University of Louisville

ThinkIR: The University of Louisville's Institutional Repository

Electronic Theses and Dissertations

8-2023

Penalized Bayesian exponential random graph models.

Vicki Modisette
University of Louisville

Follow this and additional works at: https://ir.library.louisville.edu/etd

Part of the Other Statistics and Probability Commons

Recommended Citation

Modisette, Vicki, "Penalized Bayesian exponential random graph models." (2023). *Electronic Theses and Dissertations.* Paper 4139.
Retrieved from https://ir.library.louisville.edu/etd/4139

This Doctoral Dissertation is brought to you for free and open access by ThinkIR: The University of Louisville's Institutional Repository. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of ThinkIR: The University of Louisville's Institutional Repository. This title appears here courtesy of the author, who has retained all other copyrights. For more information, please contact thinkir@louisville.edu.

PENALIZED BAYESIAN EXPONENTIAL RANDOM GRAPH MODELS

By

Vicki Modisette
B.A., Union University, 2015
M.A., University of Louisville, 2019

A Dissertation
Submitted to the Faculty of the
College of Arts and Sciences of the University of Louisville
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy
in
Applied and Industrial Mathematics

Department of Mathematics
University of Louisville
Louisville, Kentucky

August 2023

PENALIZED BAYESIAN EXPONENTIAL RANDOM GRAPH MODELS

Submitted by

Vicki Modisette

A Dissertation Approved on

July 7, 2023

by the Following Dissertation Committee:

Dr. Dan Han,
Dissertation Director

Dr. Ryan Gill

Dr. Csaba Biro

Dr. Jian Du-Caines

DEDICATION

To my first math teacher, my mother

To my biggest supporter, my husband

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation to my advisor Dr. Han. Thank you for your patience, dedication, interest, and skill. You have been the best.

To my dissertation committee, thank you. Your attentiveness and encouragement are so appreciated.

Thank you to the kind and supportive faculty, staff, and students at the University of Louisville. Working with you has been an honor and joy.

Finally, I want to extend a special thank you to my family and friends, particularly, Erica, Chas, Amy, Jessica, Jonathan, Jerry, and Ciana. Your support means the world to me. And to Ryan, thank you for believing in me.

ABSTRACT

PENALIZED BAYESIAN EXPONENTIAL RANDOM GRAPH MODELS

Vicki Modisette

July 7, 2023

Networks have the critical ability to represent the complex interconnectedness of social relationships, biological processes, and the spread of diseases and information. Exponential random graph models (ERGM) are one of the popular statistical methods for analyzing network data. ERGM, however, struggle with computational challenges and degeneracy issues, further exacerbated by their inability to handle high-dimensional network data. Bayesian techniques provide a promising avenue to overcome these two problems. This paper considers penalized Bayesian exponential random graph models with adaptive lasso and adaptive ridge penalties to perform variable selection and reduce multicollinearity on a variety of networks. The experimental results demonstrate their effectiveness in variable selection and reduction of multicollinearity across diverse networks, outperforming the widely used Bayesian exponential random graph model proposed by Caimo et al. [10], which lacks regularization capabilities. This paper presents a valuable extension to network models for large-scale high-dimensional data and offers opportunities for advancing research in diverse fields.

TABLE OF CONTENTS

DEDICATION iii

ACKNOWLEDGEMENTS iv

ABSTRACT v

LIST OF TABLES ix

LIST OF FIGURES x

1. INTRODUCTION 1

   1.1 Network Substructures 2

      1.1.1 Node-Level Substructures 2

      1.1.2 Dyad-Level Substructures 4

      1.1.3 Geometrically Weighted Edgwise Shared Partners 5

   1.2 Motivation and Outline 6

2. EXPONENTIAL RANDOM GRAPH MODELS 9

   2.1 Introduction and Overview 9

   2.2 Model Structure 11

   2.3 Classical Inference for ERGMs 13

      2.3.1 Maximum Pseudolikelihood Estimation 13

      2.3.2 Monte Carlo maximum likelihood estimation 14

3. BAYESIAN EXPONENTIAL RANDOM GRAPH MODELS 16

   3.1 Adaptive Parallel Direction Sampling Algorithm 18

   3.2 Bayesian Exponential Random Graph Model algorithm 19

4. BAYESIAN ADAPTIVE LASSO EXPONENTIAL RANDOM GRAPH MODEL 21

4.1 Background of the Lasso Penalty 21

4.2 Model Derivation 22

4.3 Sampling Methods for Lambda 25

4.4 Unimodalilty of Posterior Distribution 28

5. BAYESIAN ADAPTIVE LASSO ERGM - FULL BAYES 32

5.1 Hierarchical Model 32

5.2 Joint Distribution 33

5.3 Gibbs Sampling Implementation 33

5.4 Full Bayes Algorithm 36

6. BAYESIAN ADAPTIVE LASSO ERGM - DELTA EMPIRICAL . . . 38

6.1 Empirical Derivation 38

6.2 Empirical Delta Algorithm 39

7. BAYESIAN ADAPTIVE LASSO ERGM - LAMBDA EMPIRICAL . . 41

7.1 Empirical Derivation 41

7.2 Full Empirical Algorithm 42

8. BALERGM APPLICATIONS AND SIMULATIONS 44

8.1 Simulation 44

8.1.1 Data Description 44

8.1.2 Results 46

8.1.3 Goodness of Fit 52

8.2 Variable Selection 55

8.2.1 Data Description 55

8.2.2 Results 57

8.2.3 Goodness of Fit 61

8.3 Application 62

8.3.1 Data Description 63

8.3.2 Results 64

9. BAYESIAN ADAPTIVE RIDGE EXPONENTIAL RANDOM GRAPH MODEL 69
9.1 Classical Ridge Penalty 69
9.2 Prior Specification 71
9.3 Bayesian Ridge Exponential Random Graph Model- Cauchy Prior 71
9.3.1 Half Cauchy 72
9.3.2 Hierarchical Model One 73
9.3.3 Gibbs Sampling 73
9.4 Algorithm 76
9.5 Bayesian Ridge Exponential Random Graph Model- Gamma Prior 76
9.5.1 Hierarchical Model 77
9.5.2 Gibbs Sampling 78
9.6 Algorithm 79
9.7 Data Description 80
9.8 Results 82
9.8.1 VIF function 85
9.9 Goodness of Fit 86
10. CONCLUSIONS 88
REFERENCES 91
CURRICULUM VITAE 104

LIST OF TABLES

5.1 Sampling distributions from joint distribution for each variable . 34
8.1 Number of student friendships in each grade pair 46
8.2 Initial settings for the simulation runs 47
8.3 Results of simulating 30 graphs and comparing results for BERGM and BALERGM using means as the estimates of $\boldsymbol{\theta}$ 50
8.4 Results of simulating 30 graphs and comparing results for BERGM and BALERGM using medians as the estimates of $\boldsymbol{\theta}$ 51
8.5 Results of both BERGM and BALERGM using formula y ∼ edges + nodematch("Grade")+nodematch("race") + nodematch("sex") 52
8.6 Initial settings for the variable selection runs 57
8.7 BALERGM estimates of $\boldsymbol{\theta}$ 59
8.8 Results three methods of BALERGM using formula y ∼ edges + nodematch("Grade") + nodecov("GPA") + nodecov("Wealth") + nodematch("Sport") 60
8.9 Results from BALERGM with Variable Selection on SCAL data 67
8.10 Variable Selection with Different Tolerance Levels 68
9.1 Sampling distributions from joint distribution for each variable . 75
9.2 Estimates for $\boldsymbol{\theta}$ by BERGM 82
9.3 Estimates for $\boldsymbol{\theta}$ by Ridge BERGM with Cauchy prior 82
9.4 Estimates for $\boldsymbol{\theta}$ by Ridge BERGM with Gamma prior 83
9.5 VIF results for Bayesian adaptive Ridge BERGM with both Gamma and Cauchy priors and BERGM 86

LIST OF FIGURES

1.1 Network representations 2

1.2 Examples of Node-level Network Structures 3

1.3 Examples of Dyad-Level Network Structures 4

1.4 One, two, and three edgewise-shared partners. 6

3.1 Parallel Adaptive Sampling Diagram 20

8.1 Faux Dixon High School Plot 45

8.2 MCMC output of BERGM algorithm 48

8.3 MCMC output of BALERGM Method 1 algorithm 48

8.4 MCMC output of BALERGM Method 2 algorithm 49

8.5 MCMC output of BALERGM Method 3 algorithm 49

8.6 Goodness of Fit Diagnostics for BALERGM Method 1 53

8.7 Goodness of Fit Diagnostics for BALERGM Method 2 54

8.8 Goodness of Fit Diagnostics for BALERGM Method 3 54

8.9 Faux Magnolia High School Plot 55

8.10 Histogram of the simulated GPA values for Faux Magnolia High 56

8.11 MCMC output of BALERGM Method 1 algorithm 58

8.12 MCMC output of BALERGM Method 2 algorithm 58

8.13 MCMC output of BALERGM Method 3 algorithm 59

8.14 Goodness of Fit Diagnostics for BALERGM Method 1 61

8.15 Goodness of Fit Diagnostics for BALERGM Method 2 62

8.16 Goodness of Fit Diagnostics for BALERGM Method 3 62

8.17 Generated in R, this plot shows results of asking “Have you heard of a group called Sumter County Active Lifestyles (SCAL)?” . . 64
9.1 Size of node determined by the size of population served, the color of node determined by years 81
9.2 MCMC plot for BERGM 83
9.3 MCMC plot for Ridge with Cauchy prior 84
9.4 MCMC plot for Ridge with Gamma prior 84
9.5 Bayesian goodness of fit diagnostics for ridge exponential random graph model 87

CHAPTER 1
INTRODUCTION

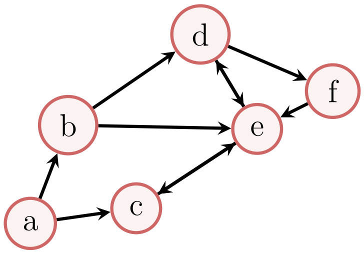
Networks are an important method of representing relational data in many fields. For example, in sociology and psychology, researchers have modeled friendships and the spread of information with networks of nodes representing people and edges formed on some condition of connection. An early example is Moreno and Jennings' work in Sociometry using networks to describe the friendships of a girls' school [69]. Later, Granovetter highlighted the strength of networks in this realm as a method for representing both micro/individual and macro/society decisions [37]. In a more modern context, researchers have studied networks of terrorists to better understand the structures of these groups [78]. Similarly, Tsvetovat and Carley use properties of networks to provide recommendations for the most effective destabilizing techniques for law enforcement seeking to remove the threats from terrorist networks [95].

Biological contexts for networks abound including gene regulatory networks and protein interaction networks in addition to the neural networks in our brains. Since the interactions between the nodes in a gene regulatory network (GRN) inform the development of the cells in living organisms, understanding the dynamics of these networks has implications across the field of biology [75]. Additionally, protein-protein interaction networks are significant to understanding many processes in the human body [57] [15].

Network models have proven value across a wide variety of disciplines despite the limitations of current models. This next section lays out the mathematical representation of networks and the substructures that form the building blocks of the models.

1.1 Network Substructures

Mathematical analysis of these networks requires representation beyond a graphical perspective. The structure of a network with n nodes can be represented with an n by n adjacency matrix. Figure 1.1a illustrates the graphical representation of a directed network with six nodes. The corresponding adjacency matrix is represented in 1.1b. In this matrix, a tie between any two nodes is denoted by 1 in the corresponding position of the adjacency matrix. Unlike an undirected graph that is symmetric, this matrix is asymmetric since an edge from node a to node b is distinguished from an edge from node b to node a.



$$\begin{array}{cc} \begin{array}{cccccc} a & b & c & d & e & f \end{array} & \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} & \begin{array}{c} a \\ b \\ c \\ d \\ e \\ f \end{array} \end{array}$$

(a) Directed network

(b) Adjacency Matrix

Figure 1.1: Network representations

Once this adjacency matrix is defined, we can observe substructures representing the local dynamics of the network [70]. For this context, we will focus on node-level and dyad-level substructures.

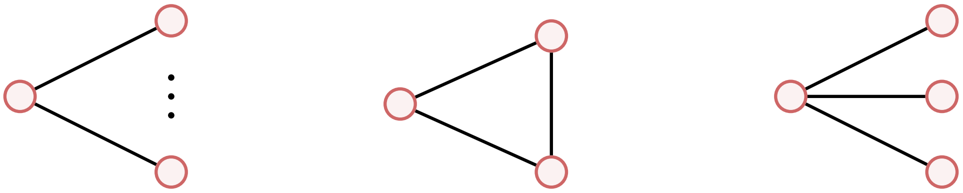
1.1.1 Node-Level Substructures

Node-level statistics consider the attributes of nodes themselves providing information about the network through the characteristics of the nodes. Figure 1.2

represents three common node-level statistics: degree, triangles, and the k-star substructure. The degree of a node or the number of nodes it is connected to can indicate the amount of influence or interaction a node has with a network. This feature has significant application for the study of both information and disease spreading with the degree distribution showing the presence of hubs or high contact nodes.

The number of triangles or the connection of three nodes as seen in Figure 1.2b can indicate the level of connection in a network with higher numbers of this feature indicating a more densely connected network. This utility is particularly important to the social sciences and the context of social networks.

Finally, the n-star formation for any $n \in \mathbb{N}$ is shown in Figure 1.2c with $n = 3$. This structure occurs on undirected graphs, but similar but more detailed structures exist in directed graphs. Ties directed to the central node (instar) or directed out from the central node (outstar) can also be helpful distinctions depending on the network context. Nodes that have a large number of nodes connected to them, exhibit popularity and can indicate a node or agent with influence. Activity refers to the tendency of nodes that have k-stars with edges directing to other agents. Both of these substructures are particularly significant to understanding influential nodes in a network.



(a) Degree (b) Triangle (c) 3-star

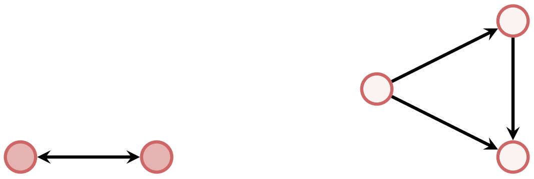
Figure 1.2: Examples of Node-level Network Structures

Using the adjacency matrix in the style of Figure 1.1, the counts of these network substructures can be calculated for any given network. Given the adjacency matrix $\boldsymbol{Y}$ with entries in the matrix defined as $Y_{i,j}$ for a network, the following sums calculate the occurrences of the respective network configurations for the network statistic.

$$\text{Degree of node i: } \sum_j Y_{i,j} \quad \text{Triangles: } \sum_{i<j<k} Y_{j,k}Y_{i,k}Y_{i,j} \quad \text{3-star: } \sum_{i<j<k<l} Y_{i,l}Y_{j,l}Y_{k,l}$$

1.1.2 Dyad-Level Substructures

Dyad-level statistics consider the interactions in a network between pairs of nodes. Each of these substructures provides information about agent-level connections driving the network shape and structure.



(a) Homophily (b) Transitive Triad

Figure 1.3: Examples of Dyad-Level Network Structures

Figure 1.3 displays two of these examples of network substructures: homophily and transitive triads. Homophily refers to the tendency of nodes with similar characteristics to be connected in a network. Here such nodes are denoted by the same color. The inclusion of homophily in the network analysis allows us to study social dynamics and community-based questions. In social networks, this is a frequently observed phenomenon when people form friendships or other connections with those that share a common demographic or occupational context.

Next, for directed graphs, a transitive triad considers three pairs of dyads with node 1 connected to node 2, node 2 connected to node 3, and node 1 connected to node 3 in contrast to triangles that merely consider three nodes with the connections between. Again a transitive triad is frequently experienced in social contexts: a friend of your friend is more likely to be your friend. This hints at the foundational assumption that distinguishes network analysis from classical regression techniques that assume observations are independent as the foundation for any model. The network models of this paper incorporate dependence by including these network substructures as components of the terms of the model. This structure allows for the analysis of data from a completely new angle and has promoted the utility of network models across many areas of study.

The count of these networks statistics is found with the adjacency matrix realization $\boldsymbol{y}$ of $\boldsymbol{Y}$ with i-j entry in the matrix defined as y_{ij}. For an undirected network, the following summations demonstrate a method for counting these substructures.

$$\text{Homophily: } \frac{1}{2}\sum_{i<j} Y_{i,j}Y_{j,k}\delta(X_i, X_j) \quad \text{Transitive Triad: } \sum_{i<j<k} Y_{i,j}Y_{j,k}Y_{i,k}$$

where δ is the Kronecker delta function returning 1 if a node attribute X_i is equal to the node attribute X_j.

1.1.3 Geometrically Weighted Edgewise Shared Partners

In addition to the previously mentioned network statistics based on simple configurations, there are other functions of the network topologies that model more complex interactions. The geometrically weighted edgewise shared partner (GWESP) network statistic captures the tendency for nodes in a network to form connections with common neighbors. It extends the standard edgewise shared partners (ESP) term (see Figure 1.4) by incorporating the following geometrically weighted function:

$$v(\boldsymbol{y};\theta_t) = e^{\theta_t}\sum_{i=1}^{n-2}\left\{1-(1-e^{-\theta_t})^i\right\} EP_i(\boldsymbol{y}) \tag{1.1}$$

for a given network $\boldsymbol{y}$ where θ_t is a constant decay parameter, n is the number of nodes in a network, and $EP_i(\boldsymbol{y})$ is the count of the node pairs that have exactly i edgewise shared partners [53]. The upper limit of the summation, $n-2$, is the maximum number of edgewise-shared partners for a graph with n nodes.

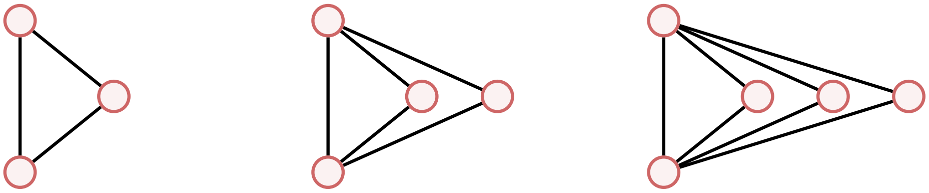


Figure 1.4: One, two, and three edgewise-shared partners.

1.2 Motivation and Outline

Network models are built on the task of finding the probability of a tie between any two nodes. Initial models for networks assumed that the probability of a tie between any two nodes was constant. In other words, the chance of a tie between any two nodes was independent of the other nodes' connections. This makes every edge a Bernoulli trial, so the probability of the entire graph is

$$P(\mathbb{G}) = p^M (1-p)^{\binom{n}{2}-M} \tag{1.2}$$

where $\mathbb{G}$ is a graph with M edges [21]. The assumptions of this model fail to capture many real-world examples of networks motivating the next big step in network models. Modern network models work from the assumption that local selection forces are dependent on factors in the network at large. The exponential random graph models (ERGM), also known as p^* models, emerged in the late 1970s and early 1980s as a statistical framework to capture the complex dependencies and patterns in social networks by using those local sub-configurations [64].

ERGM have proven to be valuable tools for analyzing network data, however, ERGM have computational limitations and are prone to degeneracy, particularly in the case of large or complex networks. Degeneracy occurs when the estimated parameters from the ERGM process indicate graphs unlikely to be applicable. such as the empty or complete graph; the indicated network fails to reflect the observed data showing a lack of goodness of fit. Additionally, the normalizing constant required for direct estimation requires millions of calculations for a graph of only 7 nodes with the required calculations increasing at an exponential rate for each additional node. While there are methods that attempt to get around this difficulty (discussed in Chapter 2), these come with compromises and assumptions [42]. The ergm package for R contains functions attempting to diagnose degeneracy, but methods to reduce it are lacking [52] [43] [59].

Thus, there are three main motivations for this dissertation. First, while potential applications abound, ERGM have computational difficulties, second, current network models have significant degeneracy issues, particularly for networks of common interest i.e. large and/or complex networks, and finally, ERGM lack regularization methods for variable selection and handling multicollinearity on the high dimensional networks. In this era of big data, the need for effective models that can assist with cutting through the complexity to reveal features of interest is an increasingly pressing need. The Bayesian exponential random graph model discussed in Chapter 3, provides the computational improvements that ERGM need but still lacks regularization methods. This thesis pioneers the Bayesian adaptive lasso exponential random graph and the Bayesian ridge exponential random graph addressing the need for parameter selection and reducing multicollinearity. These models have the potential to pivotally contribute to the future of network studies.

This dissertation is constructed to address these challenges as follows. Chapter 2 discusses the history and methods of ERGM. Next, Chapter 3 lays out the foundation for the Bayesian exponential random graph model which provides the double benefit of computational efficiency and prior specification advantages. We then lay out the new Bayesian Adaptive Lasso exponential random graph model (BALERGM) in chapter 4. To perform variable selection, we provide three alternative methods of parameter estimations in Chapters 5 (Method 1), Chapter 6 (Method 2), and Chapter 7 (Method 3). Each of these subsections includes the results of the new Bayesian Adaptive Lasso exponential random graph model compared to the existing Bayesian exponential random graph model, demonstrating the advantages of the lasso penalty in parameter selection on three data sets.

Then, the new ridge penalized Bayesian exponential random graph model and results are in Chapter 8. This chapter presents two variations of a new Bayesian ridge exponential random graph model with two different priors. Finally, the discussion and plans for future work are in the conclusion.

