THE MODEL FORMULA
#------------------------------------------------------------------------

getModelFormula <- function(curved.par, net.name) {
  out <- paste(net.name, " ~ edges + b2nodematch('firm.type',
            diff = TRUE, keep = 2:4, beta = ", curved.par, ")", sep = ""
  out <- formula(out)
  return(out)
}
```
```r
#
# INPUT PARAMETERS
#
set.seed(12345678)
load("GlobalExpansion/globalExpansion.RData")

net <- globalExpansion
nstats <- 4
nfactors <- 1
npars <- nstats + nfactors
nchains <- npars * 2
func.list <- list(inv.logit = function(x) {exp(x)/(1 + exp(x))})
rho <- 0.5
N <- 10000
burnin <- 500

# FITTING THE MODEL
#
fit = bcergm(y = net, getModelFormula, N, nstats, nfactors, nchains, rho,
func.list,
burnin = burnin,
mu.prior = NULL,
sigma.prior = diag(400, npars),
sigma.proposal = diag(0.0025, npars),
aux.iters = 20000,
thinning = 1)
```
3.4.3 Efficiency (Acceptance Rates, No: of Iterations, Time Taken)

In this section, we try to compare and contrast the performances of the three algorithms, EX-ADS-MIX, EX-ADS-BETA and EX-ADS-NORM. In terms of the time taken for an equal number of main iterations, we see from Table 3.1 that the algorithm EX-ADS-MIX runs a little faster than the other two algorithms, while estimating all the parameters satisfactorily (See Table 1 in Appendix A.2)—in comparison to the results obtained via MCMC-MLE (See Table 3.2). However, in terms of the mixing of the Markov chains, EX-ADS-NORM appears to be the best one. In most examples, 10000 main iterations suffices to obtain an appropriate set of estimates and in that case, the gain by EX-ADS-MIX in terms of time is not as appealing as the gain by EX-ADS-NORM in terms of the mixing of Markov Chains.

Table 3.1: Global Expansion Data: Acceptance Rates and Time Taken for the algorithms EX-ADS-MIX (Mixture Proposal), EX-ADS-BETA (Dependent Beta Proposal) and EX-ADS-NORM (Multivariate Normal Proposal) each with 10000 main iterations, 20000 auxiliary iterations, and 500 burnin iterations.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Time Taken (in Mins)</th>
<th>Acceptance Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>EX-ADS-MIX</td>
<td>349.7</td>
<td>13.9%</td>
</tr>
<tr>
<td>EX-ADS-BETA</td>
<td>355.6</td>
<td>12.7%</td>
</tr>
<tr>
<td>EX-ADS-NORM</td>
<td>360.4</td>
<td>19.3%</td>
</tr>
</tbody>
</table>

Table 3.2 shows that irrespective of the method that we choose, the most appropriate value for the discount factor $\beta$ seems to be zero (or very close to zero). Also, we notice that all three homophily terms have negative coefficients while the edges term is always positive. The MLE-based results and the full Bayesian (i.e. Bayesian inference for both $\eta$ and $\beta$) results and the original Bergm results which are shown in column 2 agree to a great extent. $\beta$ being 0 and the homophily terms being all negative mean that if at least one firm of a particular type exists in a given city, then it is less likely that another firm of the same type has a branch in that city, which means there is an anti-homophily effect due to the firm type.

Figure 3.2 shows that we have a pretty tight posterior density around zero for parameter $\beta$ and
**Table 3.2:** Global Expansion Data: Parameter estimates; MCMC-MLE (with corresponding standard error) and posterior means (with corresponding standard deviations) for MLE-based and Bayesian methods, respectively.

<table>
<thead>
<tr>
<th>Statistic/Discount Factor</th>
<th>MCMC-MLE</th>
<th>Bayesian (fixed $\beta$)</th>
<th>EX-ADS-NORM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Est. (S.E.)</td>
<td>Est. (Std. Dev.)</td>
<td>Est. (Std. Dev.)</td>
</tr>
<tr>
<td>edges</td>
<td>4.1738 (0.3728)</td>
<td>3.8793 (0.3379)</td>
<td>4.1378 (0.5726)</td>
</tr>
<tr>
<td>b2nodematch.Advertising($\beta$)</td>
<td>-7.2070 (0.7458)</td>
<td>-6.5286 (0.6611)</td>
<td>-7.0838 (1.1245)</td>
</tr>
<tr>
<td>b2nodematch.BankingFinance($\beta$)</td>
<td>-7.5435 (0.7589)</td>
<td>-6.8949 (0.6713)</td>
<td>-7.4372 (1.1310)</td>
</tr>
<tr>
<td>b2nodematch.Law($\beta$)</td>
<td>-9.3817 (0.6532)</td>
<td>-8.8050 (0.6090)</td>
<td>-9.2952 (1.0180)</td>
</tr>
<tr>
<td>$\beta$ (was fixed at 0)</td>
<td>(was fixed at 0)</td>
<td>0.0003 (0.0019)</td>
<td></td>
</tr>
</tbody>
</table>

The autocorrelations between samples disappear very fast. Similar diagnostics for the other four parameters were obtained and are shown in Figure 3.3.

**Figure 3.2:** MCMC diagnostic plots for a $\beta$-based ERGM fitted for global expansion data (Based on EX-ADS-NORM algorithm). $\theta_5$ stands for the fifth parameter in the model, $\beta$. Left most plot is the posterior density of $\beta$ followed by the time-series plot of the estimates and the right most plot is the autocorrelation plot of $\beta$ samples. The results are based on 10000 main iterations and 10 parallel Markov chains with thinning = 10.

### 3.4.4 Simulation Study

In Section 3.4, the true values of the parameters are unknown. Hence, we cannot guarantee that we recovered the true parameters via any of the above methods used. Thus, we conduct a simple
simulation study to see how well we recover the true parameters.

The simulation study was conducted as follows. First, an MLE-based model was fitted to global expansion data with $\beta = 0$ as before. Secondly, five networks were simulated from this fitted model. Then, we estimated all the parameters using the three algorithms described in Section 3.3.3. As we simulated networks from the fitted model, we know the true values of the parameters, which therefore allows us to compare the estimation with the truth. The results shown in Table 3.3 are for the EX-ADS-NORM and the other two algorithms have similar results.

The results in Table 3.3 reveal that the modified version of the combination of exchange algo-
rithm and the parallel adaptive sample we used in this study nicely recover the true parameters.

Table 3.3: Global Expansion Data: Posterior means along with 2.5% and 97.5% quantiles of the posterior for five networks simulated from the model \( \text{network} \sim \text{edges} + b2\text{nodematch}(\text{‘firm.type’}, \text{diff = TRUE, keep = 2:4, } \beta = 0) \) (Based on the results from EX-ADS-NORM algorithm). First row shows the true values of the parameters.

<table>
<thead>
<tr>
<th>Net. ID</th>
<th>Post. Mean</th>
<th>Post. Mean</th>
<th>Post. Mean</th>
<th>Post. Mean</th>
<th>Post. Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.2 (3.1, 5.7)</td>
<td>-7.6 (-10.4, -5.2)</td>
<td>-7.6 (-10.5, -5.2)</td>
<td>-9.4 (-12.1, -7.3)</td>
<td>0.0004 (4e-23, 0.005)</td>
</tr>
<tr>
<td>2</td>
<td>4.1 (3.0, 5.4)</td>
<td>-7.2 (-9.9, -5.0)</td>
<td>-7.3 (-9.9, -5.0)</td>
<td>-9.2 (-11.6, -7.1)</td>
<td>0.0004 (3e-23, 0.004)</td>
</tr>
<tr>
<td>3</td>
<td>4.4 (3.1, 5.9)</td>
<td>-7.8 (-10.7, -5.3)</td>
<td>-7.9 (-10.9, -5.4)</td>
<td>-9.7 (-12.4, -7.5)</td>
<td>0.0003 (2e-24, 0.004)</td>
</tr>
<tr>
<td>4</td>
<td>4.0 (2.9, 5.4)</td>
<td>-7.1 (-9.8, -4.8)</td>
<td>-7.2 (-9.9, -4.9)</td>
<td>-9.3 (-11.6, -7.3)</td>
<td>0.0005 (2e-23, 0.004)</td>
</tr>
<tr>
<td>5</td>
<td>3.9 (2.8, 5.1)</td>
<td>-6.8 (-9.3, -4.7)</td>
<td>-6.9 (-9.4, -4.7)</td>
<td>-9.0 (-11.1, -7.0)</td>
<td>0.0004 (9e-24, 0.004)</td>
</tr>
</tbody>
</table>

3.5 Conclusions

In this chapter, we successfully attempted Bayesian inference on ERGMs with additional parameters that appear in the network statistics. The results of the simulation study in Section 3.4.4 provide us a greater level of confidence on the recovery of true parameters by the algorithms implemented. The ability to conduct Bayesian inference for usual ERGM parameters already existed via the \textbf{Bergm} package in \texttt{R}. However, additional parameters such as \( \beta \) that are part of some in network statistics themselves make it impossible to use \textbf{Bergm} as it is, as there is no way to update \( \beta \) and evaluate the network statistics with the updated \( \beta \) at each iteration. This extension was the main purpose of this study, which was achieved with promising results.

Given that we need to estimate the \( \beta \) parameter along with other usual parameters \( \eta \), the algorithm we implemented produces stable results in a few hours (e.g., for 10000 main iterations, 10 parallel Markov chains with 20000 nested iterations and 500 burnin took around 6 hours). If we know an appropriate value to be used for the discount parameter(s) in advance, the MCMC-MLE approach can estimate the other parameters quickly (around two minutes). However, we do not
have an intuitive way of deciding the value of the discount factors ($\alpha$ or $\beta$) in advance. Using 
an MLE-based trial and error method requires a series of fixed $\beta$ values, which may take a very 
long time to run, depending on how many values are used. Therefore, even though the exchange 
algorithm-based Bayesian estimation may take more time per iteration when compared with the 
MCMC-MLE for fixed $\beta$, it might still be worthwhile to conduct the former estimation method to 
estimate $\beta$. If one has more than one discount parameter, using a trial and error method to find 
the MCMC-MLE will become almost impossible, whereas the EX-ADS-NORM algorithm coded 
in Appendix (A.1) is flexible enough to easily estimate all the coefficients in one go. An example 
of how to prepare the inputs when you have more than one discount parameter to estimate is also 
shown in Appendix (A.1). Also, one may use the EX-ADS-NORM algorithm directly to estimate 
both the type of the discount parameter ($\alpha$ or $\beta$) and its value. This makes the EX-ADS-NORM 
algorithm more appealing relative to all other algorithms mentioned in this chapter including the 
MCMC-MLE trial and error approach. All three algorithms described in this chapter can be 
directly applied to networks of various sizes and with varying number of curved and/or non-curved 
parameters.
Bayesian Inference for Contact Networks

Given Incomplete Epidemic Data

4.1 Introduction

Network analysis techniques have earned a lot of attention during the past decade in studying the dynamics of epidemics. One of the few different types of assumptions made in such analysis is that the epidemic spreads only across the edges of a contact network (Keeling and Eames, 2005; Meyers et al., 2005; Ferrari, 2006). Most of the work under this assumption is based on simulations. First, some network with a set of properties is either considered to be given or simulated and then a disease outbreak is simulated on that contact network which is then used to study the properties of the epidemic (e.g. Volz, 2008; Barthlemy et al., 2005). As Welch et al. (2011) point out, an approach which is more in line with statistical inference is trying to answer the questions such as what are the properties of the disease outbreak and the network on which it spread given epidemic data that are assumed to come from a disease spread across that contact network. This is the focus of Groendyke et al. (2011a), which extended the work of Britton and O’Neill (2002). Discovering the properties of disease outbreaks enables the researchers to test different theories on
the transmission of a given disease and also to formulate more effective containment strategies. In reality, the underlying contact network may change over time and it may also change depending on the disease status of the individual. However, for now we work with the simplifying assumption that the contact network does not change over time and/or the disease status.

In this chapter, we extend the work of Groendyke et al. (2012a) and increase the flexibility of the underlying model. This is joint work with Dr. Michael Schweinberger and the corresponding manuscript is still in progress. The contents of Sections 4.3.1, 4.3.2, 4.3.3, 4.3.4 and 4.4 are taken directly from Schweinberger and Bomiriya (2014) and they describe the model we use in detail and some definitions we need later in this chapter.

4.2 Methods and Models

We will first introduce the notations and the methods described in Groendyke et al. (2012a) as we will be extending the concepts from that point onwards. The unpublished extensions begin from Section 4.3. Our assumption in this chapter is that the edges in the contact network $G$—an undirected network—correspond to all possible edges through which the disease can transmit, even though the disease may actually spread across a subset of those edges, which is represented by a directed network called the transmission tree that we denote by $P$. Next, we try to model our epidemic via an SEIR epidemic model that we describe in Section 4.2.1.

In this chapter, we label the individuals—or nodes—as integers going from 1 to $N$ and the existence of an edge represents a relationship that is sufficient for the particular disease to transmit from one to the other. Groendyke et al. (2012a) use ERGMs—the class of models we described in Section 2.1—in order to describe the contact network of the susceptible individuals in the population. More specifically, they use dyadic independence ERGMs to describe the contact network, where the probability of an edge between individual $i$ and $j$ is given by $p_{ij}$, independent of the presence of other edges in the network, and
\[ \log \left( \frac{p_{ij}}{1 - p_{ij}} \right) = \sum_{m} \nu_m X_{\{i,j\},m}, \] (4.1)

where \( X \) is a matrix of dyadic covariates and \( \nu_m \) is the corresponding parameter vector (Groendyke et al., 2012a).

### 4.2.1 The Susceptible-Exposed-Infectious-Removed (SEIR) Model

The SEIR model is commonly used for modelling disease transmissions (Keeling and Rohani, 2008). Throughout this chapter we assume that there is one individual who is infectious in this population initially, called the root node, and the remaining nodes are susceptible. In Chapter 5, we discuss a possible extension to this with multiple root nodes. The susceptible individuals can get exposed to the disease via sufficient contact with any of their already infectious contacts— infectious nodes who share an edge with the particular contact in susceptible state. Here, we model the disease transmission through an edge by an exponential distribution with mean \( 1/\beta \).

The exposed individuals become infectious after a certain period of time, which we model via a gamma distribution with mean \( k_E \theta_E \) and variance \( k_E \theta_E^2 \). Then, the infectious individuals will be removed from that state after remaining in that state for another period of time, which we model by another gamma distribution with mean \( k_I \theta_I \) and variance \( k_I \theta_I^2 \). Once they are removed, it is assumed that they no longer play any role in the current epidemic. Figure 4.1 graphically presents the SEIR process discussed above.

### 4.2.2 Data

As we attempt to model an epidemic based on the SEIR model, we consider the times at which each individual entered the exposed, infectious, and removed states to be the primary data for our model. We will denote all these times using the same notation that Groendyke et al. (2012a) used in their paper. For an individual \( j \), the exposed, infectious and removal times are denoted by \( E_j \), \( I_j \), and \( R_j \).
Figure 4.1: SEIR process for individual $i$ who was infected by individual $j$

$I_j$ and $R_j$, respectively; the sets of all such times are denoted $E$, $I$, and $R$, while the collective set of all times is denoted by $T = (E, I, R)$.

Making use of these times and any covariate values as we discussed earlier in Section 4.2, we first need to generate a contact network and then, conditional on the contact network, we need to generate a transmission tree. Figure 4.2 visualizes this data generating process.

Figure 4.2: Data-generating process: (a) generate a contact network; (b) generate an epidemic conditional on the contact network; initially infected population member is in green and labeled by $\kappa$, other infected members are in red, susceptible population members are in black; black lines indicate contacts, blue arrows indicate transmissions.

If not all values in $T$ are observed, the individual who was exposed to the disease at first may be unknown and hence the identity of the initially infected node will be considered as a parameter to estimate. We borrow all the notations from Groendyke et al. (2011a) and Britton and O'Neill
(2002) and extend them as required.

The identity of the initial exposed will be denoted by $\kappa$ and the set of the exposure times except for the initial exposed, that is, $E \setminus E_\kappa$, will be denoted by $E_{-\kappa}$. We label the nodes so that the ones who were infected during the epidemic are $1,\ldots,m$, where $m$ is the total number of infected nodes and $1 \leq m \leq N$. Groendyke et al. (2011a) perform inference on the model parameters using a Bayesian approach and denote the prior distribution for a generic parameter (say, $\delta$) by $\pi_\delta(\cdot)$, the likelihood function by $L(T|\delta,\ldots)$, and the posterior distribution of $\delta$ by $\pi_\delta(\cdot|T)$. For an edge $(a,b)$, $(a,b) \in \mathcal{P}$ if and only if $a$ infects $b$. Notice that if $(a,b) \in \mathcal{P}$, we must satisfy

$$I_a < E_b < R_a. \quad (4.2)$$

In addition, we have the following relationships: $m - 1 = |\mathcal{P}| \leq |\mathcal{G}| \leq N$, where $|\mathcal{P}|$ and $|\mathcal{G}|$ denote the number of (directed) edges in $\mathcal{P}$ and the number of (undirected) edges in $\mathcal{G}$, respectively.

### 4.2.3 Likelihood Calculation

In order to calculate the likelihood for the model, summing over all possible $\mathcal{G}$ and $\mathcal{P}$ would be necessary. We will once again borrow the content from Groendyke et al. (2011a, p. 3-8) as the likelihood and its decomposition are already clearly written in that paper. The likelihood is given by

$$L(T|\beta,k_E,\theta_E,k_I,\theta_I,p) = \sum_{\mathcal{G}} \sum_{\mathcal{P}} L(T|\beta,k_E,\theta_E,k_I,\theta_I,p,\mathcal{G},\mathcal{P}) f(\mathcal{P}|\mathcal{G}) f(\mathcal{G}|p). \quad (4.3)$$

The number of summands in Equation (4.3) is too high unless the problem we deal with has a very small number of nodes, hence, we treat $\mathcal{G}$ and $\mathcal{P}$ as additional parameters—as the computations become relatively easy given the values of $\mathcal{G}$ and $\mathcal{P}$. However, it is important to note that we are more interested in learning about the model parameters governing the contact network than the contact network itself, simply because the latter is not something that we can easily nail down.