CHAPTER 2
EXPONENTIAL RANDOM GRAPH MODELS

The first section of this chapter lays out the background and context for the exponential random graph model. The next section covers the structure of this model incorporating the network statistics from Chapter 1. Next, Section 2.3 focuses on the traditional methods of estimation for ERGM. This model has proved to be widely applicable but faces computational issues motivating the theory of the next chapter.

2.1 Introduction and Overview

Disciplines such as sociology, political science, and biology have all relied on networks as a way of understanding and representing relationships between friends, global trading partners, proteins, and genes. Classical statistical methods have limited tools for fully capturing this relational data. Exponential random graph models (ERGMs) allow for quantifying this type of data demonstrating how the local selection forces shape the global structure of a network. Unlike the typical assumption of independence in classical regression, exponential random graph models assume an interdependence built into the structure of the network; the probability of any given edge existing is related to the existence of other edges and the nature of nodes in the entire graph. This assumption of dependence reflects an intuitive understanding of how certain networks are built. Assuming a dependence between the existence of ties was the catalyst for the development of ERGMs by Frank and Strauss in 1986 [25].

Historically, Erdös and Rényi propose the first random graph model in the late 1950s [21]. Initially, these models assumed that all graphs of the same size were equally likely or that the edges are independent, but such models have obvious limitations.

In 1981, Holland and Leinhardt produce the next development with a model for directed graphs limited to only using dyads that were assumed to be independent [50]. Later work moved away from the limitations of the assumption of independence and introduced Markov random graph models to create the structure of ERGMS that would stand for decades [25].

Fundamentally, ERGMs are analogous to logistic regression when dyads are independent; these models perform regression-like analysis but on a random network. ERGMs calculate the probability that a pair of nodes in a network will have a tie between them. However, the implementation of the estimations methods has computational difficulties because of the intractability of the normalizing constant and degeneracy problems [14]. Bayesian computational methods have provided ways to get around these difficulties. We will discuss this development further in Chapter 3.

Exponential Random Graph Models (ERGM) are widely applicable to research questions in the social and health sciences. In psychology, researchers studied Romanian school children's friendship networks to find that sex and mental health showed patterns of homophily, concluding that ERGM are a "promising avenue for further research" [4]. Also in the social and health sciences, Becker et al. considered the friendship network of members of a sorority and the influence of disordered eating habits on friendship finding that women tended to have disordered eating habits unlike their friends [6]. This unexpected result has implications for understanding the complex social dynamics that go into a serious health concern. Solo et al. note the utility and suitability of ERGM for modeling connections within the brain compared to more traditional methods, though they also note the computational difficulty of ERGM [88]. On a much larger scale, ERGM have been used to understand the influences of information sharing on tourism. The model helped answer questions about the existence of patterns in the network including whether or not the network exhibited the characteristic of homophily and how organizations should understand their role in the network [102]. In the realm of biology, Stivala et al. show that ERGM can address some of the limitations that previous research had found in modeling biological processes [89]. This small sampling of papers shows the incredible flexibility and significance of exponential random graph models.

2.2 Model Structure

The connectivity of the network's graph is described by an $n \times n$ adjacency matrix $\boldsymbol{Y}$. Its i-j entry $Y_{i,j} = 1$ if node i will give referral to node j and $Y_{i,j} = 0$ otherwise. Let $\mathscr{Y}$ be the set of all possible graphs on n nodes and let $\boldsymbol{y}$ be a realization of $\boldsymbol{Y}$. A given network $\boldsymbol{y}$ consists of n nodes and m edges that define a relationship between pairs of nodes called dyads. The adjacency matrix of the network graph $\boldsymbol{Y}$ allows for the analysis of the structural relationship in the observed network.

Using the methods of Chapter 1, we can define $s(\boldsymbol{y})$ as a vector of network statistics of the counts of the network substructures. We can use these network structures in the construction of the Exponential Random Graph Model. We see this in Ove Frank and David Straus' work in 1986 [25] [50] and expanded in more recent works [52] [64] [14] [70] [98].

For general exponential random graph models, the network has the following exponential family type density [64]:

$$\pi(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{1}{z(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^T s(\boldsymbol{y})} \tag{2.1}$$

where $\boldsymbol{y}$ is the observed network, $\boldsymbol{\theta}$ is a vector of parameters, and $s(\boldsymbol{y})$ is a vector of network statistics. Each i-th network statistic $s_i(\cdot)$ has a corresponding parameter θ_i. Here $z(\boldsymbol{\theta})$ is a normalizing constant, and $z(\boldsymbol{\theta}) = \sum_{\boldsymbol{y} \in \mathscr{Y}} e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}$ where $\mathscr{Y}$ is the set of all possible graphs with the same number of nodes as $\boldsymbol{y}$. The number of possible graphs with n nodes is $2^{n(n-1)/2}$ which becomes very large for all but the smallest graphs [64]. Hence, the calculation of $z(\boldsymbol{\theta})$ is feasible only for small networks in computer computation. It becomes challenging to find this normalization constant for large networks or even moderate-sized networks. The intractability of $z(\boldsymbol{\theta})$ is a well-known difficulty of ERGM.

We need one final component to the model. As we saw earlier, the transitive triad is just one example of many showing that it is not reasonable to assume that every tie between nodes is an independent variable. Instead, we see that the probability of a tie is dependent on the ties around it. With this assumption and the need to modify the model to account for all the ties at once, since the probability of one tie is connected to the existence of all other ties. We consider the change statistics or the difference in the occurrences.

Let $\boldsymbol{\delta} = s(\boldsymbol{y}_{ij}^{+}) - s(\boldsymbol{y}_{ij}^{-})$ be the vector of changes in the statistics in $\boldsymbol{s}$ when the edge y_{ij} between node i and j in the graph $\boldsymbol{y}$ changes from 1 to 0 along with the complement part $\boldsymbol{y}_{ij}^{c}$ same. Conditioned on the state of the rest of the graph represented as $\boldsymbol{Y}_{-ij}$, the log odds of the probability of a tie existing between node i and j is:

$$\log \frac{P(Y_{ij} = 1 | \boldsymbol{Y}_{-ij} = \boldsymbol{y}_{-ij}, \boldsymbol{\theta})}{P(Y_{ij} = 0 | \boldsymbol{Y}_{-ij} = \boldsymbol{y}_{-ij}, \boldsymbol{\theta})} = \boldsymbol{\theta}^{T} \boldsymbol{\delta}. \tag{2.2}$$

These network statistics can be overlapping subgraph configurations such as the number of edges, mutual edges, triangles, and uniform homophily etc. The representation above gives the intuitive explanation of the model parameter $\boldsymbol{\theta}$ about their effect on the probability of an edge between node i and j [58].

We can see the similarity to typical classical regression with constants that indicate the relative significance of the corresponding predictor variables for the probability of a tie. The magnitude of a number indicates the significance of that variable to the network with large numbers showing the variable is more significant. Positive values indicate a higher probability of a tie, while negative values decrease the probability of a tie [58].

2.3 Classical Inference for ERGMs

The inferential statistical goal is to find an appropriate estimate of $\boldsymbol{\theta}$ such that the corresponding generated network has the probability distribution centered on the observed network on average. That is, we want to solve the moment equation:

$$\mathbb{E}_{\boldsymbol{\theta}}(s(\boldsymbol{y})) = s(\boldsymbol{y}_{\text{obs}}) \tag{2.3}$$

where $\boldsymbol{y}_{obs}$ is the observed network and $s(\boldsymbol{y})$ is a vector of network statistics in the proposed graph and $s(\boldsymbol{y}_{\text{obs}})$ is a vector of the network statistics in the observed graph. However, in most cases, the moment equation cannot be solved analytically. This challenge leads to two mainstream simulations: maximum pseudolikelihood estimation and Monte Carlo maximum likelihood estimation.

2.3.1 Maximum Pseudolikelihood Estimation

The direct maximum likelihood estimation of ERGMs is complicated since the likelihood function is difficult to compute for models and networks of moderate or large size. [90] proposed a standard approximation with maximum pseudolikelihood estimation (MPLE). Instead of conditioning each tie on the state of the entire graph, the assumption is that the dependence of each dyad is weak. In particular, the MPLE estimates can be obtained by assuming the independence among values of Y_{ij}:

$$P(Y_{ij} = 1 | \boldsymbol{\delta}_{-ij} = \boldsymbol{y}_{-ij}) = P(Y_{ij} = 1). \tag{2.4}$$

This allows for the pseudolikelihood function that has the strength of quick estimation but has been shown to not provide reliable estimates [96] [26].

$$\pi(\boldsymbol{y}|\boldsymbol{\theta}) \approx \pi_{pseudo}(\boldsymbol{y}|\boldsymbol{\theta}) = \prod_{i\neq j} \pi(y_{ij}|\boldsymbol{y}_{-ij}, \boldsymbol{\theta}) \tag{2.5}$$

$$= \prod_{i\neq j} \frac{\pi(y_{ij} = 1|\boldsymbol{y}_{-ij}, \boldsymbol{\theta})^{y_{ij}}}{[1 - \pi(y_{ij} = 0|\boldsymbol{y}_{-ij}, \boldsymbol{\theta}]^{y_{ij}-1}} \tag{2.6}$$

This will only provide the true estimate for ERGM with dyadic independence or when the change statistics can be found only considering one tie without knowing the rest of the graph. Research by [96] compares the maximum pseudo-likelihood and maximum likelihood estimates, and their study shows the pseudo-likelihood estimation is biased and MPLE can only approximate the transitivity pattern in the network well.

2.3.2 Monte Carlo maximum likelihood estimation

Similar to methods in linear regression, ERGMS are log-linear, and a typical method for finding the maximum likelihood requires finding the roots of the derivative of the log of the function. This results in the $s(\boldsymbol{y})^T - \mathbb{E}_{\boldsymbol{\theta}}(s(\boldsymbol{y})) = 0$ found earlier. The Monte Carlo maximum likelihood estimation in ERGM case needs to find the following important ratio [96]:

$$\frac{z(\boldsymbol{\theta})}{z(\boldsymbol{\theta}_0)} = \mathbb{E}_{\boldsymbol{y}|\boldsymbol{\theta}_0}\left[\frac{e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}}{e^{\boldsymbol{\theta}_0^T s(\boldsymbol{y}_{\text{obs}})}}\right] \tag{2.7}$$

The log-likelihood equation, however, is not directly solvable without computing the normalizing constant. As previously mentioned, this is computationally intensive for all but the smallest graphs. With this approximation, though, the normalizing constant can be estimated by generating m graphs from the density $\pi(\boldsymbol{\pi}|\boldsymbol{\theta}_0)$ and finding $e^{(\boldsymbol{\theta}-\boldsymbol{\theta}_0)^T s(\boldsymbol{y}_i)}$ for each graph and use importance sampling technique. The estimates of $\boldsymbol{\theta}$ can be obtained by maximizing the log-likelihood ratio approximated as the following:

$$\ell(\boldsymbol{\theta}) - \ell(\boldsymbol{\theta}_0) \approx (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T - \ln \left[\frac{1}{m} \sum_{i=1}^{m} e^{(\boldsymbol{\theta}-\boldsymbol{\theta}_0)^T s(\boldsymbol{y}_i)} \right] \tag{2.8}$$

However, in this method, the choice of the initial $\boldsymbol{\theta}_0$ is tricky and should be near the maximum likelihood estimate of $\boldsymbol{\theta}_0$. Poor choice of $\boldsymbol{\theta}_0$ can lead to the failure of the maximization log-likelihood function and degeneracy problem [96] [42].

CHAPTER 3

BAYESIAN EXPONENTIAL RANDOM GRAPH MODELS

A Bayesian approach provides a potential solution to address the two issues of ERGM intractability and degeneracy as discussed in Chapter 2. In this regard, Caimo and Friel [9] introduce a Bayesian exponential random graph model (BERGM) which improves upon the Monte Carlo maximum likelihood method of Geyer [30] and the maximum pseudo-likelihood of Straus et al. [90]. By utilizing Bayesian analysis, Caimo and Friel eliminated the need for calculating the normalizing constant in ERGM. Instead, they focus on estimating the posterior distribution of model parameters given the observed data. This approach allows for more flexibility in modeling and inference, as it leverages prior information and incorporates it into the parameter estimation process. Second, the issue of degeneracy in ERGMs often arises when the model places most of its probability mass on just a few possible networks, such as the complete or empty network. This poses a challenge for common Markov Chain Monte Carlo (MCMC) algorithms used to simulate and estimate ERGMs, as they struggle to converge when parameter values are located near these degenerate regions. Consequently, in the Bayesian analysis of ERGMs, it becomes necessary to address the problems of instability and near-degeneracy by selecting a prior distribution based on a procedure that takes the data into account. This selection process ensures that the prior distribution provides appropriate regularization and helps overcome the difficulties associated with instability and near-degeneracy. Therefore, in the Bayesian analysis of ERGMs, the choice of a data-dependent prior distribution becomes crucial in mitigating the challenges posed by instability and near-degeneracy. In particular, when dealing with large-scale network data that includes numerous nodal covariates, it becomes increasingly challenging to determine the appropriate prior settings for all model parameters. The complexity and dimensionality of the data make it difficult to select suitable prior distributions that adequately capture the underlying structure of the network. This challenge serves as a key motivation for the present project, which aims to develop a network model that incorporates variable selection and addresses the issue of multicollinearity. By introducing these additional components, this project seeks to overcome the limitations of traditional approaches and enhance the modeling and inference process for large-scale network data.

This chapter outlines the methods of researchers Caimo and Friel in creating BERGM by explaining how Bayesian analysis removes the requirement for calculating the normalizing constant and adjusts the required Metropolis Hasting algorithm. Next, we detail the parallel adaptive sampling algorithm utilized by Caimo and Friel to improve the sampling of $\boldsymbol{\theta}$ [9]. Although the algorithms mentioned above are primarily used for BERGM, the principles and methodologies underlying Markov Chain Monte Carlo (MCMC) in BERGM can be adapted and extended to the penalized exponential random graph models.

Assume that a prior distribution $\pi(\boldsymbol{\theta})$ is placed on $\boldsymbol{\theta}$, and we are interested in the posterior distribution

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}). \tag{3.1}$$

While priors on $\boldsymbol{\theta}$ provide various utilities, the equation is now doubly intractable since both the normalizing constant $z(\boldsymbol{\theta})$ and model evidence $\pi(\boldsymbol{y})$ are computationally prohibitive to produce [64]. Consequently, an approximate exchange algorithm was introduced by Caimo and Friel [10]. The exchange algorithm consists of a Gibbs update of augmented $\boldsymbol{\theta}'$ followed by a Gibbs update of the network $\boldsymbol{y}'$. In this context, the standard Metropolis-Hastings ratio for the move from $\boldsymbol{\theta}$ to $\boldsymbol{\theta}'$ to generate MCMC samples for the posterior distribution is as follows:

$$H(\boldsymbol{\theta}|\boldsymbol{\theta}') = \frac{e^{\boldsymbol{\theta}'^T s(\boldsymbol{y})}\pi(\boldsymbol{\theta}')p(\boldsymbol{\theta}|\boldsymbol{\theta}')}{e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}\pi(\boldsymbol{\theta})p(\boldsymbol{\theta}'|\boldsymbol{\theta})} \Big/ \frac{z(\boldsymbol{\theta}')}{z(\boldsymbol{\theta})} \tag{3.2}$$

where $p(\boldsymbol{\theta}|\boldsymbol{\theta}')$ is the transition probability from $\boldsymbol{\theta}$ to $\boldsymbol{\theta}'$. The main difficulty of this ratio is the fact that it requires calculating the normalizing constant twice [46].

Møller et al. demonstrate adding an auxiliary variable to move from $(\boldsymbol{\theta}, x) \rightarrow (\boldsymbol{\theta}', x')$ and assuming the proposal density or the probability of moving to the new value is

$$p(y'|y, \boldsymbol{\theta}) = \pi(y'|\boldsymbol{\theta}') = \frac{e^{\boldsymbol{\theta}'^T s(\boldsymbol{y}')}}{z(\boldsymbol{\theta}')} \tag{3.3}$$

or the same density as the likelihood [74]. Now the new value $(\boldsymbol{\theta}', x')$ is accepted with the following probability where clearly the normalizing constants cancel out:

$$\min\left(1,\quad \frac{e^{\boldsymbol{\theta}'^T s(\boldsymbol{y}')}\pi(\boldsymbol{\theta}')\epsilon(\boldsymbol{\theta}|\boldsymbol{\theta}')e^{\boldsymbol{\theta}'^T s(\boldsymbol{y})}}{e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}\pi(\boldsymbol{\theta})\epsilon(\boldsymbol{\theta}'|\boldsymbol{\theta})e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}}\,\frac{z(\boldsymbol{\theta})z(\boldsymbol{\theta}')}{z(\boldsymbol{\theta})z(\boldsymbol{\theta}')}\right). \tag{3.4}$$

Thus the final ratio is

$$\min\left(1,\quad \frac{e^{\boldsymbol{\theta}'^T s(\boldsymbol{y}')}\pi(\boldsymbol{\theta}')\epsilon(\boldsymbol{\theta}|\boldsymbol{\theta}')e^{\boldsymbol{\theta}'^T s(\boldsymbol{y})}}{e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}\pi(\boldsymbol{\theta})\epsilon(\boldsymbol{\theta}'|\boldsymbol{\theta})e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}}\right). \tag{3.5}$$

The exchange algorithm offers a solution to avoid the need for calculating the normalizing constants in ERGM likelihoods. By utilizing multiple chains that interact with each other through parallel adaptive direction sampling described in the next section, the exchange algorithm will improve computational efficiency and enhance chain mixing performance.

3.1 Adaptive Parallel Direction Sampling Algorithm

There have been considerable developments in the approaches dealing with the problem of sampling from a distribution with a doubly intractable normalizing constant. For example, the easy-to-implement and more direct single variable exchange algorithm proposed by Murray et al. [73]. However, if there is strong temporal dependence in the state process and a strong correlation between model parameters, the exchange algorithm performs slow mixing. Caimo et al. in [10] and [12] apply the ideas of Murray et al. in [73] to increase MCMC sampling efficiency by combining delayed rejection and adaptive Monte Carlo techniques. First, a collection of H parallel Markov chains are generated. Then the next element of a current chain h is found using the differences of the estimates from two chains h_1 and h_2 such that $h_1 \neq h_2 \neq h$ scaled by a factor γ.

Algorithm: Parallel Adaptive Sampling Algorithm

```
while $i = 1, ..., N$ do
Define a scalar ADS move factor $\gamma$, for each chain $\boldsymbol{h} \in \{1, 2, 3, \cdots, H\}$:
        1. Sample two current states $h_1, h_2$ and $h_1 \neq h_2 \neq h$.
        2. Sample the error term from a symmetric normal distribution. $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \boldsymbol{\sigma}_{\epsilon}^2)$.
        3. The sampling of $\boldsymbol{\theta}_h$ performs a simple random walk:

                $\boldsymbol{\theta}_{h'} = \boldsymbol{\theta}_h + \gamma(\boldsymbol{\theta}_{h_1} - \boldsymbol{\theta}_{h_2}) + \boldsymbol{\epsilon}$

.       4. Sample $\boldsymbol{y'}$ from $\pi(\cdot|\boldsymbol{\theta}_h^{'})$.
        5. Accept $\boldsymbol{\theta}_h^{'}$ with probability

                $\min(1, \frac{q(\boldsymbol{y}|\boldsymbol{\theta}'_h)\pi(\boldsymbol{\theta}'_h)q(\boldsymbol{y}'|\boldsymbol{\theta}_h)}{q(\boldsymbol{y}|\boldsymbol{\theta}_h)\pi(\boldsymbol{\theta}_h)q(\boldsymbol{y}'|\boldsymbol{\theta}'_h)})$

      where $q(\boldsymbol{y}|\boldsymbol{\theta}) = e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}$ is the unnormalized likelihood.
end while
```

The move of $\boldsymbol{\theta}$ is illustrated in Figure (3.1). Here, two other chains h_1 and h_2 are chosen at random. The difference between the corresponding estimates in the other two chains $\boldsymbol{\theta}_{h_1}$ and $\boldsymbol{\theta}_{h_2}$ are used to find the distance to move away from $\boldsymbol{\theta}_h$. A normal distribution with a very small variance is used to slightly adjust the estimate for the new $\boldsymbol{\theta}$.

The parallel ADS move of $\boldsymbol{\theta}_h$ is generated based on the difference of the states $\boldsymbol{\theta}_{h_1}$ and $\boldsymbol{\theta}_{h_2}$ in other Markov chains and $\boldsymbol{\epsilon}$ is a random error term

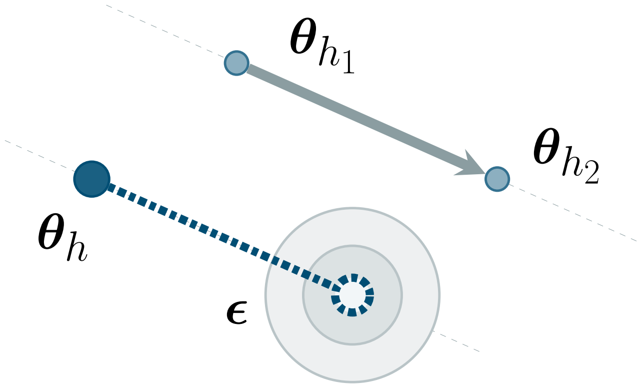


Figure 3.1: Parallel Adaptive Sampling Diagram

In this section, we will list the algorithm of the Bayesian exponential random graph model (BERGM). They set up the exchange algorithm with a Gibbs update of $\boldsymbol{\theta}'$ and then $\boldsymbol{y}'$ using Markov Chain Monte Carlo iteration without penalized terms. The algorithm can be written in the following concise way:

Algorithm: Bayesian Exponential Random Graph Model

```
while $i = 1, ..., N$ do
    while $h = 1, ..., H$ do
        1. generate $h_1$ and $h_2$ such that $h_1 \neq h_2 \neq h$
        2. generate $\boldsymbol{\theta}'_h$ from $\gamma(\boldsymbol{\theta}_{h_1} - \boldsymbol{\theta}_{h_2}) + \epsilon(\dot{}\cdot\,|\boldsymbol{\theta}_h)$
        3. simulate $y'$ from $\pi(\dot{}\cdot\,|\boldsymbol{\theta}'_h)$
        4. update $\boldsymbol{\theta}_h \rightarrow \boldsymbol{\theta}'_h$ with the log of the probability
               $\min\left(0, [\boldsymbol{\theta}_h - \boldsymbol{\theta}'_h]^T [s(\boldsymbol{y}') - s(\boldsymbol{y})] + \log\left[\frac{\pi(\boldsymbol{\theta}'_h)}{\pi(\boldsymbol{\theta}_h)}\right]\right)$
    end while
end while
```

---

where $s(\boldsymbol{y})$ and $s(\boldsymbol{y}')$ are functions of the observed and simulated vector of network statistics respectively.