In addition, we condition on the initial exposure time due to similar reasons and the fact that the likelihood depends on \( p \) only through \( G \) makes it possible for us to write the likelihood, as in Groendyke et al. (2011a), as

\[
L(E_{-}, I, R|\beta, k_E, \theta_E, k_I, \theta_I, G, P, E_{\kappa}) = L_1 L_2 L_3 L_4,
\]

where \( L_1 \) is the contribution to the likelihood from the edges over which the epidemic was transmitted (i.e. \( P \)), \( L_2 \) is the contribution to the likelihood from the edges over which the epidemic did not pass (\( G \setminus P \)), and \( L_3 \) and \( L_4 \) are the contributions because of the transition (from exposed to infectious) and removal processes, respectively. The likelihood function is defined to be 0 for any values of \( T \) that violate inequality (4.2). Thus, we obtain

\[
L_1 = \beta^{m-1} \exp \left[ -\beta \sum_{(a,b) \in P} (E_b - I_a) \right],
\]

\[
L_2 = \exp \left[ -\beta \sum_{(a,b) \in G \setminus P} \left[ \{(E_b \land R_a) - I_a\} \lor 0 \right] \right],
\]

\[
L_3 = \prod_{i=1}^{m} (I_i - E_i)^{k_{E,i-1}} \theta_E^{-m k_E} e^{-B/\theta_E} / \Gamma(k_E)^m,
\]

and

\[
L_4 = \prod_{i=1}^{m} (R_i - I_i)^{k_{I,i-1}} \theta_I^{-m k_I} e^{-C/\theta_I} / \Gamma(k_I)^m,
\]

where

\[
B = \sum_{i=1}^{m} (I_i - E_i) \quad \text{and} \quad C = \sum_{i=1}^{m} (R_i - I_i)
\]

Later in this chapter we use the notation \( \eta_{E,1}, \eta_{E,2}, \eta_{I,1} \) and \( \eta_{I,2} \) for the parameters \( k_E, \theta_E, k_I \) and \( \theta_E \) respectively so that we can denote \( \eta_E = (\eta_{E,1}, \eta_{E,2}) \) as the vector of parameters governing the time spent in the exposure state and \( \eta_I = (\eta_{I,1}, \eta_{I,2}) \) as the vector of parameters governing the time spent in the infectious state. We denote \( \beta, \eta_E \) and \( \eta_I \) collectively by \( \eta \) for simplicity.
4.3 Semi-Parametric Bayesian Population Model

It is possible that certain members in a given population have large number of contacts—higher degrees—than most other members in that population. Such members can transmit a disease to a greater number of their contacts, making them “super-spreaders” of a disease. The presence of super-spreaders suggests long-tailed degree distributions and it might be difficult to model these degree distributions properly by Binomial\((N - 1, p)\), distributions as they may lead to short-tailed degree distributions compared to real-world degree distributions (e.g., Jones and Handcock, 2003, 2004). This is the motivation for Schweinberger and Bomiriya (2014) to use semiparametric Bayesian population models based on Dirichlet processes that can model degree distributions of any form, including both short and long tailed degree distributions. The term Bayesian semiparametric is used because some parts (the part governing the degree sequence) of the model structure are not fixed, whereas the other parts (the parts governing the impact of covariates on contacts and the epidemic conditional on the contact network) are fixed. We incorporated this semi-parametric model into the R package `epinet`, making it easier to deal with a variety of degree distributions.

4.3.1 The Model

Let \(s_1(y), \ldots, s_n(y)\) be the sequence of degrees, where \(s_i(y) = \sum_{j \neq i} y_{ij}\) is the degree of population member \(i\). A natural model of the degree sequence \(s_1(y), \ldots, s_n(y)\) is given by the exponential family of distributions

\[
L(\theta; y) = \exp\left(\sum_{i=1}^{n} \theta_i s_i(y) - \psi(\theta)\right),
\]

(4.10)

where the degrees \(s_1(y), \ldots, s_n(y)\) are the sufficient statistics, the weights of the degrees \(\theta_1, \ldots, \theta_n\) are the natural parameters, and \(\psi(\theta)\) ensures that \(L(\theta; y)\) sums to 1. The exponential-family form of Equation (4.10) can be motivated by its maximum entropy property and its other attractive properties (Barndorff-Nielsen, 1978; Diaconis et al., 2011; Rinaldo et al., 2013). An additional convenient property of the exponential family is that the probability mass function (4.10)
factorizes:

\[ L(\theta; y) = \exp \left( \sum_{i=1}^{n} \theta_i s_i(y) - \psi(\theta) \right) = \prod_{j=1}^{n} \prod_{i<j} \exp \left( \chi_{ij}(\theta) y_{ij} - \psi_{ij}(\theta) \right), \] (4.11)

where

\[ \chi_{ij}(\theta) = \theta_i + \theta_j \] (4.12)

and

\[ \psi_{ij}(\theta) = \log(1 + \exp(\chi_{ij}(\theta))). \] (4.13)

To interpret the natural parameters \( \theta_1, \ldots, \theta_n \), note that the probability mass function (4.11) arises as a natural model of the degree sequence and incorporates the assumption that the \( Y_{ij} \) are independent Bernoulli(\( p_{ij} \)) random variables with probability \( p_{ij} \) and log odds \( \theta_i + \theta_j \). The first observation shows that the natural parameter \( \theta_i \) may be interpreted as the propensity of population member \( i \) to be in contact with others. The second observation shows that the log odds of the probability of a contact between population members \( i \) and \( j \) is additive in the propensities of \( i \) and \( j \) to be in contact with others.

If predictors of contacts are available, one may incorporate them by using the exponential family of distributions

\[ p(y | \theta, \vartheta) = \prod_{j=1}^{n} \prod_{i<j} \exp \left( \chi_{ij}(\theta, \vartheta) y_{ij} - \psi_{ij}(\theta, \vartheta) \right), \] (4.14)

where

\[ \chi_{ij}(\theta, \vartheta) = \theta_i + \theta_j + \vartheta^T t(v_i, v_j) \] (4.15)

and

\[ \psi_{ij}(\theta, \vartheta) = \log(1 + \exp(\chi_{ij}(\theta, \vartheta))). \] (4.16)

The function \( t(v_i, v_j) \) is a vector-valued function of covariates \( v_i \) and \( v_j \), which may be vectors, and \( \vartheta \) is a parameter vector.
4.3.2 Dirichlet process priors with hyper-priors

We assume that the degree parameters $\theta_1, \ldots, \theta_n$ are generated by a Dirichlet process prior (e.g., Ishwaran and James, 2001) of the form

$$
\begin{align*}
\theta_1 & \sim G, \\
\theta_i \mid \theta_1, \ldots, \theta_{i-1} & \sim \frac{1}{\alpha + i - 1} \left( \alpha G + \sum_{h=1}^{i-1} \delta_{\theta_h} \right), \quad i = 2, 3, \ldots,
\end{align*}
$$

(4.17)

where $\alpha > 0$ is a scaling parameter, $G$ represents a Gaussian distribution $N(\mu, \sigma^2)$ with mean $\mu$ and variance $\sigma^2$, and $\delta_{\theta_h}$ denotes a point mass at $\theta_h$. Therefore, draws from a Dirichlet process prior can be generated by first drawing from $N(\mu, \sigma^2)$ and drawing the $i$th draw with a probability proportional to $\alpha$ from $N(\mu, \sigma^2)$ and otherwise drawing one of the existing draws $\theta_1, \ldots, \theta_{i-1}$ uniformly at random. Since degree parameters are resampled, some population members share the same degree parameters. Thus, the Dirichlet process prior induces a partition of the population into subpopulations, where subpopulations share the same degree parameters.

If the parameters $\alpha$, $\mu$, and $\sigma^2$ of the Dirichlet process prior cannot be specified with confidence, one can express the uncertainty about these parameters by specifying hyper-priors, e.g., conjugate priors. We use conjugate hyper-priors described in Section 4.3.4 for all analyses in this chapter.

4.3.3 Truncation of Dirichlet process priors

To facilitate statistical computing, it is convenient to exploit the stick-breaking representation of the Dirichlet process prior. The stick-breaking construction of the Dirichlet process prior is given by

$$
\varphi_i = \theta^T Z_i, \quad i = 1, \ldots, N,
$$
where the components $\theta_k$ of parameter vector $\theta$ are subpopulation-dependent degree parameters,

$$Z_i \mid \omega_1, \ldots, \omega_K \sim \text{Multinomial}(1; \omega_1, \ldots, \omega_K), \ i = 1, \ldots, N$$

and

$$\theta_k \mid \mu, \sigma^2 \sim \mathcal{N}(\mu, \sigma^2), \ k = 1, 2, \ldots,$$

where the $Z_i$ are population member indicators of subpopulation membership. The name stick-breaking stems from the construction of the parameters $\omega_1, \ldots, \omega_K$:

$$\omega_1 = V_1,$$

$$\omega_k = V_k \prod_{j=1}^{k-1} (1 - V_j), \ k = 2, 3, \ldots,$$

where

$$V_k \mid \alpha \sim \text{Beta}(1, \alpha), \ k = 1, 2, \ldots.$$  

The Dirichlet process prior can be truncated by selecting a large integer $K > 0$ and sampling

$$\theta_k \mid \mu, \sigma^2 \sim \mathcal{N}(\mu, \sigma^2), \ k = 1, 2, \ldots, K,$$

then setting

$$\omega_1 = V_1,$$

$$\omega_k = V_k \prod_{j=1}^{k-1} (1 - V_j), \ k = 1, 2, \ldots, K - 1,$$

$$\omega_K = 1 - \sum_{k=1}^{K-1} \omega_k,$$

where

$$V_k \mid \alpha \sim \text{Beta}(1, \alpha), \ k = 1, 2, \ldots, K - 1.$$

Truncated Dirichlet process priors approximate stick-breaking priors and imply that the parameters
\( \omega_1, \ldots, \omega_K \) are governed by a generalized Dirichlet distribution (Ishwaran and James, 2001). The advantages of truncating the Dirichlet process prior are that the number of parameters of the truncated Dirichlet process prior is finite and that it facilitates the generation of samples from the posterior.

4.3.4 Markov chain Monte Carlo

We approximate the posterior given \( K \) components by combining the following Markov chain Monte Carlo steps by means of cycling or mixing (Tierney, 1994; Liu, 2008). The so-called label-switching problem of Bayesian Markov chain Monte Carlo algorithms, arising from the invariance of the likelihood function to the labeling of the subpopulations, can be solved along the lines of Stephens (2000).