CHAPTER 4
BAYESIAN ADAPTIVE LASSO EXPONENTIAL RANDOM GRAPH MODEL

With the foundation of the last chapters, we are ready for the presentation of the new Bayesian adaptive lasso exponential random graph model. This chapter will discuss the theory of the lasso penalty with its advantages in parameter selection and derive the Gibbs sampling algorithms needed to implement variable selection for network data.

4.1 Background of the Lasso Penalty

Since ERGMs have so much in common with logistic regression, let us recall the traditional lasso method in classical linear regression and discuss its development and relation to Bayesian theory so that we have some hints about the problems during developing lasso estimates on the exponential random network. In a separate chain of developments from linear regression, the least absolute shrinkage and selection operator or lasso parameter was first introduced in 1986 by Santosa et al [84]. Later, in 1996, Robert Tibshirani in his bio-statistics work in genomics brought new attention to this parameter. The lasso of Tibshirani [93] is a method for simultaneous shrinkage and model selection in regression problems. It is often applied to the linear regression model but has not been applied to the random graph. In the context of linear regression, the lasso is a regularization technique for simultaneous estimation and variable selection where if $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ where $\boldsymbol{y} = (y_1, y_2, \cdots, y_n)^\top$ is the response vector, $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_p)$ is an $n \times p$ predictor matrix, $\boldsymbol{\beta} = (\beta_1, \beta_2, \cdots, \beta_p)$

is a corresponding vector of regression coefficients, $\boldsymbol{\epsilon} = (\epsilon_1, \cdots, \epsilon_n)$ are independent normal distributed errors, then the lasso estimates are defined as

$$\hat{\beta}(lasso) = \arg\min_{\boldsymbol{\beta}} \|\boldsymbol{y} - \sum_{j=1}^{p} \boldsymbol{x_j}\beta_j\|^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{4.1}$$

where the second term in (4.1) is the so-called "l^1 penalty". The tuning parameter λ controls the amount of penalty. Fan et al. studied a class of penalized models including the lasso [23]. They proved that the lasso can perform automatic variable selection because of the singularity of l^1 penalty at the origin. If certain conditions are not satisfied, the lasso estimates could be inconsistent. Furthermore, the lasso shrinkage produces biased estimates for the large coefficients, and thus it could be suboptimal in terms of estimation risk. The asymptotic setup of the traditional lasso method is somewhat unfair because it forces the coefficients to be equally penalized in the l^2 penalty. To overcome the above issues, Zou [105] and Wang et al. [100] proposed an adaptive lasso that enjoys the consistency and the oracle properties: namely, it performs as well as if the true underlying model were given in advance. Tibshirani suggests that lasso estimates can be interpreted as posterior mode estimates when the regression parameters have independent and identical Laplace (i.e., double-exponential) priors [93]. Targeted at finding this mode, several other authors studied subsequently different Bayesian contexts. Yuan et al. studied an empirical Bayes algorithm with Laplace-like priors [104]. Park and Casella studied the practical Gibbs sampler implementation for the Bayesian Lasso and offered methods that address the choice of λ [76]. Leng et al. [62] and Alhamzawi and Ali [1] studied Bayesian Adaptive Lasso method with hierarchical Bayesian structures. However, all these studies are for linear regressions and they are not built on random networks.

4.2 Model Derivation

This work is motivated by the need to explore model uncertainty and flexibility.

With these objectives, we consider the following exponential random graph model, this model is a particular class of discrete exponential random exponential families that represent the probability distribution of the adjacency matrix $\boldsymbol{Y} \in \mathscr{Y}$ where $\mathscr{Y}$ is the set of all possible graphs on n nodes. Let $\boldsymbol{y}$ a realization of $\boldsymbol{Y}$. The likelihood function of an ERGM stands for the probability density of a random network and can be expressed as:

$$\pi(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{q(\boldsymbol{y}|\boldsymbol{\theta})}{z(\boldsymbol{\theta})} = \frac{e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}}{z(\boldsymbol{\theta})} \tag{4.2}$$

where $q(\boldsymbol{y}|\boldsymbol{\theta}) = e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}$ is the unnormalized likelihood.

We consider the following adaptive lasso estimator on the exponential random network:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\theta} l(\boldsymbol{\theta}|\boldsymbol{y}) - P(\boldsymbol{\theta}) \tag{4.3}$$

$$P(\boldsymbol{\theta}) = \sum_{j=1}^{p} \lambda_j |\theta_j| \tag{4.4}$$

where $l(\boldsymbol{\theta}|\boldsymbol{y}) = \ln(\pi(\boldsymbol{y}|\boldsymbol{\theta}))$ is the log-likelihood function of $\boldsymbol{\theta}$ and each λ_j is a different penalty parameter used for the coefficients. In dyadic independence ERGMs, maximizing the log-likelihood function (4.3) is equivalent to maximizing the following log pseudo-likelihood function:

$$l(\boldsymbol{\theta}|\boldsymbol{y}) = \sum_{\boldsymbol{y}} y_{ij} \ln(\pi_{ij}) + \sum_{\boldsymbol{y}} (1 - y_{ij}) \ln(1 - \pi_{ij}) - \sum_{j=1}^{p} \lambda_j |\theta_j| \tag{4.5}$$

where $\pi_{ij} = P(Y_{ij} = 1|\boldsymbol{y}_{ij}^c) = P(Y_{ij} = 1)$. In this case, the network estimation problems are transformed into the classical adaptive lasso logistic linear regression model. We can use LARS algorithm proposed in [20] to estimate θ_j, $j = 1, 2, 3, \cdots, p$. However, different from the generalized linear regression models, the challenge of estimation on the dyadic dependent ERGMs is that the exponential random network relies on the intractable normalizing constant appearing in the log-likelihood function. With the review of ERGMs likelihood-based methods in Chapter 2, the solution to the equation (4.3) has similar obstacles. To get around those obstacles, we will study this problem with an adaptively Bayesian estimate obtained from the lasso penalized method on the random networks.

Assume that a prior distribution $\pi(\boldsymbol{\theta})$ is placed on $\boldsymbol{\theta}$, and we are interested in the posterior distribution

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}). \tag{4.6}$$

We consider a conditional Laplace prior specification of the form similar to the classical Bayesian lasso linear regression developed in ([76]) but with different penalty terms so that we have λ_j for $j = 1, 2, 3, \cdots, p$:

$$\pi(\boldsymbol{\theta}|\sigma^2) = \prod_{j=1}^{p} \frac{\lambda_j}{2\sqrt{\sigma^2}} e^{-\lambda_j|\theta_j|/\sqrt{\sigma^2}}. \tag{4.7}$$

We can now formulate a hierarchical model on the exponential random graph, which we can use to implement this version of the Bayesian lasso with a Gibbs sampler, using the Laplace distribution as a scale mixture of Gaussians. When the mixing distribution is exponential, the resulting distribution is Laplace ([3]).

$$\frac{a}{2}e^{-a|z|} = \int_0^\infty \frac{1}{\sqrt{2\pi s}} e^{-\frac{z^2}{2s}} \frac{a^2}{2} e^{-\frac{a^2 s}{2}} ds, \quad a > 0 \tag{4.8}$$

Now we use a latent parameter τ^2 to make the prior (4.7) as a scale mixture of normal distributions (4.8). We can consider τ_js as additional parameters that assign different variances to the prior of $\boldsymbol{\theta}$. When $\tau_j \rightarrow 0$, the coefficient of $s_j(\boldsymbol{y})$ is shrunk to zero.

Assume $\boldsymbol{\theta} = (\theta_1, \theta_2, ..., \theta_p)$ follows normal distributions centered at zero with variance defined below.

$$\boldsymbol{\theta}|\sigma^2, \tau_1^2, \tau_2^2, ..., \tau_p^2 \sim \mathscr{N}(0_p,\, \sigma^2\boldsymbol{D}_\tau) \tag{4.9}$$

where $\sigma^2 > 0$ and $\boldsymbol{D}_\tau = diag(\tau_1^2, \tau_2^2, \cdots, \tau_p^2)$ is a matrix that allows each parameter to come from a normal distribution with a different variance.

Different than the basic Bayesian lasso model proposed by [76] in which $\boldsymbol{\tau}$ follows

$$\pi(\boldsymbol{\tau}^2) = \frac{\boldsymbol{\lambda}^2}{2} e^{-\frac{\boldsymbol{\lambda}^2 \boldsymbol{\tau}^2}{2}}, \tag{4.10}$$

our Bayesian adaptive lasso exponential random graph model BALERGM sets up different shrinkage parameters for different coefficients. This motivates us to define a more adaptive penalty in the hierarchical structure:

$$\pi(\sigma^2, \tau_1, \tau_2, \cdots, \tau_p | \boldsymbol{\lambda}) \propto \pi(\sigma^2) \prod_{j=1}^{p} \frac{\lambda_j^2}{2} e^{-\frac{\lambda_j^2 \tau_j^2}{2}} \tag{4.11}$$

and an independent non-informative scale-invariant marginal prior $\pi(\sigma^2) \propto \dfrac{1}{\sigma^2}$ on σ^2 suggested by Park and Casella [76]. The conditional distribution on σ^2 guarantees a unimodal full posterior distribution for the estimate $\boldsymbol{\theta}$ on the network. (See Appendix). The unimodal posterior distribution ensures the quick convergence of the Gibbs sampling algorithm and ensures the meaningful point estimate of $\boldsymbol{\theta}$.

4.3 Sampling Methods for Lambda

This paper presents three methods for sampling $\boldsymbol{\lambda}$

Method 1: The simplest prior for the penalty term λ_j, for $j = 1, 2, 3, \cdots, p$ would be a uniform distribution, but this proved to be problematic with complex networks, particularly when a model has many parameters. Thus, following the notation of Park and Casella [76], we propose an adaptive prior such that $\lambda_j^2 \sim$ Gamma (r, δ_j) so that

$$\pi(\lambda_j^2) = \frac{\delta_j^r}{\Gamma(r)} \left(\lambda_j^2\right)^{r-1} e^{-\delta_j \lambda_j^2} \qquad \text{for } \lambda_j, r, \delta_j > 0. \tag{4.12}$$

This prior mixes well with the other choices for the Gibbs sampling and as Park and Casella [76] note, this prior can approach 0 as $\boldsymbol{\lambda} \rightarrow \infty$ and can concentrate probability near the maximum likelihood estimator. This derivation is continued in Chapter 5.

Method 2: This method uses the same prior for $\boldsymbol{\lambda}$ as before:

$$\pi(\lambda_j^2) = \frac{\delta_j^r}{\Gamma(r)} \left(\lambda_j^2\right)^{r-1} e^{-\delta_j \lambda_j^2} \qquad \text{for } \lambda_j, r, \delta_j > 0. \tag{4.13}$$

In contrast to the previous method, where $\delta_j,\ j = 1, 2, \cdots, p$ were treated as fixed constants, the proposed method incorporates an empirical update of $\delta_j, j = 1, 2, \cdots, p$ based on the Expectation-Maximization (E-M) algorithm. The empirical update of $\delta_j, j = 1, 2, \cdots, p$ through the E-M algorithm is beneficial for several reasons. First, it removes the need for manual specification of the appropriate hyperparameter values. Instead, the parameter values are estimated directly from the observed data, providing a data-driven approach for hyperparameter selection. Furthermore, the empirical update of $\delta_j, j = 1, 2, \cdots, p$ allows the model to capture the nuances and complexities that may not be accounted for by method 1 with a fixed hyperparameter. This method is detailed in Chapter 6

Method 3: For the final method, $\boldsymbol{\lambda}$ is updated in entirely empirically by an E-M algorithm. For the details, see Chapter 7. While empirical Markov Chain Monte Carlo (MCMC) methods offer several advantages, such as adapting to the data and improving exploration of the parameter space, they also have certain disadvantages that should be considered.

One of the primary disadvantages of empirical MCMC is its computational cost. Empirical MCMC methods typically require additional iterations and computations compared to traditional MCMC algorithms. The empirical updates of parameters or proposal distributions can be computationally intensive, particularly when dealing with large datasets or complex models. This can result in longer execution times, limiting the scalability of the method.

Another disadvantage is the potential for bias or inefficiency in the estimation process. Empirical updates rely on the observed network data to estimate the parameters and the proposal distribution of the network. If the nodal sufficient statistics are not fully representative or the observations of nodal random variables contain outliers, the empirical estimates may introduce biases or inefficiencies in the MCMC sampling. Additionally, the convergence of method 3 needs careful tuning of the other hyperparameters to achieve optimal performance. The optimization of hyperparameters can be nontrivial and needs expert knowledge or extensive experimentation.

The first major difference of BALERGM compared with the Bayesian lasso in [76] is that the Bayesian lasso method in the work of Park and Casella [76] is applied to linear regression model $\boldsymbol{y} = \mu \mathbf{1}_n + \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ without any network structure. In other words, $\boldsymbol{y}$ in [76] follows the normal distribution $\mathscr{N}(\mu \mathbf{1}_n + \boldsymbol{X}\boldsymbol{\beta}, \sigma^2 \boldsymbol{I}_n)$, where $\boldsymbol{y}$ is a $n \times 1$ vector of responses which does not involve any random graph. Second, our model allows different penalty variables λ_j, one for each different parameter. In this case, each τ_j^2 can have its own distribution and thus the variance of each normal distribution can be different. With the flexibility of the penalties, the lasso estimate of the parameter for less important random variables on the exponential random graph will have a larger penalty. A smaller penalty will be applied to those important random variables. Compared with the existing Bayesian Adaptive Lasso model [62][1], our model is built on the random network. Additionally compared with the Bayesian Exponential Random Graph Model (BERGM) by [10], our model Bayesian Adaptive Lasso Exponential Random Graph Model (BALERGM) has more accurate estimations, and the structure is more flexible and adaptive to the network statistics level by adopting distinct shrinkage and penalties for different network statistics. The estimates $\hat{\theta}_j$ of θ_j for $j = 1, 2, 3, \cdots, p$ will be small and close to 0 if it does not provide much improvement on predicting the random network $\boldsymbol{Y}$. So it naturally leads to an estimator with an automatic variable selection property. The value of λ_j will affect the estimates θ_j. The larger $\hat{\lambda}_j$ exists in the model, the sparser $\boldsymbol{\theta}$ will be. (namely, more coefficients are small and near 0) whereas smaller $\hat{\theta}_j$ leads to a less sparse $\boldsymbol{\theta}$. Sparsity is a common expectation in high-dimensional statistics because we anticipate only a few covariates are actually related to the response and most covariates are useless. BALERGM is very powerful in this scenario because it leads to a sparse estimator on the network (many coefficients are near 0). Note that high-dimensional problems in network science are very common. For example, in genetics, there are many genes per individual but often we have few patients in our study, or in neuroscience, the fMRI machine produces many voxels per person at a given time.

4.4 Unimodalilty of Posterior Distribution

To ensure faster convergence of Gibbs sampling and have confidence in the estimates obtained, it is crucial to select a prior distribution that leads to a unimodal posterior distribution.

When the prior distribution is unimodal, it means that the prior assigns the highest probability to a single mode or peak of the parameter space. This is desirable because it indicates a clear preference or belief in a particular range of parameter values. When the prior is unimodal and aligns with the true underlying distribution, it increases the likelihood of the posterior distribution also being unimodal. Having a unimodal posterior distribution is advantageous for Gibbs sampling convergence. Gibbs sampling relies on updating each parameter in turn, conditioned on the values of the other parameters. When the posterior distribution is unimodal, the updates tend to move the parameter values toward the mode of the distribution. This promotes efficient exploration of the parameter space and faster convergence to the most probable values. The following theorem shows BERGM has a unimodal posterior distribution for faster convergence of Gibbs sampling and provides more confidence in the estimates obtained.

Theorem 1 *The joint posterior distribution is unimodal for typical choices of* $\pi(\sigma^2)$ *and any choice of* $\lambda \geq 0$.

Proof:

We begin by representing the joint distribution of θ and $\sigma^2 > 0$ using distributions already defined.

$$\pi(\boldsymbol{\theta}, \sigma^2) \propto \pi(\boldsymbol{\theta}|\sigma^2)\pi(\sigma^2) \tag{4.14}$$

$$= \prod_{j=1}^{p} \frac{\lambda_j}{2\sqrt{\sigma^2}} e^{-\lambda_j|\theta_j|/\sqrt{\sigma^2}} \frac{1}{\sigma^2} \tag{4.15}$$

We have chosen the prior such that $\pi(\sigma^2) \propto \frac{1}{\sigma^2}$ according to the recommendation of the literature [76].

We wish to show that the posterior is unimodal in the sense that every upper-level set of $\{(\boldsymbol{\theta}, \sigma^2)|\pi(\boldsymbol{\theta}, \sigma^2) > x, \sigma^2 > 0\}$, for $x > 0$ is connected. We will show this is true under a continuous transform with continuous inverse since the continuous image of a connected set is connected [72].

The posterior is shown here:

$$\pi(\boldsymbol{\theta}, \sigma^2|\boldsymbol{y}) \propto \pi(y|(\boldsymbol{\theta}, \sigma^2))\pi(\boldsymbol{\theta}, \sigma^2) \tag{4.16}$$

$$= \pi(\boldsymbol{y}|(\boldsymbol{\theta}, \sigma^2))\pi(\boldsymbol{\theta}|\sigma^2)\pi(\sigma^2) \tag{4.17}$$

$$= \frac{1}{z(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^T s(\boldsymbol{y})} \prod_{j=1}^{p} \frac{\lambda_j}{2\sqrt{\sigma^2}} e^{-\lambda_j|\theta_j|/\sqrt{\sigma^2}} \frac{1}{\sigma^2} \tag{4.18}$$

$$= \frac{e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}}{z(\boldsymbol{\theta})} \frac{1}{\sigma^2} \frac{1}{2^p \sqrt{\sigma^2}^p} \prod_{j=1}^{p} \lambda_j e^{-\lambda_j|\theta_j|/\sqrt{\sigma^2}} \tag{4.19}$$

We now take the natural log of the equation above.

$$\ln \pi(\boldsymbol{\theta}, \sigma^2|y) = -\ln(\sigma^2) + \boldsymbol{\theta}^T s(\boldsymbol{y}) - \sum_{j=1}^{p} \lambda_j|\theta_j| \frac{1}{\sqrt{\sigma^2}} + \sum_{j=1}^{p} \ln(\lambda_j) - p\ln(2) - \frac{p}{2}\ln(\sigma^2) \tag{4.20}$$

The following transform allows for easier calculation.

$$\phi_j \leftrightarrow \frac{\theta_j}{\sqrt{\sigma^2}} \qquad \rho \leftrightarrow \frac{1}{\sqrt{\sigma^2}} \qquad j = 1, 2, 3, ...p$$

This is continuous with a continuous inverse when $0 < \sigma^2 < \infty$, so the upper-level sets for the original parameters correspond under the transformation to upper-level sets for the original parameters. Let $\boldsymbol{\phi} = (\phi_1, \phi_2, ...\phi_p)^T$ be the column vector for ease of notation. This transform is one-to-one and continuous for $0 < \sigma^2 < \infty$, therefore the unimodality of the transformed equation is equivalent to the unimodality of the original equation.

Using the transform and algebra we get the following expression

$$\begin{aligned} h(\boldsymbol{\phi}, \rho) &= \ln \rho^2 + (\sqrt{\sigma^2}\boldsymbol{\phi})^T s(\boldsymbol{y}) - \sum_{j=1}^{p} \lambda_j |\phi_j| + \frac{p}{2} \ln(\rho^2) \\ &= (p+2)\ln(\rho) + \frac{\boldsymbol{\phi}^T s(\boldsymbol{y})}{\rho} - \sum_{j=1}^{p} \lambda_j |\phi_j|. \end{aligned} \tag{4.21}$$

We can show that (4.21) is unimodal by showing it is a concave function in $(\boldsymbol{\phi}, \rho)$. We will do that by considering each term of the equation in turn.

$$h_1 = \ln(\rho), \qquad h_2 = \frac{\boldsymbol{\phi}^T s(\boldsymbol{y})}{\rho}, \qquad h_3 = -\sum_{j=1}^{p} \lambda_j |\phi_j|.$$

We will determine the concavity of the first two functions by checking the spectral property of the corresponding Hessian matrix.

$$H_{h_i} = \begin{bmatrix} \frac{\partial^2 h_i}{\partial \phi^2} & \frac{\partial^2 h_i}{\partial \boldsymbol{\phi} \partial \rho} \\ \frac{\partial^2 h_i}{\partial \rho \partial \boldsymbol{\phi}} & \frac{\partial^2 h_i}{\partial \rho^2} \end{bmatrix}, i = 1, 2. \tag{4.22}$$

For the first term $h_1 = \ln(\rho)$ and the second term $h_2 = \frac{\boldsymbol{\phi}^T s(\boldsymbol{y})}{\rho}$, the corresponding Hessian matrix H_{h_1} and H_{h_2} are both negative semi-definite and thus h_1 and h_2 are both concave in $(\boldsymbol{\phi}, \rho)$.

For the third term $h_3 = -\sum_{j=1}^{p} \lambda_j |\phi_j|$, we see this is a sum of the negative of a constant times an absolute value function. This is a concave function in $\boldsymbol{\phi}, \rho$, since the j the term in h_3 is $h_3(j) = -\lambda_j|\phi_j|$ which is a concave function of ϕ_j and the sum of concave functions is a concave function.

Using the same reasoning that the sum of concave functions is concave gives that (4.21) is concave, and hence the posterior distribution is concave.

Therefore, we can conclude that our posterior distribution is unimodal.

$\square$

With the pieces needed for the Gibbs sampling, the next three chapters present the three variations of this model: full Bayes, partial empirical, and full empirical. The full Bayes method updates each parameter using the distributions found in this chapter. The partial empirical method updates the single parameter $\boldsymbol{\delta}$ for the parameterization of the gamma distribution for the $\boldsymbol{\lambda}$ update. Finally, the full empirical method updates the penalty $\boldsymbol{\lambda}$ in a fully empirical way.

Chapter 8 discusses the advantages and demonstrates the effectiveness of these models on different data sets.

CHAPTER 5
BAYESIAN ADAPTIVE LASSO ERGM - FULL BAYES

Drawing upon the theoretical hierarchical structure outlined in Chapter 4, we present a comprehensive Bayesian Adaptive Lasso algorithm with a full Bayes method. We lay out the chosen distributions in the hierarchical model in Section 5.1. Using these distributions, we combine them for the joint distribution in the following section. Finally, Sections 5.3 and 5.4 present the sampling distributions of the Gibbs sampling algorithm.

5.1 Hierarchical Model

Using the distributions defined in the previous chapter, we summarize the full Bayes hierarchical model (method 1) below.

$$\pi(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{1}{z(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^T s(\boldsymbol{y})} \tag{5.1}$$

$$\boldsymbol{\theta}|\sigma^2, \tau_1^2, \tau_2^2, ..., \tau_p^2 \sim \mathcal{N}(0_p,\, \sigma^2 \boldsymbol{D}_\tau) \tag{5.2}$$

$$\boldsymbol{D}_\tau = diag(\tau_1^2, \cdots, \tau_p^2) \tag{5.3}$$

$$\pi(\sigma^2, \tau_1, \tau_2, \cdots, \tau_p|\boldsymbol{\lambda}) \propto \pi(\sigma^2) \prod_{j=1}^{p} \frac{\lambda_j^2}{2} e^{-\frac{\lambda_j^2 \tau_j^2}{2}} \tag{5.4}$$

$$\pi(\lambda_j^2) = \frac{\delta_j^r}{\Gamma(r)} \left(\lambda_j^2\right)^{r-1} e^{-\delta_j \lambda_j^2} \tag{5.5}$$

$$\pi(\sigma^2) \propto \frac{1}{\sigma^2} \tag{5.6}$$

for σ^2, r, δ_j and $\tau_1^2, \tau_2^2, \cdots, \tau_p^2 > 0$.

Here the likelihood $\pi(\boldsymbol{y}|\boldsymbol{\theta})$ is the distribution from the ERGM. We find the conditional distribution of $\boldsymbol{\theta}|\sigma^2, \boldsymbol{\tau}^2$ as a mixture of normal distributions with individualized variances τ^2 each following an exponential distribution as seen in the fourth line. These variances are conveniently notated as the diagonal matrix $\boldsymbol{D}_\tau$. The distribution of $\boldsymbol{\lambda}$ and σ^2 are discussed in more detail in the previous chapter.

5.2 Joint Distribution

Now we will implement the model with a Gibbs sampler. The Gibbs sampling method is a Monte Carlo Markov Chain (MCMC) algorithm. In our case, the joint distribution is difficult to sample from directly, but the conditional distribution of each variable is known and is easier to sample from. The Gibbs sampling algorithm generates an instance from the distribution of each variable in turn, conditioned on the current values of the other variables. The construction of the hierarchical model (5.1) makes the derivation of the full conditional distributions for each component of the estimates feasible.

Thus we can write the joint density as the product of the conditional density of $\boldsymbol{y}|\boldsymbol{\theta}$ and the density of $\boldsymbol{\theta}$. Using the pieces of the hierarchical formulation of the model from (5.1) we can substitute in each piece that we have already chosen to find the joint distribution.

$$\begin{aligned}
&\pi(\boldsymbol{y}, \boldsymbol{\theta}, \sigma, \boldsymbol{\lambda}, \boldsymbol{\tau}) \\
&= \pi(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \\
&= \pi(\boldsymbol{y}|\boldsymbol{\theta}) \prod_{j=1}^{p} \pi(\theta_j|\tau_j^2, \sigma^2)\pi(\tau_j^2|\lambda_j)\pi(\lambda_j)\pi(\sigma^2) \\
&= \frac{1}{z(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^T s(\boldsymbol{y})} \prod_{j=1}^{p} \frac{1}{(2\sigma^2\tau_j^2)^{1/2}} e^{-\frac{1}{2\sigma^2\tau_j^2}\theta_j^2} \frac{\lambda_j^2}{2} e^{-\frac{\lambda_j^2\tau_j^2}{2}} \frac{\delta_j^r}{\Gamma(r)} \left(\lambda_j^2\right)^{r-1} e^{-\delta_j\lambda_j^2} \frac{1}{\sigma^2}
\end{aligned} \tag{5.7}$$

5.3 Gibbs Sampling Implementation

To implement the Gibbs sampling, we require the sampling distribution of each parameter $\tau_j, \lambda_j, \sigma^2$ to update in turn. From the joint distribution (5.7), we consider all parts of that joint distribution that depend on each variable.

As summarized in Table 5.1, we consider the full conditional distributions for $\tau_j, \lambda_j,$ and σ^2 respectively.

Table 5.1: Sampling distributions from joint distribution for each variable

Variable	**Proportional Distribution**
$\frac{1}{\tau_j^2}$	Inverse Gaussian $\left(\sqrt{\frac{\lambda_j^2\sigma^2}{\theta_j^2}}, \lambda_j^2\right)$
λ_j	Gamma $\left(2, \frac{\tau_j^2}{2}\right)$
σ^2	Inverse Gamma $\left(\frac{p}{2}, \frac{1}{2}\boldsymbol{\theta}^T D_{\boldsymbol{\tau}}^{-1}\boldsymbol{\theta}\right)$

Sample τ_j

For each τ_j we have the following distribution.

$$\pi(\tau_j|\boldsymbol{y}, \boldsymbol{\theta}, \sigma, \boldsymbol{\lambda}) \propto (\tau_j^2)^{\frac{-1}{2}} e^{-\frac{1}{2}\left(\frac{\theta_j^2/\sigma^2}{\tau_j^2}+\lambda_j^2\tau_j^2\right)} \tag{5.8}$$

To find what distribution each τ_j follows, we begin by considering the following transformation [16]. If a random variable $x \sim$ Inverse Gaussian(μ, λ'), that is

$$f(x, \mu, \lambda') = \left(\frac{\lambda'}{2\pi x^3}\right)^{\frac{1}{2}} e^{-\frac{\lambda'(x-\mu)^2}{2\mu^2 x}}, \tag{5.9}$$

then with the change of variable, we can find the density f' of $w = x^{-1}$ as

$$f(w, \mu, \lambda') = \left(\frac{\lambda'}{2\pi w^3}\right)^{\frac{1}{2}} e^{-\frac{\lambda'(1-\mu w)^2}{2\mu^2 w}}. \tag{5.10}$$

Hence

$$f'(w, \mu, \lambda') = \mu w f(w, \mu^{-1}, \lambda' \mu^{-2}). \tag{5.11}$$

Then we can rewrite equation (5.8) into the reciprocal of the Inverse Gaussian distribution

$$\left(\frac{1}{\tau_j^2}\right)^{-\frac{3}{2}} exp\left\{-\frac{1}{2}\left(\frac{\theta_j^2}{\tau_j^2} + \frac{\lambda_j^2}{1/\tau_j^2}\right)\right\} \propto \left(\frac{1}{\tau_j^2}\right)^{-\frac{3}{2}} exp\left\{-\frac{\theta_j^2\left(\frac{1}{\tau_j^2} - \sqrt{\frac{\lambda_j^2\sigma^2}{\theta_j^2}}\right)^2}{2\sigma^2\frac{1}{\tau_j^2}}\right\} \tag{5.12}$$

thus $\frac{1}{\tau_j^2}$ follows inverse Gaussian distribution with parameters $\sqrt{\frac{\lambda_j^2\sigma^2}{\theta_j^2}}$ and λ_j^2.

Sample λ_j^2

For each λ_j^2 we have to find the following distribution.

$$\pi(\lambda_j^2|\boldsymbol{y}, \boldsymbol{\theta}, \sigma, \boldsymbol{\tau}) \propto \frac{\lambda_j^2}{2} e^{-\frac{\lambda_j^2\tau_j^2}{2}} \tag{5.13}$$

This shows us that λ_j^2 is proportional to a gamma distribution with $\alpha = 2$ and $\beta = \frac{\tau_j^2}{2}$, since a standard gamma probability density function is $f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$. Therefore we can conclude:

$$\pi(\lambda_j^2|\boldsymbol{y}, \boldsymbol{\theta}, \sigma, \boldsymbol{\tau}) \propto \text{Gamma}(2, \frac{\tau_j^2}{2}) \tag{5.14}$$

Sample σ^2

Similar to the other distributions, we now look at σ^2.

$$\pi(\sigma^2|\boldsymbol{y}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\tau}) \propto (\sigma^2)^{-1-\frac{p}{2}} e^{-\frac{1}{2\sigma^2}\boldsymbol{\theta}^T \mathrm{D}_{\boldsymbol{\tau}}^{-1}\boldsymbol{\theta}} \tag{5.15}$$

If $x \sim$ Inverse Gamma (α, β) with the shape parameter α and scale parameter β, then it has the following density function:

$$f(x, \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} e^{-\frac{\beta}{x}}. \tag{5.16}$$

We can compare the conditional density (5.15) with (5.16) to find:

$$\pi(\sigma^2|\boldsymbol{y},\boldsymbol{\theta},\boldsymbol{\lambda},\boldsymbol{\tau}) \propto \text{Inverse Gamma } \left(\frac{p}{2},\frac{1}{2}\boldsymbol{\theta}^T D_{\boldsymbol{\tau}}^{-1}\boldsymbol{\theta}\right). \tag{5.17}$$

5.4 Full Bayes Algorithm

For the new Bayesian Adaptive Lasso model, we use the parallel adaptive direction sampler method suggested by BERGM from Chapter 3 and combine that with Gibbs sampling theory from Chapter 4 to generate samples of $\boldsymbol{\theta}$.

Algorithm: Bayesian Adaptive Lasso Exponential Random Graph Model Algorithm

```
Require: Set the initial value for $\boldsymbol{\lambda}, \sigma^2, \gamma$, Use ERGM to find MPLE (Maximizer to the Psuedolikelihood Function) to find initial values for $\boldsymbol{\theta}$. Denote samples of $\boldsymbol{\theta}$ in the $h$th chain, as $\boldsymbol{\theta}_h$.
while $i = 1, ..., N$ do
    while $h = 1, ..., H$ do
        1. sample $\boldsymbol{\theta}_h$ with Parallel Adaptive Direction Sampler:
            a. generate $h_1$ and $h_2$ such that $h_1 \neq h_2 \neq h$
            b. update $\boldsymbol{D}_{\boldsymbol{\tau}}^{-1}$
            c. generate $\boldsymbol{\theta}'_h$ from $\gamma(\boldsymbol{\theta}_{h_1} - \boldsymbol{\theta}_{h_2}) + \epsilon(\dot{}\cdot\cdot\,|\boldsymbol{\theta}_h)$
            d. simulate $\boldsymbol{y}'$ from $\pi(\dot{}\cdot\cdot\,|\boldsymbol{\theta}'_h)$
            e. update $\boldsymbol{\theta}_h \to \boldsymbol{\theta}'_h$ with the log of the probability
                $$\min\left(0, [\boldsymbol{\theta}_h - \boldsymbol{\theta}'_h]^T[s(\boldsymbol{y}') - s(\boldsymbol{y})] + \log\left[\frac{\pi(\boldsymbol{\theta}'_h)}{\pi(\boldsymbol{\theta}_h)}\right]\right)$$
            where $\pi(\boldsymbol{\theta}) \sim \mathcal{N}(0_p,\, \sigma^2\boldsymbol{D}_{\boldsymbol{\tau}})$
        2. sample $\sigma^2$ by generating a sample from Inverse Gaussian($\frac{p}{2}, -\frac{1}{2}\boldsymbol{\theta}^T\boldsymbol{D}_{\boldsymbol{\tau}}^{-1}\boldsymbol{\theta}$)
        3. sample $\tau_j^2$ for $j = 1, 2, 3, ..$ by generating a sample from
                $$\text{Inverse Gaussian } \left(\sqrt{\frac{\lambda_j^2\sigma^2}{\theta_j^2}}, \lambda_j^2\right)$$
        4. sample $\lambda_j^2$ for $j = 1, 2, 3, ..$ by generating a sample from Gamma $(2, \frac{\tau_j^2}{2})$
    end while
end while
```

This code was built with R version 4.1.1 (2021-08-10). [79] The following package versions were also used: coda 0.19-4, mcmc 0.9-7, Bergm 5.0.3, ergm.count 4.0.2, ergm 4.1.2, mvtnorm 1.1-3.

The full Byes method presented in this chapter provides a method for understanding networks in a powerful way. Chapter 8 provides examples of the effectiveness of this critical model.

CHAPTER 6
BAYESIAN ADAPTIVE LASSO ERGM - DELTA EMPIRICAL

For this chapter, we lay out the theory for the empirical update for one of the parameters for the penalty term $\boldsymbol{\lambda}$. This modification to the algorithm in Chapter 4 allows for the specification of the gamma parameters for the distribution of $\boldsymbol{\lambda}$. This modification is particularly necessary since the R code generating samples from the gamma and inverse gamma distributions is very sensitive to parameter specification. Incorrect initial parameter conditions or complex model structures can result in extreme values that can become NAs in the gamma function in R leading to models that fail to converge.

6.1 Empirical Derivation

The Monte Carlo Expectation-maximization algorithm for empirical Bayes estimation of hyperparameters proposed by [63] essentially treats the parameters as missing data and then uses the E-M algorithm to iteratively approximate the hyperparameters substituting Monte Carlo estimates for any expected values that cannot be computed explicitly. For BALERGM, the Gibbs sampler is used to estimate the expected values.

To begin this process, we consider the part of the joint distribution that depends on $\boldsymbol{\delta}$, since when taking the derivative all other terms will become zero.

$$\pi(\boldsymbol{y}, \boldsymbol{\theta}, \boldsymbol{\delta}) = \frac{\delta_j^r}{\Gamma(r)} \left(\lambda_j^2\right)^{(r-1)} e^{-\delta_j \lambda_j^2} + \text{ terms not involving } \delta_j^2 \tag{6.1}$$

We then take the natural log of the resulting equation.

$$\ln(\delta_j|\boldsymbol{y},\boldsymbol{\theta}) \propto r\ln(\delta_j) - \delta_j\lambda_j^2 \tag{6.2}$$

1. Expectation step

$$\begin{align}
Q(\delta_j|\delta_j^{(k-1)},y^{(k-1)}) &= \mathbb{E}_{\delta^{(k-1)}}\left[\ln(\delta_j|\boldsymbol{y},\boldsymbol{\theta})|\delta_j^{(k-1)},y^{(k-1)}\right] \tag{6.3}\\
&= r\ln(\delta_j) - \delta_j\mathbb{E}\left[\lambda_j^2|\delta_j^{(k-1)},y^{(k-1)}\right] + \text{ terms without } \delta_j \tag{6.4}
\end{align}$$

2. Maximization step

$$\begin{align}
\delta_j^{(k)} &= \arg\max_{\delta_j} Q(\delta_j|\delta_j^{(k-1)},\boldsymbol{y}^{(k-1)}) \tag{6.5}\\
\frac{\partial Q}{\partial \delta_j} &= \frac{r}{\delta_j} - \mathbb{E}\left[\lambda_j^2|\delta_j^{(k-1)},y^{(k-1)}\right] = 0 \tag{6.6}
\end{align}$$

Now, solving for δ_j gives the expression for estimating this parameter.

$$\delta_j = \frac{r}{\mathbb{E}\left[\lambda_j^2|\delta_j^{(k-1)},y^{(k-1)}\right]} \tag{6.7}$$

The sample mean can then be used to approximate the given expected value for λ_j^2.

6.2 Empirical Delta Algorithm

With the empirical update of $\boldsymbol{\delta}$, we can consider the previous 5 estimates of $\boldsymbol{\lambda}$ as representative of the expected value of $\boldsymbol{\lambda}$.

Algorithm: Bayesian Adaptive Lasso Exponential Random Graph Model Algorithm

---

```
Require: Set the initial value for $\boldsymbol{\lambda}, \sigma^2, \gamma$, Use ERGM to find MPLE (Maximizer to the Psuedolikelihood Function) to find initial values for $\boldsymbol{\theta}$. Denote samples of $\boldsymbol{\theta}$ in the $h$th chain, as $\boldsymbol{\theta}_h$.
while $i = 1, ..., N$ do
    while $h = 1, ..., H$ do
        1. sample $\boldsymbol{\theta}_h$ with Parallel Adaptive Direction Sampler:
            a. generate $h_1$ and $h_2$ such that $h_1 \neq h_2 \neq h$
            b. update $\boldsymbol{D}_{\boldsymbol{\tau}}^{-1}$
            c. generate $\boldsymbol{\theta}'_h$ from $\gamma(\boldsymbol{\theta}_{h_1} - \boldsymbol{\theta}_{h_2}) + \epsilon(\cdot\cdot\,|\boldsymbol{\theta}_h)$
            d. simulate $\boldsymbol{y}'$ from $\pi(\cdot\cdot\,|\boldsymbol{\theta}'_h)$
            e. update $\boldsymbol{\theta}_h \to \boldsymbol{\theta}'_h$ with the log of the probability
                $$\min\left(0, [\boldsymbol{\theta}_h - \boldsymbol{\theta}'_h]^T[s(\boldsymbol{y}') - s(\boldsymbol{y})] + \log\left[\frac{\pi(\boldsymbol{\theta}'_h)}{\pi(\boldsymbol{\theta}_h)}\right]\right)$$
           where $\pi(\boldsymbol{\theta}) \sim \mathscr{N}(0_p,\, \sigma^2\boldsymbol{D_\tau})$
        2. sample $\sigma^2$ by generating a sample from Inverse Gaussian$(\frac{p}{2}, -\frac{1}{2}\boldsymbol{\theta}^T\boldsymbol{D_\tau}^{-1}\boldsymbol{\theta})$
        3. sample $\tau_j^2$ for $j = 1, 2, 3, ..$ by generating a sample from
                Inverse Gaussian $\left(\sqrt{\frac{\lambda_j^2\sigma^2}{\theta_j^2}}, \lambda_j^2\right)$
        4. empirical update of $\boldsymbol{\delta}$ and update of $\boldsymbol{\lambda}$
            a. update $\boldsymbol{\delta}$ with the following
                $$\delta_j = \frac{r}{\mathbf{E}_{\delta_j^{k-1}}\left[\lambda_j^2|\delta_j^{(k-1)}, y^{(k-1)}\right]}$$
            estimating the expected value by the mean of the last five $\boldsymbol{\lambda}$ samples.
            b. sample $\lambda_j^2$ for $j = 1, 2, 3, ..$ by generating a sample from
                Gamma $(r + 1, \frac{\tau_j^2}{2} + \boldsymbol{\delta})$
    end while
end while
```

This code was built with R version 4.1.1 (2021-08-10) [79]. The following package versions were also used: coda 0.19-4, mcmc 0.9-7, Bergm 5.0.3, ergm.count 4.0.2, ergm 4.1.2, mvtnorm 1.1-3.

Since the exact specification of the hyperparameters for the specification of the $\boldsymbol{\lambda}$ penalty term can be difficult, the partial empirical method presented in this chapter allows for a more flexible model. Chapter 8 provides examples and applications of this model.

CHAPTER 7
BAYESIAN ADAPTIVE LASSO ERGM - LAMBDA EMPIRICAL

The final method of estimating the penalty term $\boldsymbol{\lambda}$ is a full empirical method. Here using methods similar to the previous chapter, we find that $\boldsymbol{\lambda}$ can be estimated from the averages of the generated samples.