**Scaling parameter** \( \alpha \):
If the hyper-prior of scaling parameter \( \alpha \) is Gamma\((A_1, B_1)\), we can sample \( \alpha \) from its full conditional:

\[
\alpha | A_1, B_1, \omega_1, \ldots, \omega_K \sim \text{Gamma}(A_1 + K - 1, B_1 - \log \omega_K).
\]

**Mean parameter** \( \mu \):
If the hyper-prior of mean parameter \( \mu \) is \( N(M, S^2) \), we can sample \( \mu \) from its full conditional:

\[
\mu | M, S^2, \sigma^2, \theta_1, \ldots, \theta_K \sim N \left( \frac{S^{-2}M + \sigma^{-2} \sum_{k=1}^{K} \theta_k}{S^{-2} + K \sigma^{-2}}, \frac{1}{S^{-2} + K \sigma^{-2}} \right).
\]

**Precision parameter** \( \sigma^{-2} \):
If the hyper-prior of precision parameter \( \sigma^{-2} \) is given by Gamma\((A_2, B_2)\), we can sample \( \sigma^{-2} \) from its full conditional:

\[
\sigma^{-2} | A_2, B_2, \mu, \theta_1, \ldots, \theta_K \sim \text{Gamma} \left( \frac{A_2 + K}{2}, B_2 + \sum_{k=1}^{K} \frac{(\theta_k - \mu)^2}{2} \right).
\]
Parameters $\omega_1, \ldots, \omega_K$:

We sample $\omega_1, \ldots, \omega_K$ from the full conditional by sampling

$$V_k^* \mid \alpha, Z_1, \ldots, Z_N \sim \text{Beta}\left(1 + N_k, \alpha + \sum_{j=k+1}^{K} N_j\right), \; k = 1, \ldots, K - 1$$

and setting

$$\omega_1 = V_1^*,$$

$$\omega_k = V_k^* \prod_{j=1}^{k-1} (1 - V_j^*), \; k = 2, \ldots, K - 1,$$

$$\omega_K = 1 - \sum_{k=1}^{K-1} \omega_k,$$

where $N_k$ denotes the number of population members in subpopulation $k$.

Indicators $Z_1, \ldots, Z_N$:

We sample indicator $Z_i$ from its full conditional:

$$Z_i \mid \{Z_j\}_{j \neq i}, \omega_1, \ldots, \omega_K, \vartheta, \theta_1, \ldots, \theta_K, y \sim \text{Multinomial}(1; \omega_{i,1}, \ldots, \omega_{i,K}),$$

where

$$\omega_{i,k} = \frac{\omega_k \prod_{i \neq j}^{N} p(y_{ij} \mid \vartheta, \theta_i = \theta_k, \{\varphi_h\}_{h \neq i})}{\sum_{l=1}^{K} \omega_l \prod_{i \neq j}^{N} p(y_{ij} \mid \vartheta, \theta_i = \theta_l, \{\varphi_h\}_{h \neq i}).}$$

Degree parameters $\theta_1, \ldots, \theta_K$ and covariate parameter $\vartheta$:

The full conditional of $\theta_1, \ldots, \theta_K$ and $\vartheta$ is intractable. We update $\theta_1, \ldots, \theta_K$ and $\vartheta$ by Metropolis-Hastings steps, where proposals are generated from random-walk, independence, or autoregressive proposal distributions (Tierney, 1994).
Contact network $Y$:
Denote $(E, I, R, Z)$ by $X$. If $U_{Y,ij} = 1$ where $U = \{U_{Y,ij}\}$, a matrix with its $ij$th element indicating whether or not $Y_{ij}$ is unobserved, we sample $y_{ij}$ from the full conditional:

$$Y_{ij} \mid x, \beta, u, \theta, \vartheta \sim \text{Bernoulli}(q_{ij}),$$

(4.18)

where

$$q_{ij} = \frac{\exp(-\beta \max(\min(E_j, R_i) - I_i, 0)) p_{ij}(1)}{p_{ij}(0) + \exp(-\beta \max(\min(E_j, R_i) - I_i, 0)) p_{ij}(1)}$$

(4.19)

and $p_{ij}(y_{ij})$ is given by

$$p_{ij}(y_{ij}) = \exp(\chi_{ij}(\theta, \vartheta)y_{ij} - \psi_{ij}(\theta, \vartheta)).$$

(4.20)

Transmission network:
If $U_{T,ij} = 1$ for all $i, j$, i.e., the transmission network is unobserved, we follow the approach of Groendyke et al. (2011b, 2012b) to sample the transmission network from the posterior.

Parameter $\eta$:
We follow the approach of Groendyke et al. (2011b, 2012b) and use the same Markov chain Monte Carlo steps discussed there.

4.4 Likelihood-Ignorable Incomplete-Data Mechanisms

We assume that the investigator knows which data are observed. Denote the transmission tree by $\{Z_{i,j}\}_{i,j=1}^n$. Let $A = \{A_E, A_I, A_R, A_Z, A_Y\}$ be indicators of which data are observed, where $A_E = \{A_{E,i}\}_{i=1}^n$, $A_I = \{A_{I,i}\}_{i=1}^n$, and $A_R = \{A_{R,i}\}_{i=1}^n$ indicate whether the exposure, infection, and removal times $E_i$, $I_i$, and $R_i$ are observed, respectively, and $A_Z = \{A_{Z,ij}\}_{i,j=1}^n$ and $A_Y = \{A_{Y,ij}\}_{i<j}^n$ indicate whether the values of $Z_{ij}$ and $Y_{ij}$ are observed, respectively. Also, we write $X = (E, I, R, Z)$ and refer to it as the epidemic.
The set of indicators $A$ is considered to be a random variable with observed outcome $a$ whose distribution depends on a parameter vector $\pi$. The observed and unobserved subset of $x$ (epidemic) and $y$ (contact network) are denoted by $x_{\text{obs}}$ and $x_{\text{mis}}$ and $y_{\text{obs}}$ and $y_{\text{mis}}$, respectively, and $x = (x_{\text{obs}}, x_{\text{mis}})$ and $y = (y_{\text{obs}}, y_{\text{mis}})$.

We refer to the conditional distribution $p(a \mid x, y, \pi)$ as the incomplete-data mechanism.

**Definition:** likelihood-ignorable incomplete-data mechanism. Let $\eta$ and $\theta$ denote the parameter vector of the epidemic and the contact network, respectively. If the parameters $\pi$, $\eta$, $\theta$ are distinct in the sense that the parameter space of $(\pi, \eta, \theta)$ is given by the product space $\Omega_\pi \times \Omega_\eta \times \Omega_\theta$; the parameter of the population model $\eta$ and $\theta$ and the parameters of the incomplete-data mechanism $\pi$ are independent under the prior, i.e.,

$$
p(\pi \mid \eta, \theta) = p(\pi) \text{ for all } \pi \in \Omega_\pi, \eta \in \Omega_\eta, \theta \in \Omega_\theta;
$$

the marginal prior of $\pi$ is proper; and the probability of observing data is independent of the values of the unobserved data, i.e.,

$$
p(a \mid x, y, \pi) = p(a \mid x_{\text{obs}}, y_{\text{obs}}, \pi) \text{ for all } x_{\text{obs}}, y_{\text{obs}}, x_{\text{mis}}, y_{\text{mis}}, \pi \in \Omega_\pi,
$$

then the incomplete-data mechanism is called likelihood-ignorable. Otherwise it is called non-ignorable.

### 4.5 Current Work

As we mentioned in the introduction, our intent in this chapter is to extend the work of Groendyke *et al.* (2012a) and make it more flexible and useful for the users. We have already extended this package in several ways. We have added a new feature that allows us to infer partial missing exposure or/and infectious times and also any partial missing removal removal times—when at
least one of the corresponding exposure or infectious times are known—under the assumption that the incomplete-data mechanism is likelihood-ignorable.

The second extension is with regard to modelling the varying shapes of degree distributions that we usually encounter in real networks. This was achieved by implementing the semiparametric Bayesian population model discussed in Section 4.3.

Thirdly, we extend the epinet package to condition for any observed contact information—such as a partially observed network—which was not possible with the previous version of epinet. It is conceivable in some situations to have at least partial reliable information on the contact network and we have now provided a way to make use of it. With the results that we currently have, we see that we can improve the results in Groendyke et al. (2012a) by implementing the proposed semiparametric model. We use the same Hagelloch dataset they used, in order to compare the results of our model to their model. The Hagelloch dataset is based on a measles outbreak that occurred in Hagelloch, a small town in Germany, in 1861. The data contain onset times of disease symptoms and some demographic details for each of the 188 infected individuals. The demographic information includes details such as the school class to which the individuals belong and the location of their houses. Prior to the work of Groendyke et al. (2011a) and Groendyke et al. (2012a), Britton et al. (2011) and Neal and Roberts (2004) have studied the Hagelloch data using non-network-based approaches. Neal and Roberts (2004) uses a stochastic epidemic model in which the transmission rate from an infected individual to a susceptible one is modeled as a function of the covariate information of the individuals. They fit models with different sets of available covariates and use a reversible jump MCMC algorithm to choose the best one among those models. Britton et al. (2011) uses a three-level mixing model and an SEIR epidemic model to analyze the data. The three-level structure they assumed is that every susceptible individual belongs to a household (lowest level), then to a group such as the school class (middle level) and to the whole community (highest level). Hence, the disease transmission may occur from an infected individual to another within the same household, group or community. They use independent Poisson processes with different rates to model the frequencies of each type of transmissions.
However, in this chapter, we will be looking only at network-based approaches.

Figure 4.3 (a) shows the posterior predictive check results that Groendyke et al. (2012a) obtained for the Hagelloch data based on the Erdős-Rényi model—i.e. an ERGM with only the edges term, no covariates or other model terms. The model appears to capture some information on the number of infecteds by each day of the epidemic; however, it does not pick up the mode of the corresponding distribution. Figure 4.3 (b) shows the results obtained for the same dataset but now by running a semiparametric model with three blocks without covariates. This model does better at picking up the peak of this distribution, which is one goal we wanted to achieve by implementing the semiparametric model in this package. We fit a series of models to this dataset by (i) varying the number of blocks and (ii) including and excluding the covariate information. We aim to use the results from all these runs to find the number of blocks that gives rise to the best fit as well as to demonstrate the use of covariate information in modeling. This work (Schweinberger and Bomiriya, 2014) is still in progress and we wish to use the multivariate version of Root Mean Squared Error (RMSE) as one measure for model selection. We define

$$\text{RMSE}(\hat{x}) = \sqrt{\frac{\sum_{i=1}^{s} (x_i - x_{i}^{\text{obs}})^2}{s}}$$

where $\hat{x}$ is a vector of $s$ statistics—e.g., the maximum number of infecteds estimated, the total number of infecteds estimated, and the number of days in which the estimated number of infecteds were greater than, say, 5% or 10% of the population—and $x_{i}^{\text{obs}}$ denotes the observed values of the corresponding statistics.

### 4.6 Inferring Unknown Removal Times

In the package ‘epinet’, we currently assume that some or all of the removal times are known in full. However, it is possible that the removal times are unknown while at least one of the exposure
or infectious times is known. In this section, we show that the full conditional distribution of an individual removal time can be written as a mixture of doubly- and left-truncated gamma distributions. Hence, when a removal time is unknown, we can use this mixture distribution to do a Gibbs update in the \texttt{epinet} package. However, once the full conditional distribution is derived, it seems that using a Metropolis-Hastings (M-H) update is more convenient and the Gibbs update would be more computationally expensive in this case. Hence, we implement an M-H update in the \texttt{epinet} code. Nevertheless, we provide the derivation of the full conditional distribution of a removal time for the purpose of completeness of this report.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure4.3.png}
\caption{Left plot (a): The number of individuals in the infectious state over time under the Erdős-Rényi model (Captured from Groendyke et al. (2012a)). The observed measles epidemic is given by the solid red line, whereas simulated data are shown in black, with boxplots summaries shown at each 5-day time increment. Right plot (b): The same plot obtained for the newly implemented semiparametric model with 2 blocks and without covariates (at each 5-day time increment). Here, the day axis has been centered at the mode of the distribution.}
\end{figure}
4.6.1 Full Conditional Distribution of an Individual Removal Time

Here we try to obtain the full conditional distribution of the $j^{th}$ individual removal time under the SEIR model considered in Groendyke et al. (2011a). Therefore, we consider all exposure times, all infectious times, all removal times except that of the $j^{th}$ individual, the Gamma parameters $k_E, \theta_E, k_I, \theta_I$, the exponential parameter $\beta$, the contact network, and the transmission tree to be known.

4.6.1.1 Notations

Let $I_j$ and $R_j$ represent the infectious and removal times of node $j$. Now let $a$ denote the number of contacts of $j$ that were never infected and let $c$ denote the number of contacts of $j$ that were infected after time $I_j$ by some other node.

![Figure 4.4: A timeline that displays some of the notations in this section.](image)

Furthermore, let $E_d$ denote the exposure time of the last node that was infected by node $j$ and let $E_{k_i}$ denote the exposure time of $k_i$—a contact of $j$—that was infected by some other node after time $I_j$ (we have $E_{k_1}$ up to $E_{k_c}$ if $c \geq 1$). Also, we have $X_j = (R_j - I_j) \sim \text{Gamma}(k_I, \theta_I)$ unconditionally. We denote the number of $j$’s contacts that were infected by some other node in
the interval \( (I_j, R_j] \) by \( W \), where \( W \in \{0, 1, \ldots, c\} \).

Therefore, given the contact network, transmission tree, \( k_E, \theta_E, k_I, \theta_I, \beta \), and \( I \), exactly one of, \( (E_d < R_j < E_{k_1}) \) or \( (E_{k_1} < R_j < E_{k_2}) \) or \( (E_{k_2} < R_j < E_{k_3}) \) or \( \ldots \) or \( (E_{k(c−1)} < R_j < E_{k_c}) \) or \( (R_j > E_{k_c}) \) must be true. As \( X_j = (R_j - I_j) \), we see from Figure 4.4 that

\[
W = 0 \iff d < X_j < k_1 \\
W = 1 \iff k_1 < X_j < k_2 \\
\ldots \\
W = c \iff X_j > k_c
\]

Recall that the time to transmission across a given edge is modelled by an exponential random variable with mean \( 1/\beta \). So, if \( j \) was not recovered at \( R_j \), the disease would have transmitted to the never-infected contacts of \( j \) at the above rate. Let the hypothetical transmission times from \( j \) to its never-infected contacts be \( Y_1, Y_2, \ldots, Y_a \) respectively. Let \( Y \) to be the minimum of these which means \( X_j < Y \). As all \( Y_i \) are i.i.d. \( \text{Exp}(\beta) \) random variables, we have \( Y \sim \text{Exp}(a\beta) \).

Let us denote \( P(W = i|X_j < Y, X_j > d) \) by \( w_i \), where \( i \in \{0, 1, 2, \ldots, c\} \) so that \( \sum_{i=0}^{c} w_i = 1 \).

### 4.6.1.2 The Likelihood Function

The likelihood function that we are dealing with is the one we discussed in Section 4.2.3. Let us consider the \( j \)th node and the full conditional distribution of \( X_j = R_j - I_j \) and denote its density by

\[
f^* = f(x_j|E, I, R_{-j}, \beta, k_E, \theta_E, k_I, \theta_I, G, \mathcal{P}),
\]

where \( R_{-j} \) is the set of all removal times except the removal time of node \( j \).

We know that \( f^* \propto L(E_{-\kappa}, I, R|\beta, k_E, \theta_E, k_I, \theta_I, E, G, \mathcal{P}) = L_1L_2L_3L_4 \) from Section 4.2.3. Let us look at the contribution from each of \( L_1, L_2, L_3 \) and \( L_4 \) to \( f^* \).
If \((j, b) \in \mathcal{P}\) for any \(b\), then by inequality (4.2) we have \(I_j < E_b < R_j\). Therefore, we see from Equation (4.5) that the only contribution from \(L_1\) to \(f^*\) is \(I\{X_j > d\}\) (\(\equiv I\{R_j - I_j > d\}\)), where \(d\) is as shown in Figure 4.4 in Section 4.6.1.1.

Next, we see from Equation (4.6) that the contribution from \(L_2\) to \(f^*\) comes from the contacts of node \(j\) who were infected after \(j\) became infectious but by a node other than \(j\)—there are \(c\) such nodes as per the notation in section 4.6.1.1—and from the never infected contacts of \(j\). As there are \(a\) never-infected contacts of \(j\), the contribution from them is \(e^{-a \beta X_j}\). Conditional on the contact network, the transmission tree and all the other \(E, I, R\) times, we know that \(X_j > d\) and \(X_j < Y\). If, for example, \(R_j\) falls into the interval \([d, k_1)\), then the corresponding \(L_2\) contribution to \(f^*\) from infected nodes would be \(I\{d < X_j < k_1\}e^{-(c-0) \beta X_j}\). However, there are \(c + 1\) possible intervals into which \(R_j\) can fall, as we discussed in Section 4.6.1.1. Therefore, the total contribution from \(L_2\) to \(f^*\) can be expressed as

\[
e^{-a \beta X_j} \left\{ I\{d < X_j < k_1\}e^{-(c-0) \beta X_j}P(W = 0 | X_j > d, X_j < Y) \right\} +
\]

\[
e^{-a \beta X_j} \left\{ I\{k_1 < X_j < k_2\}e^{-(c-1) \beta X_j}P(W = 1 | X_j > d, X_j < Y) \right\} +
\]

\[\vdots +
\]

\[
e^{-a \beta X_j} \left\{ I\{X_j > k_c\}e^{-(c-c) \beta X_j}P(W = c | X_j > d, X_j < Y) \right\}
\]

where \(W, a, c, Y\) are as in Section 4.6.1.1.

From Equation 4.7, we see that there is no contribution from \(L_3\) to \(f^*\). Lastly, we need to consider the contribution of \(L_4\) to \(f^*\). Equation (4.8) shows that this contribution is \(X_j^{k_i-1}e^{-x_j/\theta_I}\).

To make the notation easier, let us denote \(I\{d < X_j < k_1\}, I\{k_1 < X_j < k_2\},..., I\{X_j < k_c\}\) by \(I_0, ..., I_c\) respectively. Now, by putting everything together we have

\[
f^* \propto \sum_{i=0}^{c} P(W = i | X_j > d, X_j < Y)I_i X_j^{k_i-1}\exp\left\{ - \left[ (a + c - i)\beta + \frac{1}{\theta_I} \right] X_j \right\}I\{X_j > d\}. \tag{4.25}
\]
Let $\mathcal{I}_i X_j^{k_i-1} \exp\{-[(a + c - i) \beta + \frac{1}{\theta_J}] X_j\} I\{X_j > d\}$ (which is a kernel of a truncated gamma distribution) be denoted by $G_i$ for $i \in \{0, 1, \ldots, c\}$. Then, we can rewrite Equation (4.25) as

$$f^* \propto \sum_{i=0}^{c} w_i G_i.$$  (4.26)

We see from Equation (4.25) and (4.26) that $f^*$ is a mixture of doubly- and left-truncated gamma distributions since $X_j \sim \text{Gamma}(k_I, \theta_I)$ unconditionally. If we know the values of the $w_i$, we can update $X_j$ by drawing a random number from the above mixture distribution.