7.1 Empirical Derivation

The empirical process of estimating λ_j begins with the joint distribution terms that depend on λ_j.

$$\pi(\boldsymbol{y}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \sigma^2, \boldsymbol{\tau} | \boldsymbol{y}, s(\boldsymbol{y})) \tag{7.1}$$

$$\propto \pi(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}|\sigma^2, \boldsymbol{\tau}) \prod_{j=1}^{p} \pi(\sigma^2, \boldsymbol{\tau}|\lambda_j^2)\pi(\lambda_j^2 \tag{7.2}$$

$$= \frac{e^{\boldsymbol{\theta}^{\boldsymbol{T}} s(\boldsymbol{y})}}{z(\boldsymbol{\theta})} \prod_{j=1}^{p} \frac{1}{\sqrt{2\pi j^2}} exp\left\{-\frac{1}{2\tau_j^2}\theta_j^2\right\} \frac{\lambda_j^2}{2} \exp\left\{-\frac{\lambda_j^2 \tau_j^2}{2}\right\} \frac{\delta_j^r}{\Gamma(r)} \left(\lambda_j^2\right)^{r-1} e^{-\delta_j \lambda_j^2} \tag{7.3}$$

Next, we take the natural log:

$$\ln \pi(\boldsymbol{y}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \sigma^2, \boldsymbol{\tau} | \boldsymbol{y}, s(\boldsymbol{y})) = \sum_{j=1}^{p} \left[r \ln(\lambda_j^2) - \lambda_j^2 \left(\frac{\tau_j^2}{2} + \delta_j \right) \right] + \text{ terms not involving } \boldsymbol{\lambda} \tag{7.4}$$

Using the justification from Section 6.1, we repeat the same Expectation-Maximization method beginning by deriving the expectation step.

1. Expectation step

$$\begin{aligned} Q(\lambda|\lambda^{(k-1)},\boldsymbol{y}^{(k-1)}) &= \mathbb{E}_{\lambda^{(k-1)}}\left[\ln \pi(\boldsymbol{y},\boldsymbol{\theta},\boldsymbol{\lambda},\sigma^2,\boldsymbol{\tau}|\boldsymbol{y},s(\boldsymbol{y}))|\lambda^{(k-e)},\boldsymbol{y}^{(k-e)}\right] \\ &= \sum_{j=1}^{p} r\ln(\lambda_j^2) - \sum_{j=1}^{p}\lambda_j^2\left(\mathbb{E}_{\lambda^{(k-1)}}\left[\tau_j^2|\boldsymbol{y}^{(k-1)},\lambda^{(k-1)}\right]+\delta_j\right) \qquad (7.5) \\ &\quad + \text{ terms not involving } \boldsymbol{\lambda}. \end{aligned}$$

Now, we find the second component with the maximization step.

2. Maximization step

$$\boldsymbol{\lambda}^{(k)} = \arg\max_{\boldsymbol{\lambda}} Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(k-1)},\boldsymbol{y}^{(k-1)}). \qquad (7.6)$$

Thus

$$\begin{aligned} \frac{\partial Q}{\partial \lambda_j} &= \frac{2r}{\lambda_j} - 2\lambda_j\left(\mathbb{E}_{\boldsymbol{\lambda}^{(k-1)}}\left[\tau_j^2|\boldsymbol{y}^{(k-1)},\lambda^{(k-1)}\right]+\delta_j\right) \qquad (7.7) \\ &= 2r - 2\lambda_j^2\left(\mathbb{E}_{\boldsymbol{\lambda}^{(k-1)}}\left[\tau_j^2|\boldsymbol{y}^{(k-1)},\lambda^{(k-1)}\right]+\delta_j\right) = 0 \qquad (7.8) \end{aligned}$$

We can now solve for λ_j^2.

$$\lambda_j^2 = \frac{r}{\mathbb{E}_{\boldsymbol{\lambda}^{(k-1)}}\left[\tau_j^2|\boldsymbol{y}^{(k-1)},\lambda^{(k-1)}\right]+\delta_j}. \qquad (7.9)$$

To estimate the conditional expectations of τ_j^2, we compute the sample averages of the quantities of interest using the obtained samples from the Gibbs sampler. Specifically, for each iteration, we calculate the relevant quantity based on the current parameter values and estimates of $\lambda^{(k-1)}$ in the previous iteration. By averaging these quantities across the iterations, we obtain an estimate of the conditional expectations of τ_j^2.

7.2 Full Empirical Algorithm

Using the above theory, we present the final method of the Bayesian adaptive lasso algorithm.

Algorithm: Bayesian Adaptive Lasso Exponential Random Graph Model Algorithm

```
Require: Set the initial value for $\boldsymbol{\lambda}, \sigma^2, \gamma$, Use ERGM to find MPLE (Maximizer to the Psuedolikelihood Function) to find initial values for $\boldsymbol{\theta}$. Denote samples of $\boldsymbol{\theta}$ in the $h$th chain, as $\boldsymbol{\theta}_h$.
while $i = 1, ..., N$ do
    while $h = 1, ..., H$ do
        1. sample $\boldsymbol{\theta}_h$ with Parallel Adaptive Direction Sampler:
            a. generate $h_1$ and $h_2$ such that $h_1 \neq h_2 \neq h$
            b. update $\boldsymbol{D}_{\boldsymbol{\tau}}^{-1}$
            c. generate $\boldsymbol{\theta}'_h$ from $\gamma(\boldsymbol{\theta}_{h_1} - \boldsymbol{\theta}_{h_2}) + \epsilon(\ddots|\boldsymbol{\theta}_h)$
            d. simulate $\boldsymbol{y}'$ from $\pi(\ddots|\boldsymbol{\theta}'_h)$
            e. update $\boldsymbol{\theta}_h \rightarrow \boldsymbol{\theta}'_h$ with the log of the probability
                $$\min\left(0, [\boldsymbol{\theta}_h - \boldsymbol{\theta}'_h]^T[s(\boldsymbol{y}') - s(\boldsymbol{y})] + \log\left[\frac{\pi(\boldsymbol{\theta}'_h)}{\pi(\boldsymbol{\theta}_h)}\right]\right)$$
           where $\pi(\boldsymbol{\theta}) \sim \mathcal{N}(0_p, \sigma^2\boldsymbol{D}_{\boldsymbol{\tau}})$
        2. sample $\sigma^2$ by generating a sample from Inverse Gaussian$(\frac{p}{2}, -\frac{1}{2}\boldsymbol{\theta}^T\boldsymbol{D}_{\boldsymbol{\tau}}^{-1}\boldsymbol{\theta})$
        3. sample $\tau_j^2$ for $j = 1, 2, 3, ..$ by generating a sample from
                $$\text{Inverse Gaussian}\left(\sqrt{\frac{\lambda_j^2\sigma^2}{\theta_j^2}}, \lambda_j^2\right)$$
        4. update $\boldsymbol{\lambda}$ with the sample mean of $\boldsymbol{\tau}$ estimating the expected value
                $$\lambda_j^2 = \frac{r}{\mathbf{E}_{\boldsymbol{\lambda}_j^{k-1}}\left[\frac{\tau_j^2}{2}|\lambda_j^{(k-1)}, y^{(k-1)}\right] + \boldsymbol{\delta}}$$
    end while
end while
```

This code was built with R version 4.1.1 (2021-08-10). [79] The following package versions were also used: coda 0.19-4, mcmc 0.9-7, Bergm 5.0.3, ergm.count 4.0.2, ergm 4.1.2, mvtnorm 1.1-3.

As discussed in the next chapter, the fully empirical method conveniently removes the need to specify any hyperparameters for the $\boldsymbol{\lambda}$ penalty terms. This flexibility is key for complex networks.

CHAPTER 8
BALERGM APPLICATIONS AND SIMULATIONS

This chapter presents examples of the effectiveness of the new penalized models of the last four chapters. The first two sections compare the model convergence and parameter selection. The final section uses data collected in a recent study to show the parameter selection abilities of this new penalized Bayesian exponential random graph model.

8.1 Simulation

This first comparison of the old BERGM and three new BALERGM methods uses the Faux Dixon High School data set to demonstrate the usefulness of BALERGM.

8.1.1 Data Description

The network object Faux Dixon High represents a friendship network among junior high and high school students based on data gathered by a National Longitudinal Study of Adolescent Health. [80] This study, first conducted in 1994-1995, considered more than 90,0000 American students. Students were asked to list friends, and a tie is formed between them in the network if both students claimed friendship. [35] To preserve confidentiality, the school data was fit to a model; then the final network was obtained by simulating from that model.

Generated in R, this plot shows the clustering of student friendship with stu-

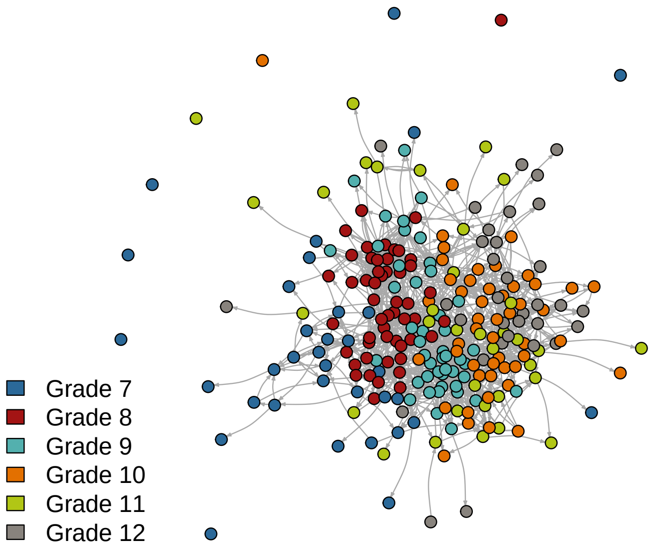


Figure 8.1: Faux Dixon High School Plot

dents in the same grade.

The final network has 248 nodes with 1,197 directed edges. Each node has three characteristics: grade, sex, and race. The grades include 7th-12th, and race is first delineated by Hispanic and non-Hispanic which was further split into Asian, Black, Native American, Other, and White. Figure 8.1 shows the network plotted with nodes colored for each grade showing the homophily.

Executing any of the three BALERGM algorithms requires choosing network statistics with both nodal and edge attributes and structural features such as triangles and triads such as those laid out in Chapter 1.

A natural network statistic for this data is the instances of homophily between students in the same grade, since as seen in the plot above 8.1, nodes with the same attribute (in this case grade) appear visually to have more connections. As seen below, with the diagonal entries of the mixing matrix from Grade i to Grade i for $i \in \{7, 8, 9, 10, 11, 12\}$, most of the connections are between students in the same grade. This feature can be included in network models with the R code nodematch.

Table 8.1: Number of student friendships in each grade pair

	Grade 7	**Grade 8**	**Grade 9**	**Grade 10**	**Grade 11**	**Grade 12**	**Sum**
Grade 7	42	5	8	3	3	1	62
Grade 8	9	263	48	10	7	4	341
Grade 9	13	53	184	35	32	15	332
Grade 10	3	14	46	183	14	13	273
Grade 11	0	2	13	12	42	16	85
Grade 12	0	4	11	10	8	71	104
Sum	67	341	310	253	106	120	1197

While it is possible just by looking at the graph to see that students are more likely to be friends with those of the same grade, this observation lacks specificity to the degree that this attraction motivates the structure of the network. While examining the matrix of student ties based on grades allows for a more specific justification of the instances of homophily, the network models offer a way to quantify the significance of these instances of homophily.

8.1.2 Results

For this comparison of the four models, we use the following network statistics: nodematch(“grade”), nodematch(“race”), and nodematch(“sex”). These each measure the homophily of three different nodal factors. This enables a comparison of the significance of having the same grade, the same race, or the same sex respectively.

To demonstrate the effectiveness of the new models, we generate 30 independent exponential random graphs based on the Faux Dixon High data set with known, fixed parameters $\boldsymbol{\theta}$. For this result, the chosen network statistics are the count of the edges in the network (θ_1) and the count of the occurrences of homophily where students of the same grade have a friendship connection (θ_2), the count of the occurrences of homophily for students with the same race (θ_3), and the counts of the occurrences of homophily with students having the same sex (θ_4). Using an initial run of BERGM algorithm to estimate, the $\boldsymbol{\theta} = (-5, 3, 0, 0)$ are fixed. Without the loss of generality, these 100 networks can be treated as new exponential random graphs with node attributes. This creates 30 opportunities to estimate $\boldsymbol{\theta}$ using both algorithms and compare performance to a true value.

Since the long-term behavior for all models is expected to be similar, we restrict the iteration numbers to better allow for the comparison of these models. For each of the 30 simulations and estimations, the initial parameters are set as below: Other settings are set at values recommended by the literature [10], the variance for the parallel adaptive sampling is 0.0001, and the scaling for the difference in the same process is 0.5.

Table 8.2: Initial settings for the simulation runs

main iterations	2,000
auxiliary iterations	200
burn-in iterations	200
chains	8

(a) Iteration numbers

r, δ	5, 1
τ^2	1
λ	5
σ^2	100

(b) Initial conditions

In each run of BERGM and BALERGM, the main chain for every model consists of 2,000 iterations after 200 burn-in iterations. In 30 simulations, each model generates a sequence of values estimating each $\boldsymbol{\theta}$ in each simulation. To confirm the stability of the model, the following representation of the MCMC results shows the strength and stability of the BALERGM algorithm after relatively few iterations.

The unimodal distribution of estimates is on the left of Figure 8.2, 8.3, 8.4, 8.5,

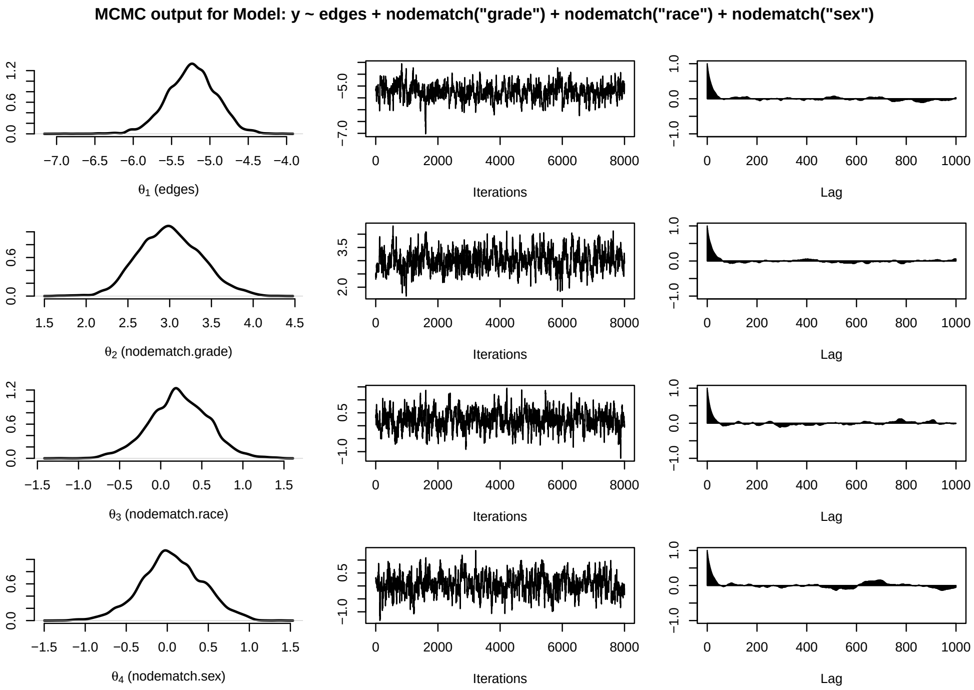


Figure 8.2: MCMC output of BERGM algorithm

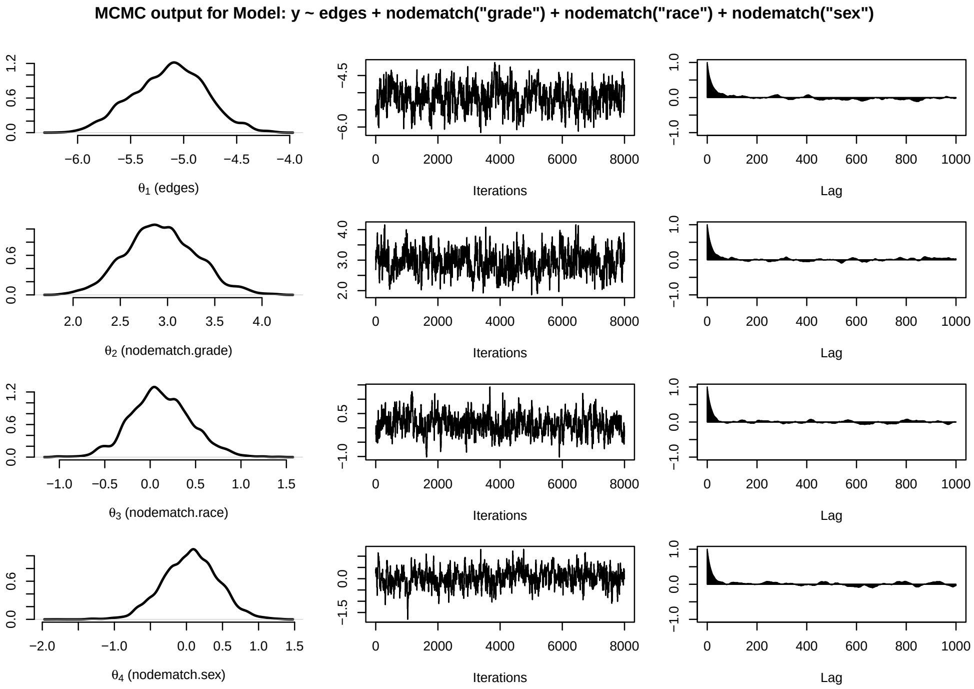


Figure 8.3: MCMC output of BALERGM Method 1 algorithm

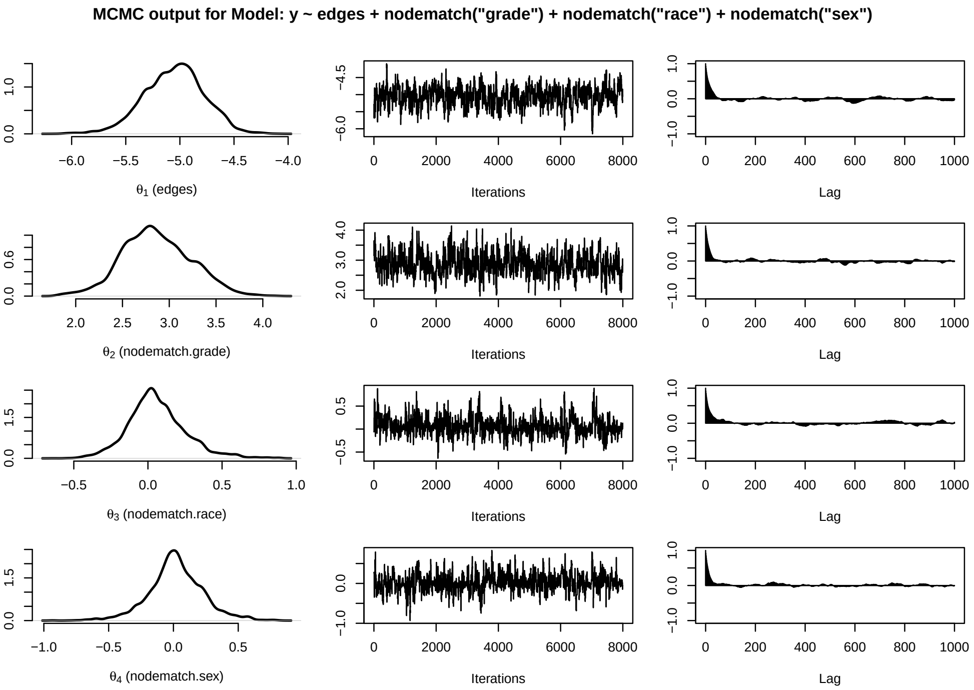


Figure 8.4: MCMC output of BALERGM Method 2 algorithm

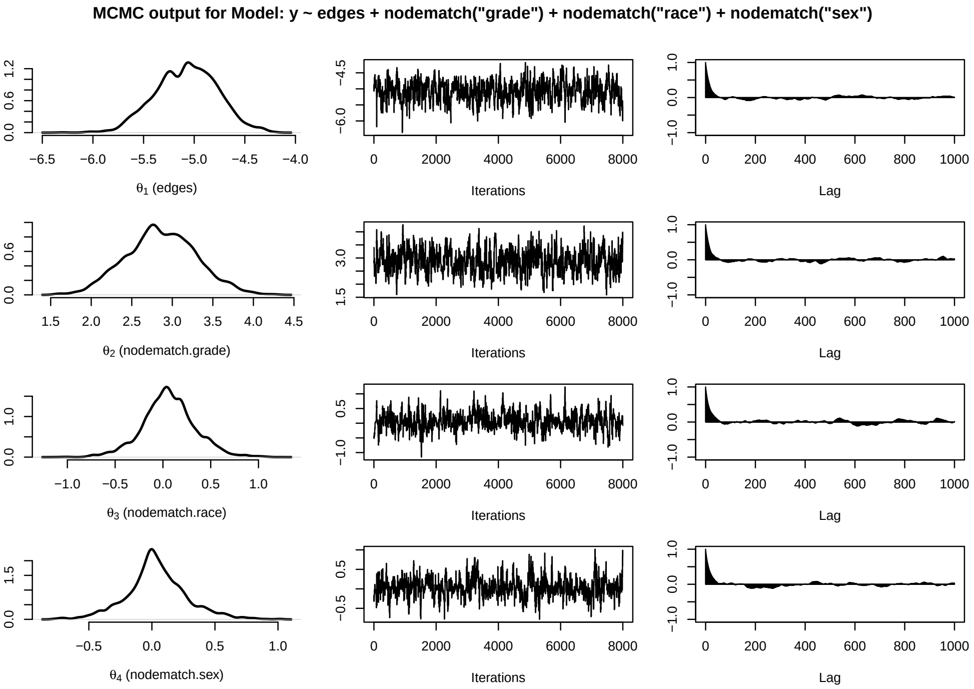


Figure 8.5: MCMC output of BALERGM Method 3 algorithm

and the center column shows the trace of the estimates indicating a stable estimating process. The final column shows the autocorrelation plot with the lag decreasing quickly; by 50 iterations, the process has stabilized to minimal lag.