### 4.6.1.3 Calculating $w_i$

Recall that $w_i = P(W = i|X_j < Y, X_j > d)$, i.e., for $i \in \{1, \ldots, c-1\}$,

$$w_i = \frac{P(k_i < X_j < k_{i+1}, X_j < Y, X_j > d)}{P(X_j > d, Y > X_j)},$$  (4.27)

and all $k_i$ are greater than $d$. Therefore, $P(k_i < X_j < k_{i+1}, X_j < Y, X_j > d)$ equals

$$P(k_i < X_j < k_{i+1}, Y > X_j) = \int_{k_i}^{k_{i+1}} \int_{x}^{\infty} f_x f_y dydx,$$  (4.28)

as $X_j \perp Y$. The two facts that $X_j \sim \text{Gamma}(k_I, \theta_I)$ and $Y \sim \text{Exp}(a\beta)$ imply

$$P(k_i < X_j < k_{i+1}, Y > X_j) = \int_{k_i}^{k_{i+1}} \frac{x^{k_i-1}e^{-(x/\theta_I)}}{\Gamma(k_I)\theta_I^{k_I}} \int_{x}^{\infty} (a\beta)e^{-(a\beta)y} dydx$$

$$= \int_{k_i}^{k_{i+1}} x^{k_i-1}e^{-(1/\theta_I+a\beta)x}/\Gamma(k_I)\theta_I^{k_I} \ dx = (1 + a\beta\theta_I)^{-k_I} \int_{k_i}^{k_{i+1}} h(k_I, 1/\theta_I+a\beta) \ dx$$

$$= (1 + a\beta\theta_I)^{-k_I} P(k_i < Z < k_{i+1}),$$

where $Z \sim \text{Gamma}(k_I, 1/\theta_I+a\beta)$ and $h(a, b)$ represents the pdf of a Gamma distribution with mean $ab$ and variance $ab^2$.

Similarly, we find that $P(X_j > d, Y > X_j) = (1 + a\beta\theta_I)^{-k_I} P(Z > d)$, where $Z$ is as above.
Therefore, we have \( w_i = P(k_i < Z < k_{i+1})/P(Z > d) \), where \( Z \sim \text{Gamma}(k_i, \frac{1}{\theta_i + a\beta}) \) for \( i \in \{1, ..., c - 1\} \). We also have \( w_0 = P(d < Z < k_1)/P(Z > d) \) and \( w_c = P(Z > k_c)/P(Z > d) \) where \( Z \) is the same as before.

To generate a random variate \( X_j \) from the mixture distribution corresponding to the kernel \( \sum_{i=0}^{c} w_i G_i \), where \( w_i \) are the weights and \( G_i \) are truncated gamma kernal, we first draw a random number \( i \) from \( \{0, 1, ..., c\} \) with probabilities \( \{w_0, w_1, ..., w_c\} \). Then if \( U \sim G_i \), \( U \) is a random variate drawn from the above mixture distribution.

Even though we derive the full conditional distribution of \( X_j \), in practice we actually implemented an M-H update due to reasons mentioned before. We tested this approach on the Hagelloch measles dataset used in both Groendyke et al. (2011a) and Groendyke et al. (2012a). Figure 4.5 shows part of the results we obtained by considering 10 known removal times (out of a total of 187 removal times) to be unknown and inferring them with the implemented M-H update. Only a few posterior distributions are shown due to space limitations, though the others look similar.

### 4.7 Simulation Study

In this section, we demonstrate the properties of the proposed semi-parameteric Bayesian model via a set of simulation studies that cover (i) Parameter Recovery and (ii) Sampling Contacts. We discuss the results of each of these studies below. Given below is the setup common for the simulations mentioned above. We generate datasets from a population of size \( N \), a fixed number, and assign each individual to one of three sub populations with probabilities \( \pi_1, \pi_2 \) and \( \pi_3 \). Next, we sample contact networks from the model in Equation (4.10). As the three subpopulations share degree parameters, we only have three degree parameters, say \( \gamma_1, \gamma_2 \) and \( \gamma_3 \). An infectious disease then transmits through the edges of these contact networks with exponential rate of infection \( \beta \). Time in the exposed state for the \( i^{th} \) individual, \( I_i - E_i \) is distributed as \( \text{Gamma}(k_E, \theta_E) \) and time in the infectious state, \( R_i - I_i \), is distributed as \( \text{Gamma}(k_I, \theta_I) \). Exposure, infection, and removal
times are all assumed to be known, while the contact network and transmission tree are considered as unobserved except in Section (4.7.2), where we assume at least part of the contact network is observed.

4.7.1 Parameter Recovery

In order to assess whether the proposed model can recover its model parameters, we generated 1000 datasets, each with three sub-populations and a population size $N = 200$ as described above. Then, we fit semi-parametric Bayesian models with three and five blocks (sub-populations) for each of these datasets. However, due to the label-switching problem mentioned in Section 4.3.4—refer to Schweinberger and Handcock (2014) for more details—the block-label-dependent statistical
inference may not be carried out based on completely unprocessed samples obtained by the models. We summarize below, the results of the two scenarios stated above. The plots are shown only for the three-block experiment, but the plots for the five-block experiment look quite similar. It is important to note that, if we work with real-world data, we may not know the true number of blocks to which the members of the population belong to. Therefore, we need to find a way to estimate the true number of blocks based on the results we obtain. We do this as follows. First, we process the block parameter results using the label-switching algorithm mentioned before. Secondly, for each individual $i$ of the population, we look at the classification probabilities—i.e., the sample proportion of times in which, individual $i$ is a member of each sub-population (block)—to see what sub-population is the most likely one for that individual to be in (Say we store this information in vector $b$). Then, we estimate the true number of blocks by the number of unique block labels in $b$. In the analysis of block parameter recovery, we eliminate the runs in which the estimated number of blocks is not equal to three. We order the rest of the block parameter estimates by their sample means and consider them as the estimates of $\gamma_1$, $\gamma_2$ and, $\gamma_3$, respectively.

The results of the model with three blocks reveal that we are able to correctly estimate the true number of blocks as three, 100% of the times—i.e., we never estimate the true number of blocks to be less than three in this scenario. In the case, where we fit a model with five blocks, this percentage becomes approximately 82%, which is still a satisfying percentage. The data-generating values of the parameters and the frequentist coverage properties of 95% posterior credible intervals are shown in Table 4.1. The coverage is obtained as the number of times in which the 95% posterior credible interval contains the true parameter—this is based only on the runs in which we estimated the true number of blocks correctly. The results in this table show that, when we use a model with three blocks, we have a coverage of approximately 95% for all the parameters except the three contact network parameters, $\gamma_1$, $\gamma_2$, and $\gamma_3$. With five blocks, we manage to recover the block parameters pretty well, when we correctly estimate the true number of blocks. Figure 4.6 shows the credible intervals for parameters for each iteration and Figure 4.8 shows the histograms of the posterior medians. It is evident from Figures 4.6 and 4.8 that the contact network parameters are somewhat biased. Figure 4.7 presents the parameter estimates for the contact network parameters.
It clearly shows that even though the contact network parameters are biased, the bias is smaller so that we still see a clear separation between the three distinct data-generating values of these parameters. These results suggest that we can recover the contact network parameters to a large extent despite the fact that we don’t observe the contacts between individuals. In Section 4.7.2, we demonstrate that we can reduce the associated statistical error by sampling contacts. Therefore, the results of this simulation study indicates that we can recover all parameters, including the block parameters, with the proposed semi-parametric Bayesian model.

**Table 4.1:** Frequentist coverage properties of 95% posterior credible intervals: Number of times 95% posterior credible intervals covered data-generating values of parameters $\beta$, $k_E$, $\theta_E$, $k_I$, $\theta_I$, $\gamma_1$, $\gamma_2$, and $\gamma_3$, based on models with three blocks and five blocks, respectively.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>$\beta$</th>
<th>$k_E$</th>
<th>$\theta_E$</th>
<th>$k_I$</th>
<th>$\theta_I$</th>
<th>$\gamma_1$</th>
<th>$\gamma_2$</th>
<th>$\gamma_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>True Value</td>
<td>2</td>
<td>8</td>
<td>0.25</td>
<td>4</td>
<td>0.25</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
</tr>
<tr>
<td>Coverage</td>
<td>With 3 blocks</td>
<td>95.4%</td>
<td>95.4%</td>
<td>93.5%</td>
<td>95.2%</td>
<td>94.3%</td>
<td>83.4%</td>
<td>97.4%</td>
</tr>
<tr>
<td>Coverage</td>
<td>With 5 blocks</td>
<td>93.0%</td>
<td>96.3%</td>
<td>96.3%</td>
<td>96.0%</td>
<td>95.7%</td>
<td>98.4%</td>
<td>100%</td>
</tr>
</tbody>
</table>

**Figure 4.6:** Credible intervals of the parameters $\eta_{E,1}$, $\eta_{E,2}$, $\eta_{I,1}$, $\eta_{I,2}$, $\beta$, $\gamma_1$, $\gamma_2$, and $\gamma_3$ based on the results by running a semi-parametric model with 3 blocks on 1000 simulated datasets.
Figure 4.7: Estimates of the block parameters $\gamma_1 = -2$, $\gamma_2 = -1$ and $\gamma_3 = 0$ based on the results of running a semi-parametric model with 3 blocks on 1000 simulated datasets with an MCMC sample of size 1000, for each dataset.

4.7.2 Sampling Contacts

To illustrate how the associated statistical error can be reduced by sampling contacts, we generate 1000 datasets of population size $N = 187$ coming from a model with degree parameters $\gamma_1 = -3.5$ (low-degree), $\gamma_2 = -1.5$ (moderate-degree) and $\gamma_3 = 0.5$ (high-degree). Then, 1000 egocentric samples of size $n = 25, 50, 75, 100, 125, 150, 187$—i.e., random samples of $n$ individuals of the population from whom we collect information on all of their contacts—are generated for each dataset. By construction, we expect the parameters relating to infection and removal times—i.e., $k_E$, $\theta_E$, $k_I$, and $\theta_I$—to be insensitive to the sample size $n$. The RMSE (square root of the MSE) of posterior medians and means of the four parameters across different sample sizes $n$ are shown in Figure 4.10 and we see that the results are well in line with our expectation of no sensitivity with regard to the sample size $n$. Figure 4.9 shows the reduction in RMSE with increasing sampled contacts for the rate of infection, $\beta$, and for the contact network parameters $\gamma_1$, $\gamma_2$ and $\gamma_3$. 
4.8 Discussion & Conclusions

In this chapter, we demonstrate how a semi-parametric Bayesian population model can improve the inference we make on infectious disease dynamics given epidemic data, especially in terms of
Figure 4.10: RMSEs for $k_I, k_E, \theta_I, \theta_E$ for varying sample sizes.

capturing the important peaks of the distributions. We are in the process of analyzing the the Hagelloch data related results obtained with covariate information and with varying number of blocks. Based on the results we obtain for models with different block sizes but without using covariate information, we observe that we can obtain very good results in terms of reproducing the observed curve of infectious individuals by day, even by using a small number of blocks such as 2. We conducted a simulation study to assess how well we can recover the SEIR model parameters and contact network parameters and also to see how the availability of information on contacts can improve the quality of the estimates. Based on results in Section 4.7.1, it is evident that we can recover the model parameters. Section 4.7.2 results show that sampling contacts can greatly reduce the bias and RMSE of the parameters $\beta$ and $\gamma$ and also confirm our belief that the bias and RMSE of parameters $k_I, k_E, \theta_I$ and $\theta_E$ are insensitive to the sample size $n$. We are continuing the analysis based on the covariate information and hope to demonstrate the usefulness of using such information in modeling. We have implemented all extensions discussed in this chapter as
modifications to the package `epinet`; namely, we have implemented (i) the estimation of fully or partially missing Exposure (E), Infectious (I) and/or Removal (R) times (keeping in mind that we must observe at least one of E, I, and R times for each individual), (ii) the semi-parametric Bayesian population model, and (iii) the option to include full or partial information on the network ties if known. Currently, we are in the process of integrating these changes with a completely different set of modifications made to `epinet` by the owner of the package, Chris Groendyke, and his collaborator, David Welch.
Chapter 5

Summary & Discussion

5.1 Introduction

This thesis is comprised of three major chapters, all of which fall under the common theme of models for random networks based on linear combinations of statistics derived from the structural features and covariate information of the associated networks.

Social ties are often dependent upon actor and/or event attributes in addition to the structural aspects of the underlying social network. Chapter 2 extends the related attribute-based homophily concept from the unipartite case to the bipartite case in a way that avoids possible degeneracy issues that may occur with existing measures such as the direct event and actor two star statistics described in Agneessens et al. (2004). The proposed terms are readily available to the public as a part of the R package `ergm`, a package for network analysis that can be used to fit, simulate and diagnose ERGMs and is a part of the `statnet` suite of packages for network analysis. In Chapter 2, we only discuss models that include a minimal number of statistics in order to keep the interpretations simple. However, if interested, one may include any other appropriate statistic(s) into the model along with the proposed statistics—see Wang et al. (2013) for a wide range of statistics defined for bipartite/multilevel networks. Even though the idea of discounting attribute-
based two-stars (or two-paths) in the proposed homophily terms appears to be natural, estimation, particularly maximum likelihood estimation, for the underlying discount parameters is somewhat complicated. To deal with this issue, we take two approaches: (i) Fix the discount parameter at various values in the interval \([0,1]\), estimate the other model parameters, and pick the discount parameter that results in the highest loglikelihood (trial-and-error method) as we do in Chapter 2; and (ii) let the discount parameter be unknown and estimate it simultaneously with the other parameters via Bayesian inference. A general version of the latter approach is the topic covered in Chapter 3. Bayesian inference for ERGMs is possible using the R package \texttt{Bergm}. We extend the capabilities of this package by allowing one to estimate the discount parameters mentioned above or any other curved ERGM parameters together with the other parameters in the model. Time taken per iteration may be longer for estimating the model via the Bayesian approach, however, in order to come to a conclusion based on the MLE approach, one must run a series of models with varying values of the discount parameter. Thus, the MLE may require more computing time overall. One main advantage of the Bayesian approach over the MLE approach is that, in addition to being able to estimate the discount parameter, it has the flexibility to estimate the type of the discount parameter—i.e., \(\alpha\) or \(\beta\) in Chapter 2 as a new parameter. If one uses the MLE approach to select the value of the discount parameter and to select the type of the discount parameter, two series of models need to be run, one based on \(\alpha\) and the other based on \(\beta\). The extensions carried out in Chapter 3 have been implemented as R functions and we have discussed making them available as a part of the package \texttt{Bergm} in the near future with the maintainers of that package.

Understanding the properties of a disease outbreak is vital for determining effective containment strategies and testing a variety of hypotheses to develop novel theories. One strategy to go about this is to assume that the disease spreads only through the edges of the underlying social network structure and analyze the disease outbreak under this assumption given epidemic data. In Chapter 4, we extend the work of Groendyke \textit{et al.} (2012a), which is based on the above strategy, in several different ways. The tools developed by Groendyke \textit{et al.} (2012a) are publicly available in the form of an R package called \texttt{epinet}, which provides tools for simulating disease transmission
though a contact network and conducting Bayesian inference on network and epidemic parameters given epidemic data. The work of Groendyke et al. (2012a) assumes the underlying contact networks are completely unknown and may be inferred as part of the process. However, if we have at least partial information about the contacts, this information will enhance the quality of the inferences we make. Thus, we have added this feature to the soon-to-be published version of epinet. The relationship between Chapter 2 and Chapter 4 comes into play when we model the network ties based on the structural features and covariates of the unknown or partially known contact network. If the underlying network is assumed to be bipartite or multi-level in nature, we can incorporate the statistics proposed in Chapter 2 when modeling the contact network as part of larger ERGMs. This has not yet been implemented as part of ergm, however, if one may wish to do so, this is possible. It may also be possible to extend these ideas to track and make decisions on an ongoing disease outbreak.

We identify several other extensions with regard to the work in Chapter 4. For example, in the current version of epinet, we assume that we only have one infected node initially and the disease starts spreading across the contacts of this initially infected node, which leads to the transmission tree that we discussed in Chapter 4. However, it is possible that multiple nodes get infected around the same time and start spreading the disease to their own contacts simultaneously. For example, suppose a family of four (mother, father, son and daughter) who were traveling outside the country got infected on their way back home, but were unaware of the infection. As soon as they join their community, they start spreading the disease to the ones they come into contact with. Children may spread the disease to their friends in school. The parents may spread the disease among their peers at work etc. This leads us to have multiple disjoint disease transmission trees rather than a single tree as we currently have. Therefore, we see that allowing multiple root nodes will help us model the spread of the disease more realistically.

There are several issues that we need to consider here. The first issue is that we may not know the number of initially infected nodes. Therefore, we may need to estimate that as an additional parameter (Say \( \nu \)). Secondly, we need to draw \( \nu \) possible parents for these transmission trees.
Then, we need to update the transmission trees. The main task with regard to this extension would be designing an MCMC algorithm that can sample from all possible sets of disjoint trees. For example, the MCMC algorithm should be able to swap nodes from one tree to another disjoint tree—as in Figure 5.1—in an efficient manner.

![Current state](image) ![Next state](image)

**Figure 5.1:** An example of a transmission tree update that should be possible with the new MCMC algorithm for an epidemic with two root nodes.

In our current work, we assume that the number of infected nodes and the identity of these nodes are known. However, it is possible that we know only a sample of the infecteds in the population and we may need to estimate this number of infecteds. This change, however, would have a big impact on the current code and we would need to design a new MCMC algorithm. It may be be better to design an MCMC algorithm that works for both this extension and the extension related to multiple root nodes as it will save a lot of time and effort by doing so compared to doing them separately. We may also consider other possibilities, such as allowing the recovered nodes to become susceptible nodes once again rather than ignoring them completely after the recovery. This would be an interesting feature to have in an analysis of a disease where people can get infected multiple times. This means we are not ending the process with susceptible-exposed-infectious-removal states, but continuing it cyclically as susceptible-exposed-infectious-recovered-susceptible and so on. The moment that we assume the recovered individuals can become susceptible again, the disease transmission can no longer be expressed in terms of a tree, as it may now contain loops.

In summary, we have developed tools for (i) modeling homophily in exponential random graph models for networks that are bipartite in nature, (ii) estimating additional parameters of an ERGM simultaneously with the other coefficients, and (iii) inferring infectious disease dynamics given epi-
emic data via semiparametric Bayesian population models that accommodate degree distributions of any form. Chapters 2, 3 and 4 of this thesis can be viewed as contributions to three different existing R packages, namely, \texttt{ergm}, \texttt{Bergm}, and \texttt{epinet}, respectively, and the contributions will soon be available to the public.