These MCMC plots show that all three methods of the Bayesian adaptive lasso exponential random graph model can produce a stable approximation of $\boldsymbol{\theta}$

In Tables 8.3 and 8.4, the true known value of each $\boldsymbol{\theta}$ is estimated by either the mean or median of the generated samples. The quantiles for estimates of $\boldsymbol{\theta}$ show the spread of each estimate.

Table 8.3: Results of simulating 30 graphs and comparing results for BERGM and BALERGM using means as the estimates of $\boldsymbol{\theta}$

Mean of the MCMC output as the estimate for $\boldsymbol{\theta}$

		True Value [a]	Estimate [b]	Quantiles [c]				
				2.5%	*25%*	*50%*	*75%*	*97.5%*
BERGM	θ_1	-5.0	-5.1414	-5.283	-5.171	-5.349	-5.095	-5.043
	θ_2	3.0	2.9990	2.889	2.950	3.005	3.037	3.118
	θ_3	0.0	0.1371	0.035	0.092	0.132	0.175	0.265
	θ_4	0.0	0.0354	-0.069	0.015	0.044	0.067	0.185
BALERGM *Method 1*	θ_1	-5.0	-5.0713	-5.217	-5.110	-5.065	-5.017	-4.973
	θ_2	3.0	2.9214	2.799	2.878	2.904	2.972	3.048
	θ_3	0.0	0.1089	0.032	0.070	0.102	0.135	0.208
	θ_4	0.0	0.0096	-0.117	0.038	0.019	0.055	0.126
BALERGM *Method 2*	θ_1	-5.0	-5.0420	-5.147	-5.070	-5.035	-5.007	-4.962
	θ_2	3.0	2.9161	2.813	2.868	2.926	2.961	3.018
	θ_3	0.0	0.0432	0.001	0.026	0.042	0.052	0.118
	θ_4	0.0	0.0076	-0.030	-0.011	0.008	0.023	0.045
BALERGM *Method 3*	θ_1	-5.0	-5.0144	-5.127	-5.052	-4.995	-4.972	-4.945
	θ_2	3.0	3.8708	2.736	2.827	2.867	2.972	3.011
	θ_3	0.0	0.0272	0.005	0.021	0.041	0.053	0.114
	θ_4	0.0	0.0039	-0.040	0.009	0.006	0.021	0.048

---

[a] Chosen true value for parameter for each simulated graph
[b] Mean of MCMC outputs
[c] Quantiles from MCMC output

Table 8.4: Results of simulating 30 graphs and comparing results for BERGM and BALERGM using medians as the estimates of $\boldsymbol{\theta}$

Median of the MCMC output as the estimate for $\boldsymbol{\theta}$

		True Value	Estimate [a]	Quantiles				
				2.5%	*25%*	*50%*	*75%*	*97.5%*
BERGM	θ_1	-5.0	-5.1271	-5.283	-5.171	-5.349	-5.095	-5.043
	θ_2	3.0	2.9848	2.889	2.950	3.005	3.037	3.118
	θ_3	0.0	0.1388	0.035	0.092	0.132	0.175	0.265
	θ_4	0.0	0.0354	-0.069	0.015	0.044	0.067	0.185
BALERGM *Method 1*	θ_1	-5.0	-5.0589	-5.217	-5.110	-5.065	-5.017	-4.973
	θ_2	3.0	2.9082	2.799	2.878	2.904	2.972	3.048
	θ_3	0.0	0.1087	0.032	0.070	0.102	0.135	0.208
	θ_4	0.0	0.0095	-0.117	0.038	0.019	0.055	0.126
BALERGM *Method 2*	θ_1	-5.0	-5.0297	-5.147	-5.070	-5.035	-5.007	-4.962
	θ_2	3.0	2.9007	2.813	2.868	2.926	2.961	3.018
	θ_3	0.0	0.0351	0.001	0.026	0.042	0.052	0.118
	θ_4	0.0	0.0069	-0.030	-0.011	0.008	0.023	0.045
BALERGM *Method 3*	θ_1	-5.0	-5.0055	-5.127	-5.052	-4.995	-4.972	-4.945
	θ_2	3.0	3.8708	2.736	2.827	2.867	2.972	3.011
	θ_3	0.0	0.0272	0.005	0.021	0.041	0.053	0.114
	θ_4	0.0	0.0039	-0.040	0.009	0.006	0.021	0.048

---

[a] Median of MCMC outputs

Using the stable estimating process demonstrated in the last few graphs and tables, we can more directly compare the abilities of the four models. The table (8.5) shows that using either the mean or median of the generated estimates in MCMC for $\boldsymbol{\theta}$ First, BALERGM has a better overall acceptance rate and effective sample size on average than BERGM. The acceptance rate or the percentage of generated samples that are accepted in the MCMC process is increased. With the larger effective sample size, this implies all three methods of BALERGM adjust to the true parameter for each single variable faster than BERGM. Secondly, BALERGM offers an improvement over BERGM with a lower mean squared error (MSE). This can be seen in the quantiles for each estimate of $\boldsymbol{\theta}$ since the true values are $\boldsymbol{\theta} = (-5, 3, 0, 0)$, the

BALERGM estimates are much closer to these true values. The mean squared error is dramatically lower with the BALERGM process no matter whether the mean or median in MCMC is used as the estimate for $\boldsymbol{\theta}$. The three methods of BALERGM decrease the MSE when compared to the old method BERGM by %32, %68, and %53 respectively when using the mean or %32, %68, and %52 when using the median.

Table 8.5: Results of both BERGM and BALERGM using formula y $\sim$ edges + nodematch("Grade")+nodematch("race") + nodematch("sex")

Results					
		Median AR	Mean ESS	Mean Squared Error	Median Squared Error
BERGM	θ_1	0.33	238.7	0.01449166	0.01360988
	θ_2		229.3		
	θ_3		245.4		
	θ_4		243.6		
BALERGM *Method 1*	θ_1	0.34	251.2	0.009889426	0.00999344
	θ_2		245.5		
	θ_3		253.8		
	θ_4		249.3		
BALERGM *Method 2*	θ_1	0.35	272.5	0.00465299	0.004939428
	θ_2		215.8		
	θ_3		255.7		
	θ_4		268.0		
BALERGM *Method 3*	θ_1	0.33	223.2	0.00676853	0.006586248
	θ_2		215.8		
	θ_3		255.7		
	θ_4		268.0		

8.1.3 Goodness of Fit

As modeled in [10] one method of evaluating the efficacy of BALERGM is through a Bayesian goodness of fit diagnostic. To run Bayesian goodness of fit diagnostics, 100 graphs are simulated from 100 independent realizations taken from the estimated posterior distributions.

These 100 graphs are compared to the original network data in three non-explicit factors: degree distributions, the minimum geodesic distance, and the number of edge-wise shared partners. Since the Faux Dixon High School network graph is a directed graph, the degree distributions for both in and out degrees are included. Since the graph includes isolated nodes and clusters such that there is no path between some nodes, the minimum geodesic distance or the minimum number of edges needed to connect any two nodes is infinite leading to the spike in the plot for minimum geodesic distance in Figures 8.6, 8.7, and 8.8. Finally, the edge-wise shared partners is concentrated in the lower values since the number of nodes in common for any number of edges is small.

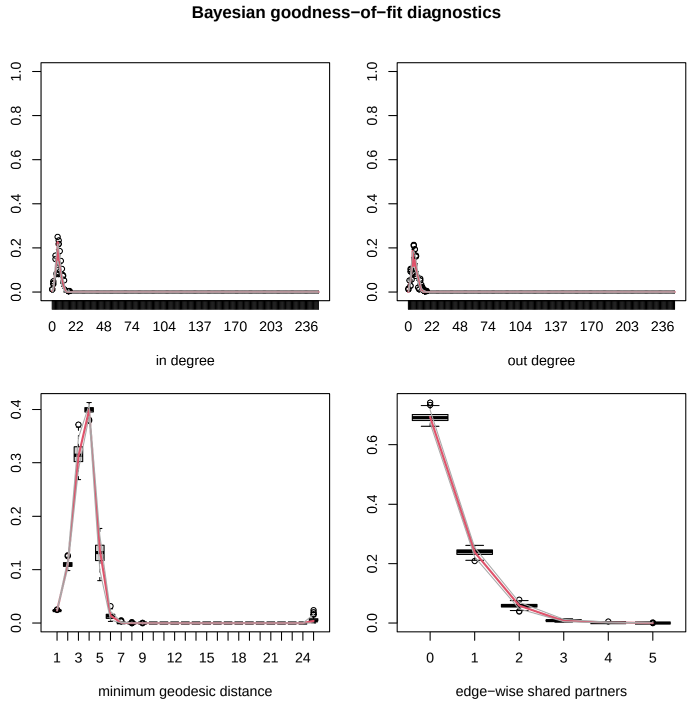


Figure 8.6: Goodness of Fit Diagnostics for BALERGM Method 1

Figures 8.6, 8.7, and 8.8 show the summary results of the 100 generated graphs in black and gray compared to the original network in red showing a strong match in all these high-level characteristics which are not modeled explicitly. This demon-

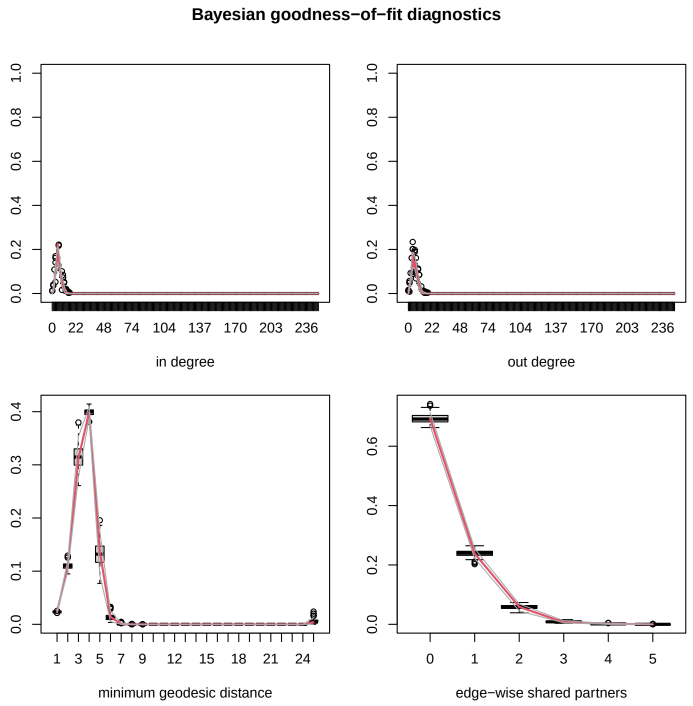


Figure 8.7: Goodness of Fit Diagnostics for BALERGM Method 2

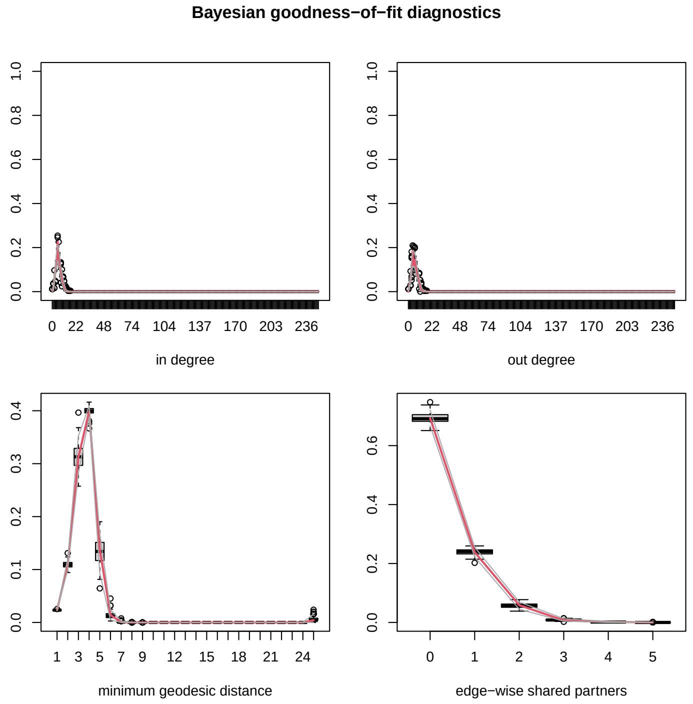


Figure 8.8: Goodness of Fit Diagnostics for BALERGM Method 3

strates that the posterior mean found through BALERGM correctly produces networks with matching structures.

8.2 Variable Selection

The next set of comparisons here highlights all three BALERGM methods' abilities to penalize network statistics that do not contribute to the structure of the network. Considering the frequency of networks with many covariates, parameter selection is a critical component of network models.

8.2.1 Data Description

The network object faux.magnolia.high represents a friendship network among junior high and high school students based on data gathered by a National Longitudinal Study of Adolescent Health [80] with faux.magnolia.high being based on two particular schools in the American South.

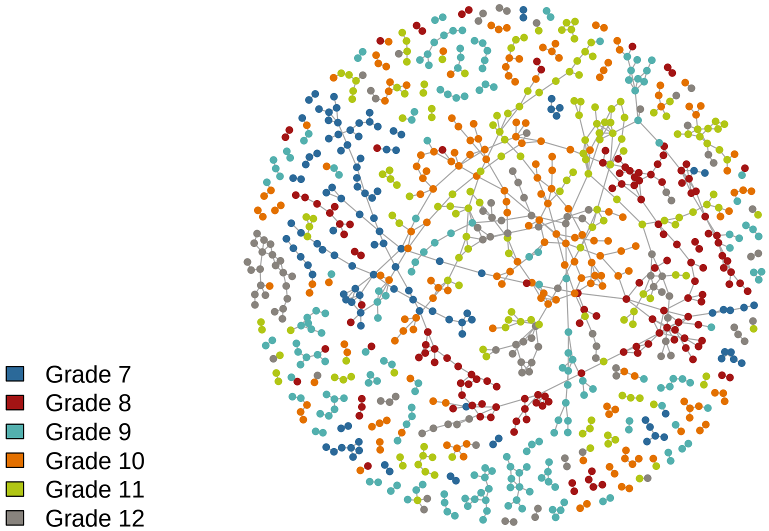


Figure 8.9: Faux Magnolia High School Plot

Generated in R, this plot shows the clustering of student friendships with students that have the same grade represented by ties between nodes of the same color.

The final network has 1,461 nodes with 974 undirected edges connecting the nodes. Each node has three characteristics: grade, sex, and race. The grades include 7th-12th, and race is first delineated by Hispanic and non-Hispanic which was further split into Asian, Black, Native American, Other, and White.

For this test of the model, we generated three fake variables. Since these variables are constructed in a random/uniform way, they should not be influential to the structure of the network and thus the corresponding θ_j should be zero.

The first variable is "GPA." This variable simulates a supposed grade point average for each student. Using a beta distribution with a shape parameter of 8 and a scale parameter of 2, this variable gives every student a GPA between 0 and 4.

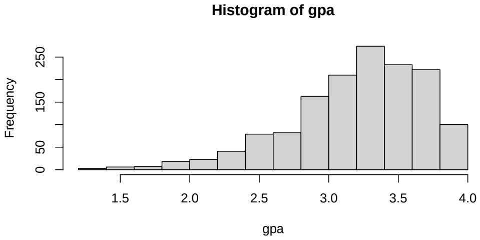


Figure 8.10: Histogram of the simulated GPA values for Faux Magnolia High

The second variable is generated as the estimated salary in thousands of dollars of each student's household. This is done as a uniform random variable between 20 and 75. The final variable is the estimated number of sports that each student plays. This variable is simulated as an integer between 0 and 5 with all values equally likely. While these variables are intended to have some connection to the real-life data they are meant to emulate, the key feature is that they are generated independently of the network structure and should not collate with the structural features of the network.

8.2.2 Results

Thus using the generated network statistics, we can run the following formula: y $\sim$ edges + nodematch("Grade") + nodecov("GPA") + nodecov("Wealth") + nodematch("Sport"). To understand the long-term behavior of this process, we choose a large number of iterations and chains. This allows for the comparison of these methods in performing variable selection. The initial parameters are set as below:

Table 8.6: Initial settings for the variable selection runs

main iterations	2,000
auxiliary iterations	1,000
burn-in iterations	3,000
chains	500

(a) Iteration numbers

r, δ	7.5, 1
τ^2	1
λ	10
σ^2	100

(b) Initial conditions

To improve the mixing, the variance for the parallel adaptive sampling is 0.0001, and the scaling for the difference in the same process is 0.5.

These MCMC outputs show a stable estimating process with convergent sampling distributions, a steady trace of estimates, and a quickly decreasing lag. Interestingly, BALERGM method 3 with the fully empirical estimation produces the sharpest distributions reflecting the double exponential distribution. All three methods are able to identify the three extraneous variables of nodecov("GPA"), nodecov("Wealth"), nodematch("Sport") all generated apart from the structure of the network. This means that all three estimates for the corresponding θ_j should be zero indicating that these variables are not significant to the graph. Table 8.7 shows that the estimates for θ_3, θ_4 and θ_5 are all very close to zero.