Appendix

A.1 R Code - Bayesian Inference for Curved ERGMs

#-----------------------------------------------
# BAYESIAN INFERENCE FOR CURVED ERGMs
#-----------------------------------------------
#
# Date: 11/25/2013 (By : Rashmi Bomiriya)
#
# A modified version of "bergm" function of R package "Bergm" ("Bergm" package
# does not deal with curved ergms unless discount parameters are considered to
# be fixed. Here, we modify the bergm code to allow the discount parameters or
# any other additional parameters involved in curved ERGMS to be estimated
# along with the other parameters).
#
# Inputs:
#-----------------------------------------------

# y - observed network as a network object
# get.formula -function to get model formula with different curved.pars
bcergm <-function(y, get.formula, N, nstats, nfactors, nchains, rho, func.list, 
               burnin = 0, 
               mu.prior = NULL, 
               sigma.prior = NULL, 
               sigma.proposal = NULL, 
               aux.iters = 10000, 
               thinning = 1 
           )
require(ergm)
require(mixtools)
ptm <-proc.time()
y0 <<- y

#----------------- processing inputs - I -----------------
thin.ind <- seq(1 + burnin, N + burnin, by = round(thinning))
npars <- nstats + nfactors
Theta <- array(NA, c(length(thin.ind), npars, nchains))

#---- Setting up starting values of the chains ----------
theta <- matrix(runif(npars * nchains, min = -0.1, max = 0.1),
                 npars, nchains)
c.par <- t(sapply(1:nfactors,
                  FUN = function(i)func.list[[i]](theta[nstats+i,])))

acc.counts <- rep(0, nchains)
theta1 <- rep(0, npars)
tot.iters <- burnin + N
ind <- 1:nstats
control <- control.simulate.formula(MCMC.burnin = aux.iters,
                                     MCMC.interval = 0)
control$MCMC.samplesize <- 1
ADS.move <- 0

#----------------- processing inputs - II -----------------

if (is.null(mu.prior))
    mu.prior <- rep(0, npars)
if (is.null(sigma.prior))
  sigma.prior <- diag(100, npars)
if (is.null(nchains))
  nchains <- 2 * (npars)
if (is.null(sigma.proposal))
  sigma.proposal <- diag(0.0025, npars)
if ((npars-1) == 1)
  nchains <- 1
  sigma.epsilon <- diag(rho, npars)

i <-1

#----------------------------------------------------------
for (k in 1:tot.iters)
  for (h in 1:nchains)
    if (npars > 1 && nchains > 1)
      ADS.move <- rho * apply(theta[,sample(seq(1,
          nchains)[-h], 2)], 1, diff)

#-----------------
eta.curr <- theta[, h]
eta.nw <- eta.curr + ADS.move +
  rmvnorm(1, sigma = sigma.proposal)[1, ]
c.curr <- c.par[, h]
c.nw <- t(sapply(1:nfactors, 
    FUN = function(i)func.list[[i]](eta.nw[nstats+i])))

pr <- dmvnorm(rbind(eta.nw, eta.curr),
    mu = mu.prior, sigma = sigma.prior)

#----------------- Gibbs update for y (network) -----------

form.1 <- get.formula(c.curr, "y0")
model.1 <- ergm.getmodel(form.1, y0)
    Clist.1 <- ergm.Cprepare(y0, model.1)
    MHprop.1 <- MH.prop(model.1, y0, control)

    set.seed(i)
    delta.1 <- ergm.mcmcslave(Clist.1, MHprop.1, eta0 = eta.nw[ind],
        control, verbose = FALSE)$s

form.2 <- get.formula(c.nw, "y0")
model.2 <- ergm.getmodel(form.2, y0)
    Clist.2 <- ergm.Cprepare(y0, model.2)
    MHprop.2 <- MH.prop(model.2, y0, control)

    set.seed(i)
    delta.2 <- ergm.mcmcslave(Clist.2, MHprop.2, eta0 = eta.nw[ind],
        control, verbose = FALSE)$s
i <- i + 1

# ---------- Calculating Acceptance Prob ---------------

prob <- (t(eta.curr[ind]) %*% delta.1 - t(eta.nw[ind]) %*% delta.2
        + log(pr[1]) - log(pr[2]))

# ---------- Accept / Reject Proposal -----------------

if (prob >= log(runif(1)))
    theta[, h] <- eta.nw
    c.par[, h] <- as.vector(c.nw)
    if (k > burnin)
        acc.counts[h] <- acc.counts[h] + 1

# END OF h LOOP

if (is.element(k, thin.ind))
    Theta[match(k, thin.ind), , ] <- theta

# END OF k LOOP

if (nchains == 1)
    Theta <- as.matrix(Theta[, , 1])

for(i in 1:nfactors)
    Theta[, (nstats+i), ] <- func.list[[i]](Theta[, (nstats+i), ])
time <- proc.time() - ptm
out = list(nchains = nchains, Theta = Theta, acc.rates = acc.counts/N, 
          time = time, dim = npars)
return(out)

#------------------------------------------------------------------------
# FUNCTION TO SET UP M-H PARAMETERS IN ERGM
#------------------------------------------------------------------------

MH.prop <-function(model, netwk, control)
MHprop <- MHproposal.ergm(object = model, constraints = ~., 
                         arguments = control$MCMC.prop.args, nw = netwk, 
                         weights = control$MCMC.prop.weights, 
                         class = "c", reference = ~Bernoulli, response = NULL)
return(MHprop)

#------------------------------------------------------------------------
# COEXPANSION DATASET - EXAMPLE 01
#------------------------------------------------------------------------

getModelFormula <-function(curved.par, net.name)
out <-paste(net.name, " ~ edges + b2nodematch('firm.type',

diff = TRUE, keep = 2:4, beta = ", curved.par[1], ") +
  b1nodematch(’city’, beta = ", curved.par[2], ")", sep = "")
out <-formula(out)
return(out)

#---------------------------------------------------------------
# INPUT PARAMETERS
#---------------------------------------------------------------
set.seed(12345678)
load("globalExpansion.RData")

net <-globalExpansion
nstats <-5
nfactors <-2
npars <-nstats + nfactors
nchains <-npars * 2
func.list <-list(inv.logit = function(x)exp(x)/(1 + exp(x)),
  inv.logit = function(x)exp(x)/(1 + exp(x)))
rho <-0.5
N <-10
burnin <-0

#---------------------------------------------------------------
# FITTING THE MODEL
#---------------------------------------------------------------
fit = bcergm(y = net, getModelFormula, N, nstats, nfactors, nchains, rho,
  func.list,
  burnin = 0,
  mu.prior = NULL,
A.2 A Comparison of Parameter Estimates

Table 1 summarizes the posterior means and the corresponding standard deviations obtained by running the three algorithms; EX-ADS-MIX, EX-ADS-BETA and EX-ADS-NORM on global expansion data.

Table 1: Global Expansion Data: Parameter estimates (posterior means) based on 10000 MCMC iterations of the algorithms EX-ADS-MIX (Mixture Proposal), EX-ADS-BETA (Dependent Beta Proposal) and EX-ADS-NORM (Multivariate Normal Proposal).

<table>
<thead>
<tr>
<th>Statistic</th>
<th>EX-ADS-MIX</th>
<th>EX-ADS-BETA</th>
<th>EX-ADS-NORM</th>
</tr>
</thead>
<tbody>
<tr>
<td>edges</td>
<td>4.1143 (0.7303)</td>
<td>4.3650 (1.3929)</td>
<td>4.1378 (0.5726)</td>
</tr>
<tr>
<td>b2nodematch.firm.type.Advertising</td>
<td>-6.9787 (1.4060)</td>
<td>-7.3211 (2.4872)</td>
<td>-7.0838 (1.1245)</td>
</tr>
<tr>
<td>b2nodematch.firm.type.BankingFinance</td>
<td>-7.3498 (1.4163)</td>
<td>-7.7082 (2.4283)</td>
<td>-7.4372 (1.1310)</td>
</tr>
<tr>
<td>b2nodematch.firm.type.Law</td>
<td>-9.2197 (1.2811)</td>
<td>-9.6634 (2.2398)</td>
<td>-9.2952 (1.0180)</td>
</tr>
<tr>
<td>( \beta )</td>
<td>0.0016 (0.0062)</td>
<td>0.0036 (0.0126)</td>
<td>0.0003 (0.0019)</td>
</tr>
</tbody>
</table>
Bibliography


Vita
Rashmi P. Bomiriya

Research Interests: Statistical Analysis on Social Networks, Exponential Random Graph Modeling, Statistical Computing, Bayesian Modeling

Education:
- Doctor of Philosophy in Statistics, Department of Statistics, Pennsylvania State University, August 2014
  Advisor: David R. Hunter, Professor of Statistics, Head of the Department of Statistics
- Bachelor of Science in Statistics (Hons.), Department of Statistics, University of Colombo, Sri Lanka, October 2008

Experience:
- Senior Student Statistical Consultant/Research Assistant, Statistical Consulting Center, Department of Statistics, Pennsylvania State University, Fall 2013
- Joint instructor for Applied Forensic Science Statistics (FRNSC 597A), Forensic Science, Pennsylvania State University, Fall 2013
- Student Statistical Consultant, Statistical Consulting Center, Department of Statistics, Pennsylvania State University, Fall 2012, Spring 2013
- Professional Experience:
  - Associate Vice President, Amba Research Lanka (Pvt) Ltd, Sri Lanka, April 2010 - August 2010

Honors and Awards:

Submitted Journal Publications:

Papers in Preparation:
- Schweinberger, Michael and Bomiriya, Rashmi, A Semiparametric Bayesian Approach to Network-Based Stochastic Epidemics with Likelihood-Ignorable Incomplete-Data Mechanisms, 2014

Conferences:
- Oral presentation on Modeling Homophily in ERGMs for Bipartite Networks at Sunbelt XXXIV International Social Network Conference, St. Petersburg, Florida, Feb 2014
- Poster presentation on Modeling Homophily in ERGMs for Bipartite Networks at Rao Prize Conference, Pennsylvania State University, Oct 2013
- Joint oral presentation on Dashboards to Manage Student Project Work: Innovation in Statistical Consulting at the Penn State Symposium for Teaching and Learning with Technology, Mar 2014