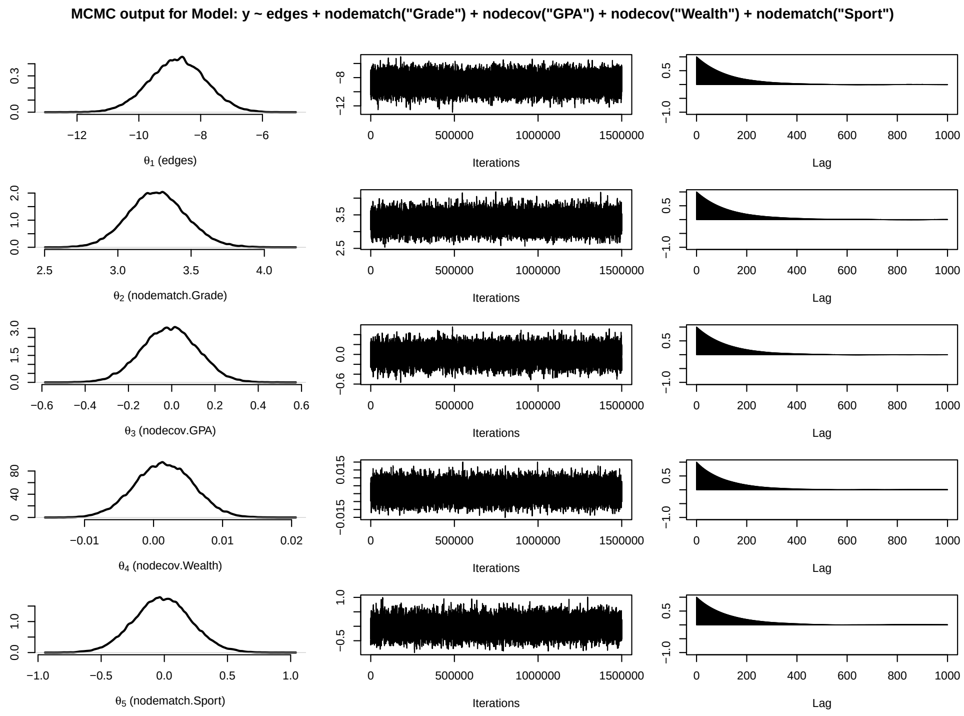


Figure 8.11: MCMC output of BALERGM Method 1 algorithm

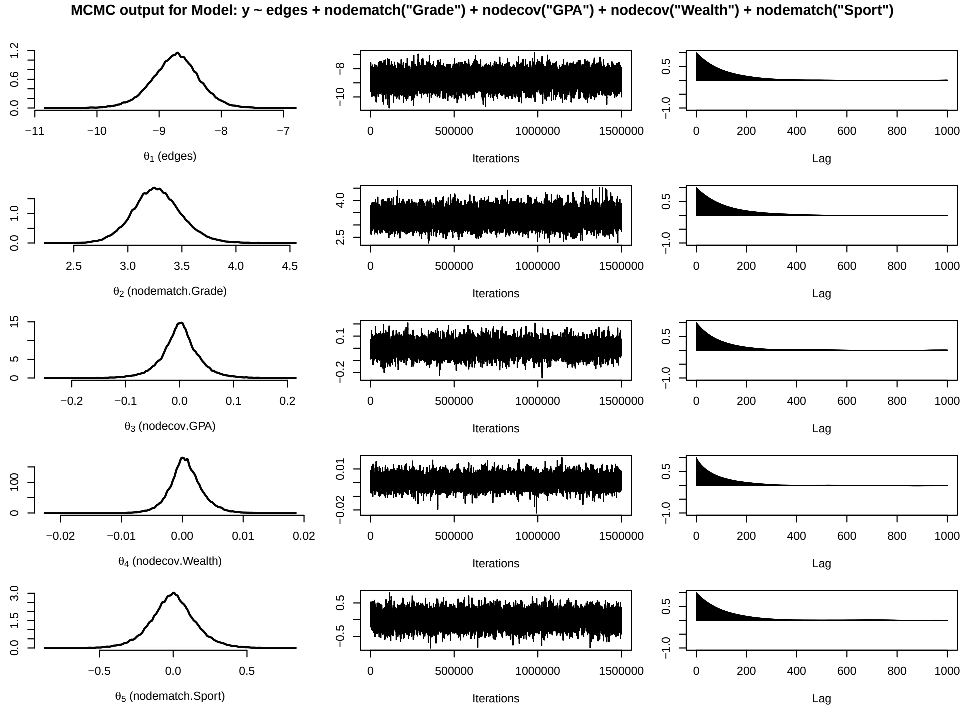


Figure 8.12: MCMC output of BALERGM Method 2 algorithm

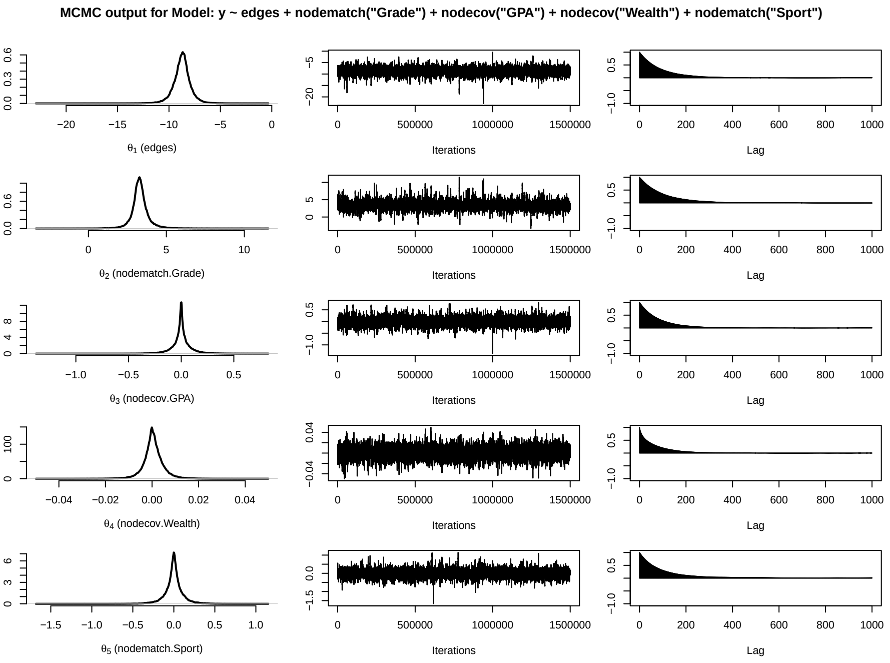


Figure 8.13: MCMC output of BALERGM Method 3 algorithm

Table 8.7: BALERGM estimates of $\boldsymbol{\theta}$

Mean of the MCMC output as the estimate for $\boldsymbol{\theta}$							
		Estimate[a]	Quantiles				
			2.5%	*25%*	*50%*	*75%*	*97.5%*
BALERGM *Method 1*	θ_1	-8.7467	-10.543	-9.356	-8.740	-8.126	-6.980
	θ_2	3.2707	2.893	3.137	3.267	3.400	3.668
	θ_3	-0.0094	-0.264	-0.097	-0.009	0.078	0.241
	θ_4	0.0015	-0.006	-0.001	0.001	0.004	0.009
	θ_5	-0.0136	-0.453	-0.165	-0.015	0.136	0.435
BALERGM *Method 2*	θ_1	-8.0297	-9.533	-8.979	-8.728	-8.488	-7.984
	θ_2	3.2720	2.836	3.122	3.264	3.414	3.7434
	θ_3	0.0008	-0.075	-0.019	0.001	0.020	0.076
	θ_4	0.0005	-0.004	-0.001	0.000	0.002	0.006
	θ_5	0.0004	-0.315	-0.095	-0.001	0.093	0.321
BALERGM *Method 3*	θ_1	-8.7408	-10.550	-9.183	-8.713	-8.283	-7.013
	θ_2	3.3131	2.207	3.037	3.280	3.546	4.625
	θ_3	3.334e-05	-0.196	-0.032	0.000	0.032	0.197
	θ_4	4.903e-04	-0.008	-0.001	0.000	0.002	0.010
	θ_5	-7.526e-04	-0.252	-0.047	0.000	0.046	0.246

---

[a] Mean of MCMC outputs

Table 8.8 demonstrates that BALERGM Method 3 with a fully empirical update of the penalty term $\boldsymbol{\lambda}$ is the most efficient method. Analyzing the long-term behavior of the MCMC process indicates that the performance of BALERGM Method 3 provides the highest number of samples that can be considered to be drawn independently.

Table 8.8: Results three methods of BALERGM using formula y ∼ edges + nodematch(“Grade”) + nodecov(“GPA”) + nodecov(“Wealth”) + nodematch(“Sport”)

Results			
		Effective Sample Size	Run time
BALERGM *Method 1*	θ_1	6,108	2.55 hours
	θ_2	6,075	
	θ_3	6,150	
	θ_4	6,893	
	θ_5	6,254	
BALERGM *Method 2*	θ_1	7,221	2.60 hours
	θ_2	6,834	
	θ_3	8,505	
	θ_4	9,767	
	θ_5	7,392	
BALERGM *Method 3*	θ_1	8,950	2.54 hours
	θ_2	8,054	
	θ_3	9,837	
	θ_4	13,351	
	θ_5	9,099	

Once we have the sampling distribution for each $\boldsymbol{\theta}$, we can find the probability of a sample less than zero. If this probability is about 0.5, then we can claim a distribution centered at zero for some significance level α. Thus we have:

$$|P(\boldsymbol{\theta} < 0) - 0.5| = \alpha.$$

This provides a more rigorous method of ranking variables and quantifying which variables contributed the least/most to the network.

Using this calculation, all three methods of BALERGM are able to select out the irrelevant variables θ_3, θ_4 and θ_5 when α is 0.1 or greater. Though all methods are able to perform the variable selections, in this particular larger network, BALERGM Method 3 is able to perform the more efficiently.

8.2.3 Goodness of Fit

To run Bayesian goodness of fit diagnostics, 100 graphs are simulated from 100 independent realizations taken from the estimated posterior distributions and compared to the original network data in three non-explicit factors: degree distributions, the minimum geodesic distance, and the number of edge-wise shared partners.

The original Faux Magnolia High School network graph is relatively sparse; out of 1461 nodes, 524 or 35.87% are isolated and have a degree of zero. In our goodness of fit experiments, the simulated graphs follow the same proportion of 35% of nodes having degree zero on average.

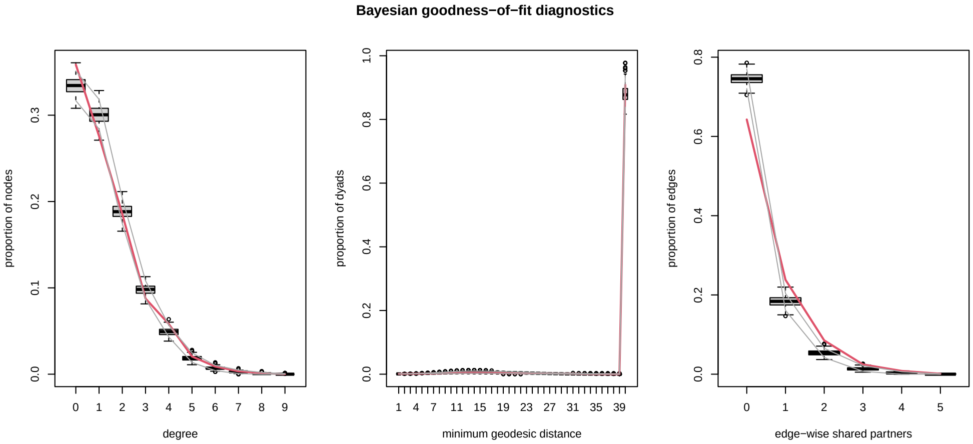


Figure 8.14: Goodness of Fit Diagnostics for BALERGM Method 1

This fact of the network also affects the minimum geodesic distance since that distance is considered to be infinite for nodes that are isolated. That is why in the middle of Figures 8.14, 8.15, and 8.16, there is a sharp increase on the right end of

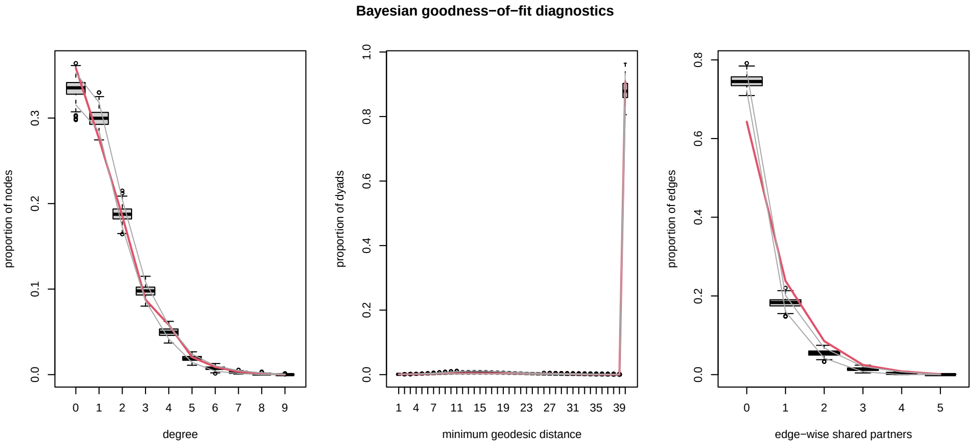


Figure 8.15: Goodness of Fit Diagnostics for BALERGM Method 2

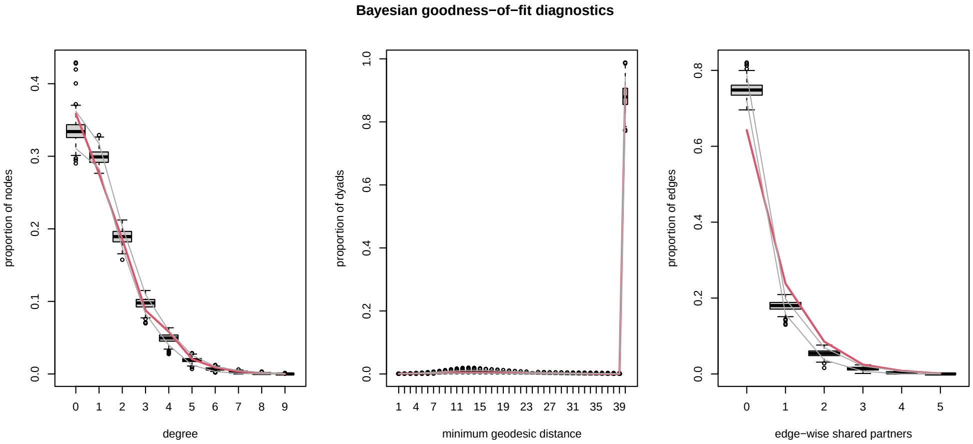


Figure 8.16: Goodness of Fit Diagnostics for BALERGM Method 3

the graph. However, our simulated graphs based on BALERGM in red track this sparsity feature in the original data as well. From Figures 8.14, 8.15, and 8.16, our generated graphs in red match the original network in black in all these high-level characteristics which are not modeled explicitly.

8.3 Application

In this final application section, we demonstrate the critical improvement offered by the BALERGM algorithm. The BERGM algorithm takes a prohibitively long time to produce estimates $\boldsymbol{\theta}$ for the following data set, but BALERGM penalizes non-important statistics allowing for parameter selection in an efficient way. In cases such as this where the number of covariates is large, BALERGM effectively estimates $\boldsymbol{\theta}$ providing a distinct advantage over older models.

8.3.1 Data Description

With reports by [92] of worsening metrics of American health, communities are working on addressing and understanding the factors that might improve health outcomes. To this end, the University of South Carolina Prevention Research Center and Sumter County Active Lifestyles (SCAL) based in Sumter County, South Carolina conducted a respondent-driven sampling study in 2014 to better understand the dynamics of social networks and health outcomes.

In this study, community ambassadors chosen for their history of community involvement were given a set compensation for their participation. Each ambassador was instructed to share the survey with those in their social network. Each of these respondents was also compensated for both completion of the survey and sharing the survey with others that completed the survey. Using referral codes, a network can be created with nodes representing survey respondents, edges formed by survey sharing, and nodal characteristics from the results of the survey. The final network has 80 nodes with the data for 30 questions for each respondent.

The survey was intended to be a brief but broad look at self-reported health benchmarks. Questions cover demographic characteristics revealing that the respondents are primarily white (87%), female (78%), likely to be older than 50 (44%), and more educated with 46% being college graduates. Other questions focused on self-reported health outcomes and activities including exercise habits, eating habits, and social support dynamics. The question forms included qualitative questions about physical activities and opportunities for physical activities in the community. For the purposes of this network, network attributes were assigned using the answers to only multiple-choice questions.

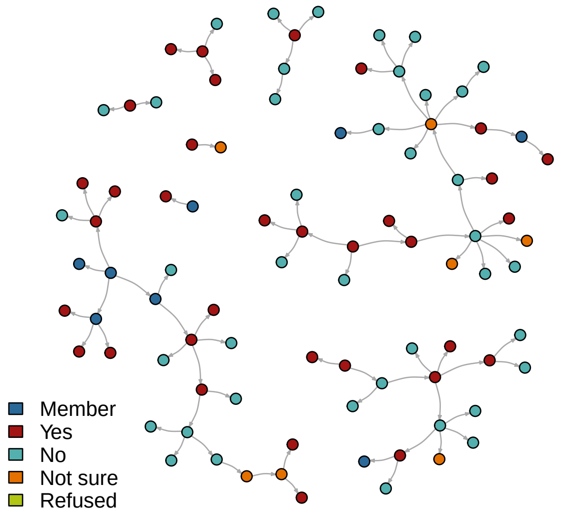


Figure 8.17: Generated in R, this plot shows results of asking "Have you heard of a group called Sumter County Active Lifestyles (SCAL)?"

The resulting network contains many nodal attributes. This motivates a model like BALERGM which enables understanding which of these network statistics contribute less to the network structure.

8.3.2 Results

Using the SCAL data set from the previous section, we use the network statistics in Table 8.9 to analyze this model.

The ergm terms "nodematch," "nodefactor," and "nodecov" all provide measures of homophily. Nodematch counts the instances of nodes with the same attribute for a given attribute. Nodecov performs a similar function but for continuous variables. Nodefactor creates network statistics for each discrete level of a nodal attribute and counts the occurrences of connected nodes with the same attribute level. For more details, see [70].

Table 8.9 shows the BALERGM output on the SCAL social network. Here the sparsity of the network can be seen in the large negative values for the network statistics for edges and the out-degree of the nodes. While the standard deviations vary with each estimate, the MCMC outputs show stable estimating with symmetric distributions as the quantile values indicate.

The adaptive lasso penalty in BALERGM is useful for shrinking $\boldsymbol{\theta}$ values for network statistics that are less significant to the network structures. Depending on the model and network conditions, the parameter estimate might not reach exactly zero. For example, the estimate for both $\theta_{26} = -.001$ and $\theta_{20} = -.076$ are small, but this mean of the generated samples as the single factor utilized doesn't allow for a nuanced ranking of how significant each parameter is. Using the distribution of $\boldsymbol{\theta}$ found in the Gibbs sampling process, we can find the probability that half of the distribution is less than zero at some significance level α:

$$|P(\boldsymbol{\theta} < 0) - 0.5| = \alpha$$

. This creates the ability to rank variables. The following Table 8.10 shows the variables less significant to the construct of the network at various significance levels. This calculation takes into account the spread and centering of the distribution to quantify how close an estimate truly is to zero. This table shows that the network statistic of the nodecov of the age of the participant (θ_{26}) is less significant than for the network statistic of nodematch of having heard of the SCAL program (θ_{20}). While neither are primary factors in the network structure, the values found by BALERGM give researchers insights into the social dynamics of Sumter County allowing for targeted programs to improve health outcomes.

Interestingly, θ_6 through θ_{13} which correspond to questions about participants' diets are all positive values indicating that participants are more likely to share a connection if they have similar eating habits. Additionally, participating in the walking program (θ_{19}) is a much stronger predictor of a tie than having heard of the SCAL program θ_{20}. This is a reflection on the structure of the network in this respondent-driven sampling survey, demonstrating the type of information that estimates for $\boldsymbol{\theta}$ can indicate.

Table 8.9: Results from BALERGM with Variable Selection on SCAL data

Result for Parameter Selection							
Network Statistic	**Mean**	**SD**	**Quantiles**				
			2.5%	25%	50%	75%	97.5%
θ_1 (edges)	-5.794	0.995	-7.770	-6.460	-5.782	-5.121	-3.879
θ_2 (out degree 0)	1.212	0.64	-0.051	0.795	1.216	1.631	2.477
θ_3 (out degree 1)	-1.025	0.526	-2.066	-1.372	-1.025	-0.679	0.016
θ_4 (out degree 2)	-0.629	0.407	-1.428	-0.901	-0.632	-0.363	0.189
θ_5 (out degree 3)	-0.207	0.319	-0.853	-0.410	-0.209	-0.002	0.431
θ_6 (1 serving fruit/day) [a]	0.654	0.308	0.046	0.453	0.655	0.859	1.265
θ_7 (2 servings fruit/day) [a]	0.536	0.304	-0.063	0.334	0.534	0.738	1.134
θ_8 (3-4 servings fruit/day) [a]	0.608	0.329	-0.054	0.391	0.612	0.829	1.245
θ_9 (5+ servings fruit/day) [a]	0.877	0.461	-0.066	0.577	0.887	1.186	1.772
θ_{10} (1 serving vegetables/day) [a]	0.451	0.310	-0.164	0.248	0.454	0.655	1.062
θ_{11} (2 servings vegetables/day) [a]	0.477	0.295	-0.106	0.283	0.478	0.675	1.056
θ_{12} (3-4 servings vegetables/day) [a]	0.587	0.292	0.018	0.392	0.587	0.782	1.165
θ_{13} (5+ servings vegetables/day) [a]	0.096	0.359	-0.644	-0.128	0.109	0.335	0.770
θ_{14} (vigorous phys. activities/week)[b]	-0.011	0.206	-0.426	-0.145	-0.008	0.125	0.389
θ_{15} (moderate phys. activities/week)[c]	0.008	0.042	-0.075	-0.019	0.009	0.037	0.090
θ_{16} (days walking 10min/week)[c]	-0.018	0.035	-0.088	-0.042	-0.019	0.006	0.052
θ_{17} (days using parks/month) [c]	-0.030	0.028	-0.090	-0.049	-0.030	-0.011	0.022
θ_{18} (heard of walking program) [b]	0.333	0.214	-0.087	0.190	0.334	0.478	0.752
θ_{19} (participate in walking program)[b]	0.186	0.244	-0.305	0.027	0.188	0.346	0.668
θ_{20} (heard of SCAL)[b]	0.076	0.191	-0.310	-0.049	0.081	0.206	0.448
θ_{21} (general health is very good)[a]	-0.203	0.206	-0.616	-0.339	-0.201	-0.065	0.194
θ_{22} (general health is good) [a]	-0.247	0.204	-0.650	-0.381	-0.248	-0.113	0.154
θ_{23} (general health is fair) [a]	-0.072	0.207	-0.470	-0.211	-0.077	0.063	0.349
θ_{24} (general health is poor) [a]	-0.479	0.477	-1.505	-0.773	-0.456	-0.160	0.402
θ_{25} (gender) [a]	-0.064	0.15	-0.362	-0.163	-0.064	0.035	0.226
θ_{26} (age) [c]	-0.001	0.005	-0.012	-0.005	-0.001	0.002	0.009
θ_{27} (highest year of school completed)[b]	-0.165	0.205	-0.571	-0.302	-0.166	-0.029	0.240

---

[a] nodefactor

[b] nodematch

[c] nodecov

Table 8.10: Variable Selection with Different Tolerance Levels

Tolerance Level	Variable Index Number
0.05	26
0.10	16 20 25 26
0.15	4 16 20 25 26
0.20	4 14 15 16 20 22 25 26

CHAPTER 9
BAYESIAN ADAPTIVE RIDGE EXPONENTIAL RANDOM GRAPH MODEL

In addition to the lasso penalty, the remaining sections of this dissertation will discuss the ridge penalty for the Bayesian exponential random graph. This modification enables improved estimations for models involving multicollinearity. This chapter introduces the theory behind the ridge penalty in the context of classical regression before then presenting the theory for the Bayesian adaptive ridge exponential random graph model. We first develop the Cauchy prior for the $\boldsymbol{\lambda}$ penalty at the recommendation of the literature. While this prior results in a model comparable with the BERGM, we also develop a model for the Bayesian adaptive ridge exponential random graph model with a gamma distribution prior for the $\boldsymbol{\lambda}$ parameter. This distribution proves advantageous as the concentration of the probability of the gamma distribution more closely matches the model requirements.

9.1 Classical Ridge Penalty

The ridge penalty term is first introduced in the context of biased estimation for nonorthogonal problems in classical regression by Hoerl and Kennard [49]. Within a few years, in 1975, Marquardt and Snee note the utility of ridge regression when data is highly correlated and indicate ridge regression's relative ease of calculation [65]. Later the same year, McDonald and Galarneau develop Monte Carlo methods for estimating β with ridge estimation [66]. Gibbons in [31] used these Monte Carlo methods to compare 10 different penalty calculation methods finding [17], [34], and

[48] show strong performance confirming previous work while [47] performed poorly. Hsiang [51] first suggests the combination of Bayesian analysis and ridge regression for the classical regression setting. In [60], Lawless and Wang show that incorporating a Bayesian approach to finding the penalty parameters improves estimates better than Hoerel and Kennard [49] who estimate parameters from the data. Erp et al. compare the results of several popular penalty priors including ridge, lasso, group lasso, hyper lasso, elastic net, and others on both full and empirical Bayes methods for finding ridge penalties. The conclusions show ridge penalty priors outperform all other methods in MSE tests under certain conditions [97]. Similarly [56] and [2] find improvements over ordinary least squares estimates. Later, [71] compare estimators from [56] and [55] to find the size of σ and the degree of correlation between variables influence the effectiveness of each method with the newly proposed methods and those of [56] surpassing others. The ridge penalty and in particular Bayesian ridge penalty has a long history of utility particularly in the face of multicollinearity issues.

Considering the typical classical regression model with $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ where $\boldsymbol{y} = (y_1, y_2, \cdots, y_n)^\top$ is the vector of observations, $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_p)$ is an $n \times p$ predictor matrix, $\boldsymbol{\beta} = (\beta_1, \beta_2, \cdots, \beta_p)$ is a corresponding vector of regression coefficients, $\boldsymbol{\epsilon} = (\epsilon_1, \cdots, \epsilon_n)$ is a vector of independent normal distributed errors, then the ridge estimates are defined as

$$\hat{\beta}(lasso) = \arg\min_{\boldsymbol{\beta}} \|\boldsymbol{y} - \sum_{j=1}^{p} \boldsymbol{x_j}\beta_j\|^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{9.1}$$

This minimum is achieved at $\hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\mathbb{I})^{-1}\boldsymbol{X}^T\boldsymbol{y}$ Because of the structure of this penalty, ridge penalties in contract to other penalties such as lasso do not penalize parameters to zero.

While the classical regression theory for ridge penalty is well developed, the corresponding theory for networks is a more open field. In 2014, [41] with their package [40] explored ridge penalties for graphical models in biology particularly in the context of gene networks. More recently, [101] use Bayesian adaptive ridge regression to understand blood protein graphs. While the context of graphical models is interesting, they fail to fully capture the information contained in networks, namely the assumption of interdependence foundational to ERGMs.

In this paper, we incorporate the strengths of Bayesian analysis and the adaptive ridge penalty to produce an effective model for estimating and understanding parameters in networks.

9.2 Prior Specification

With the ability to use Markov Chain Monte Carlo methods, we now develop the needed prior. The ridge estimates for $\boldsymbol{\theta}$ maximize the likelihood $l(\boldsymbol{\theta}|\boldsymbol{y})$ with a penalty term defined below.

$$\hat{\boldsymbol{\theta}} = \arg\max_{\theta} l(\boldsymbol{\theta}|\boldsymbol{y}) - P(\boldsymbol{\theta}) \tag{9.2}$$

$$P(\boldsymbol{\theta}) = \sum_{j=1}^{p} \lambda_j \theta_j^2 \tag{9.3}$$

Modeling after the work of Fu in [27] for the general case of the bridge regression we choose the following prior which is then equivalent to having the penalty l_2 penalty as in Equation 9.2

$$\pi_\lambda(\boldsymbol{\theta}) = C(\lambda) e^{-\lambda \|\boldsymbol{\theta}\|_q^2} \tag{9.4}$$

Comparing to the pdf of the normal distribution, we choose normal priors for each θ_j each centered on zero. Thus the conditional prior is as below

$$\pi(\theta_j|\sigma^2, \lambda) = \sqrt{\frac{\lambda}{2\pi\sigma^2}} \exp\left\{-\frac{\lambda\theta_j^2}{2\sigma^2}\right\}. \tag{9.5}$$

9.3 Bayesian Ridge Exponential Random Graph Model- Cauchy Prior

Using the recommendation in the literature from [97] and [77], the distribution for the penalty λ is chosen to follow a half-Cauchy distribution with parameters 0 and 1.

$$\theta_j|\lambda_j, \sigma^2 \sim \text{Normal}(0, \frac{\sigma^2}{\lambda_j}), \text{for } j = 1, ..., p \tag{9.6}$$

$$\boldsymbol{\lambda} \sim \text{half-Cauchy}(0, 1) \tag{9.7}$$

The Gibbs sampling for the current parameters is not straightforward, so a brief discussion of the half-Cauchy distribution allows for utilizing this process.

9.3.1 Half Cauchy

The half-Cauchy prior proves to provide computational difficulties. We have in [99] several steps that will allow for the Gibbs sampling to be developed to avoid these difficulties.

First, consider the Half-Cauchy distribution density below:

$$\pi(\lambda_j|\mu, \sigma) = \frac{2}{\pi\sigma}\frac{1}{1 + (\lambda_j - \mu)^2/\sigma^2} \qquad \text{for } y \geq \mu. \tag{9.8}$$

When $\lambda_j \sim$ Half-Cauchy $(0, A)$, λ_j is a Half t distribution with one degree of freedom such that $\lambda_j \sim$ Half -$t(A, 1)$. Here is the general form for the pdf of the Half t distribution

$$\frac{2\Gamma(\frac{\nu+1}{2})}{\sqrt{\pi\nu}\Gamma(\frac{\nu}{2})A[1 + (\frac{\lambda_j}{A})^2/\nu]^{\frac{\nu+1}{2}}} \text{ for } A, \nu > 0. \tag{9.9}$$

Substituting in the conditions $\nu = 1, A = \gamma_j$ we get the following distribution

$$\frac{2\Gamma(\frac{1+1}{2})}{\sqrt{\pi 1}\Gamma(\frac{1}{2})\gamma_j\{1 + (\frac{\lambda_j}{\gamma_j})^2/1\}^{\frac{1+1}{2}}} \text{ for } \gamma_j > 0. \tag{9.10}$$

Simplifying we get the following which is analogous to the Half-Cauchy distribution.

$$\frac{2}{\pi\gamma_j\{1 + (\lambda_j/\gamma_j)^2\}} \text{ for } \gamma_j > 0 \tag{9.11}$$

We also have a result that relates the Half t distribution and a scale mixture of Inverse Gamma distributions [99].

Let x and a be random variables such that

$$x|a \sim \text{ Inverse-Gamma}(\frac{\nu}{2}, \frac{\nu}{a}) \text{ and } a \sim \text{ Inverse-Gamma}(\frac{1}{2}, \frac{1}{A^2}).$$

Given these conditions, we then know $\sqrt{x} \sim \text{Half-}t(A, \nu)$.

Using the above statements, we use a conditional distribution of $\lambda_j|\gamma_j$ and the distribution of γ_j both following inverse gamma distributions.

$$\begin{aligned} \lambda_j|\gamma_j &\sim \text{ Inverse Gamma } \left(\frac{1}{2}, \frac{1}{\gamma_j}\right), \\ \gamma_j &\sim \text{ Inverse Gamma } \left(\frac{1}{2}, 1\right). \end{aligned} \tag{9.12}$$

9.3.2 Hierarchical Model One

With these pieces, we can now set up the complete hierarchical model with θ_j following a normal distribution from the chosen prior,

$$\pi(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{1}{z(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^T s(\boldsymbol{y})} \tag{9.13}$$

$$\theta_j|\tau_j^2, \sigma^2\lambda \sim \text{Normal}(0, \sigma^2\tau_j^2), \text{for } j = 1, ..., p \tag{9.14}$$

$$\lambda_j|\gamma_j \sim \text{Inverse Gamma}(1/2, 1/\gamma_j) \tag{9.15}$$

$$\gamma_j \sim \text{Inverse Gamma}(1/2, 1) \tag{9.16}$$

$$\pi(\sigma^2) \propto \pi(\frac{1}{\sigma^2}). \tag{9.17}$$

9.3.3 Gibbs Sampling

As before, we begin with the joint distribution.

$$
\begin{aligned}
&\pi(\boldsymbol{y}, \boldsymbol{\theta}, \sigma^2, \boldsymbol{\lambda}, \boldsymbol{\gamma}) \\
&= \pi(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \\
&= \pi(\boldsymbol{y}|\boldsymbol{\theta}) \prod_{j=1}^{p} \left(\pi(\theta_j|\lambda_j, \sigma^2)\pi(\lambda_j|\gamma_j)\pi(\gamma_j)\right) \pi(\sigma^2) \\
&= \frac{1}{z(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^T s(\boldsymbol{y})} \frac{1}{\sigma^2} \prod_{j=1}^{p} \frac{1}{\sqrt{2\pi\frac{\sigma^2}{\lambda_j}}} \exp\left\{-\frac{\theta_j^2}{2\frac{\sigma^2}{\lambda_j}}\right\} \frac{\left(\frac{1}{\gamma_j}\right)^{1/2}}{\Gamma(\frac{1}{2})} \left(\frac{1}{\lambda_j}\right)^{\frac{1}{2}+1} e^{-\frac{1}{\gamma_j\lambda_j}} \frac{1}{\Gamma(\frac{1}{2})} \left(\frac{1}{\gamma_j}\right)^{\frac{1}{2}+1} e^{-\frac{1}{\gamma_j}}
\end{aligned}
\tag{9.18}
$$

We now find the sampling distribution for each parameter.

Sample λ_j

$$
\begin{aligned}
&\frac{1}{\sqrt{2\pi\frac{\sigma^2}{\lambda_j}}} \exp\left\{\frac{\theta_j^2}{2\frac{\sigma^2}{\lambda_j}}\right\} \left(\frac{1}{\lambda_j}\right)^{\frac{1}{2}+1} e^{-\frac{1}{\gamma_j\lambda_j}} \\
&\propto \frac{\sqrt{\lambda_j}}{\sqrt{2\pi\sigma^2}} \exp\left\{-\lambda_j\frac{\theta_j^2}{2\sigma^2}\right\} \left(\frac{1}{\lambda_j}\right)^{\frac{3}{2}} e^{-\frac{1}{\gamma_j\lambda_j}} \\
&\propto \lambda_j^{-\frac{1}{2}} \exp\left\{-\lambda_j\frac{\theta_j^2}{2\sigma^2} - \lambda_j^{-1}\left(\frac{1}{\gamma_j}\right)\right\}
\end{aligned}
\tag{9.19}
$$

Recall the pdf of Generalized inverse Gaussian is proportional to

$$
f(x) = x^{(p-1)} e^{-(ax+\frac{b}{x})/2} \tag{9.20}
$$

where $a, b,$ and $x > 0$. Thus $\lambda_j \sim$ Inverse Gaussian $(p = 0, a = \frac{\theta_j^2}{2\sigma^2}, b = \frac{1}{\gamma_j})$

Sample γ_j

$$
\begin{aligned}
&\frac{\left(\frac{1}{\gamma_j}\right)^{1/2}}{\Gamma(\frac{1}{2})} e^{-\frac{1}{\gamma_j\lambda_j}} \left(\frac{1}{\gamma_j}\right)^{\frac{1}{2}+1} e^{-\frac{1}{\gamma_j}} \\
&\propto \gamma_j^{-\frac{1}{2}} e^{-\frac{1}{\gamma_j}\frac{1}{\lambda_j}} \gamma_j^{-\frac{3}{2}} e^{-\frac{1}{\gamma_j}} \\
&= \gamma_j^{-2} e^{-\frac{1}{\gamma_j}\left(1+\frac{1}{\lambda_j}\right)}
\end{aligned}
\tag{9.21}
$$

We have $\gamma_j \sim$ Inverse Gamma $\left(1, \left(1 + \frac{1}{\lambda_j}\right)\right)$

Sample σ^2

Pieces that involve σ^2

$$\begin{aligned}
&\frac{1}{\sigma^2} \prod_{j=1}^{p} \frac{1}{\sqrt{2\pi \frac{\sigma^2}{\lambda_j}}} \exp - \left\{ \frac{\theta_j^2}{2\frac{\sigma^2}{\lambda_j}} \right\} \\
&\propto \frac{1}{\sigma^2} \left(\frac{1}{\sqrt{\sigma^2}} \right)^p \exp - \left\{ \frac{1}{\sigma^2} \sum_{j=1}^{p} \frac{\theta_j^2 \lambda_j}{2} \right\} \qquad (9.22) \\
&= (\sigma^2)^{-\left(\frac{2+p}{2}\right)} \exp - \left\{ \frac{1}{\sigma^2} \left(\sum_{j=1}^{p} \frac{\theta_j^2 \lambda_j}{2} \right) \right\}
\end{aligned}$$

Comparing to the pdf of the inverse gamma distribution:

$$\frac{B^A}{\Gamma(A)} x^{-A-1} e^{-B/x} \text{ for } A, B > 0$$

we have $\sigma^2 \sim$ Inverse Gamma $\left(\frac{p}{2}, \left(\sum_{j=1}^{p} \frac{\theta_j^2 \lambda_j}{2}\right)\right)$

We can summarize the findings of the last three steps in Table 9.1.

Table 9.1: Sampling distributions from joint distribution for each variable

Variable	**Proportional Distribution**
λ_j	Inverse Gaussian $\left(0, \frac{\theta_j^2}{2\sigma^2}, \frac{1}{\gamma_j}\right)$
γ_j	Inverse Gamma $\left(1, \left(1 + \frac{1}{\lambda_j}\right)\right)$
σ^2	Inverse Gamma $\left(\frac{p}{2}, \left(\sum_{j=1}^{p} \frac{\theta_j^2 \lambda_j}{2}\right)\right)$

Thus, we can lay out the full algorithm using the parallel adaptive direction sampler method to update the estimates of $\boldsymbol{\theta}$, and the sampling distributions from Table 9.1.

9.4 Algorithm

Algorithm: Bayesian Ridge Exponential Random Graph Model- Cauchy Prior

---

```
Require: Set the initial value for $\boldsymbol{\lambda}, \sigma^2, \boldsymbol{\gamma}$, Use ERGM to find MPLE (Maximizer to the Psuedolikelihood Function) to find initial values for $\boldsymbol{\theta}$. Denote samples of $\boldsymbol{\theta}$ in the $h$th chain, as $\boldsymbol{\theta}_h$.
while $i = 1, ..., N$ do
    while $h = 1, ..., H$ do
        1. sample $\boldsymbol{\theta}_h$ with Parallel Adaptive Direction Sampler:
            a. generate $h_1$ and $h_2$ such that $h_1 \neq h_2 \neq h$
            b. update $\boldsymbol{D}_{\boldsymbol{\tau}}^{-1}$
            c. generate $\boldsymbol{\theta}'_h$ from $\gamma(\boldsymbol{\theta}_{h_1} - \boldsymbol{\theta}_{h_2}) + \epsilon(\dot{}\cdot\cdot\,|\boldsymbol{\theta}_h)$
            d. simulate $\boldsymbol{y}'$ from $\pi(\dot{}\cdot\cdot\,|\boldsymbol{\theta}'_h)$
            e. update $\boldsymbol{\theta}_h \to \boldsymbol{\theta}'_h$ with the log of the probability
                $$\min\left(0, [\boldsymbol{\theta}_h - \boldsymbol{\theta}'_h]^T[s(\boldsymbol{y}') - s(\boldsymbol{y})] + \log\left[\frac{\pi(\boldsymbol{\theta}'_h)}{\pi(\boldsymbol{\theta}_h)}\right]\right)$$
            where $\pi(\boldsymbol{\theta}) \sim \mathscr{N}(0_p,\, \sigma^2\boldsymbol{D}_{\boldsymbol{\tau}})$
        2. sample $\lambda_j$ for $j = 1, 2, 3, ..$ by generating a sample from
                $$\text{Inverse Gaussian }\left(0, \frac{\theta_j^2}{2\sigma^2}, \frac{1}{\gamma_j}\right)$$
        3. sample $\gamma_j$ for $j = 1, 2, 3, ..$ by generating a sample from
                $$\text{Inverse Gamma }\left(1, \left(1 + \frac{1}{\lambda_j}\right)\right)$$
        3. sample $\sigma^2$ by generating a sample from Inverse Gamma($\left(\frac{p}{2}, \left(\sum_{j=1}^{p} \frac{\theta_j^2\lambda_j}{2}\right)\right)$
    end while
end while
```

This process has introduced an additional parameter γ_j with its own distribution, and since this type of model is sensitive to initial conditions this increased the difficulty of finding appropriate tuning. This drawback motivates the next proposed model which improves the flexibility of $\boldsymbol{\lambda}$ without increasing the number of hyperparameters.

9.5 Bayesian Ridge Exponential Random Graph Model- Gamma Prior

While it has been suggested in the literature from [97] and [77], the distribution for the penalty λ is chosen to follow a half-Cauchy (0,1), we have found in the context of networks the concentration of probability between 0 and 1 in the half-Cauchy distribution is insufficiently flexible for the models. Also, to mix with the other distributions, the half-Cauchy distribution required using definitions to add an additional parameter to enable Gibbs sampling.

9.5.1 Hierarchical Model

Using the same reasoning as the previous chapter, we choose a normal prior for each θ_j

$$\pi(\theta_j|\sigma^2, \lambda) = \sqrt{\frac{\lambda}{2\pi\sigma^2}} \exp\left\{-\frac{\lambda\theta_j^2}{2\sigma^2}\right\} \tag{9.23}$$

While the recommendation from the classical regression context was the half-Cauchy distribution, for networks with higher dimensions we need more possibilities for the $\boldsymbol{\lambda}$ penalty such that the probability is not always concentrated near zero. We propose gamma$(r, \delta_j), j = 1, 2, \cdots, p$. This allows for a more flexible control of the range of the $\boldsymbol{\lambda}$ better allowing for larger penalties.

Thus, following the notation of Park and Casella [76] we propose a prior such that $\lambda_j \sim$ Gamma (r, δ_j).

$$\pi(\lambda_j) = \frac{\delta_j^r}{\Gamma(r)} (\lambda_j)^{r-1} e^{-\delta_j\lambda_j} \qquad \text{for } \lambda_j, r, \delta_j > 0 \tag{9.24}$$

The product of this specification of $\boldsymbol{\lambda}$ and the likelihood combine well in the joint distribution for the Gibbs sampling, and the specification of either r or δ allows the specification of the distribution mean tuning the amount of penalty as required.

Therefore, the final hierarchical model is

$$\pi(\boldsymbol{y}|\boldsymbol{\theta}) = \frac{1}{z(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^T s(\boldsymbol{y})} \tag{9.25}$$

$$\theta_j|\sigma^2\lambda \sim \text{Normal}(0, \sigma^2/\lambda_j), \text{for } j = 1, ..., p \tag{9.26}$$

$$\lambda_j \sim \text{ Gamma } (r, \delta_j) \tag{9.27}$$

$$\pi(\sigma^2) \propto \pi\left(\frac{1}{\sigma^2}\right) \tag{9.28}$$

9.5.2 Gibbs Sampling

Using the distributions defined in the hierarchical model, we can define the joint distribution below.

$$\begin{aligned}
\pi(\boldsymbol{y}, \boldsymbol{\theta}, \sigma^2, \boldsymbol{\lambda}) &\propto \pi(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}) \\
&\propto \pi(\boldsymbol{y}|\boldsymbol{\theta}) \prod_{j=1}^{p} \left(\pi(\theta_j|\lambda_j, \sigma^2)\pi(\lambda_j)\right) \pi(\sigma^2) \\
&\propto \frac{e^{\boldsymbol{\theta}^T s(\boldsymbol{y})}}{z(\boldsymbol{\theta})} \frac{1}{\sigma^2} \prod_{j=1}^{p} \frac{1}{\sqrt{2\pi\frac{\sigma^2}{\lambda_j}}} \exp - \left\{ \frac{\theta_j^2}{2\frac{\sigma^2}{\lambda_j}} \right\} \frac{\delta_j^r}{\Gamma(r)} (\lambda_j)^{r-1} e^{-\delta_j\lambda_j}
\end{aligned} \tag{9.29}$$

To sample from this distribution, we will update each parameter in turn by considering the terms of this distribution that depend on that parameter.

Sample σ^2

The pieces of 9.29 that involve σ^2 are below:

$$\frac{1}{\sigma^2} \prod_{j=1}^{p} \frac{1}{\sqrt{2\pi\frac{\sigma^2}{\lambda_j}}} \exp - \left\{ \frac{\theta_j^2}{2\frac{\sigma^2}{\lambda_j}} \right\} \tag{9.30}$$

$$\propto \frac{1}{\sigma^2} \left(\frac{1}{\sqrt{\sigma^2}}\right)^p \exp\left\{ -\frac{1}{\sigma^2} \sum_{j=1}^{p} \frac{\theta_j^2\lambda_j}{2} \right\} \tag{9.31}$$

$$= (\sigma^2)^{-\left(\frac{2+p}{2}\right)} \exp\left\{ -\frac{1}{\sigma^2} \left(\sum_{j=1}^{p} \frac{\theta_j^2\lambda_j}{2} \right) \right\}. \tag{9.32}$$

Comparing the simplified form to the pdf of the inverse gamma distribution:

$$\frac{B^A}{\Gamma(A)} x^{-A-1} e^{-B/x} \text{ for } A, B > 0$$

we have $\sigma^2 \sim$ Inverse Gamma $\left(\frac{p}{2}, \left(\sum_{j=1}^{p} \frac{\theta_j^2 \lambda_j}{2}\right)\right)$

Sample λ_j

In a similar method as before, we consider the components of 9.29 that depend on λ_j.

$$\frac{1}{\sqrt{2\pi \frac{\sigma^2}{\lambda_j}}} \exp\left\{-\frac{\theta_j^2}{2\frac{\sigma^2}{\lambda_j}}\right\} (\lambda_j)^{r-1} \exp\{-\delta_j \lambda_j\} \tag{9.33}$$

$$\propto \sqrt{\lambda_j}\,(\lambda_j)^{r-1} \exp\left\{-\lambda_j \frac{\theta_j^2}{2\sigma^2}\right\} \exp\{-\delta_j \lambda_j\} \tag{9.34}$$

$$\propto \lambda^{r-\frac{1}{2}} \exp\left\{-\lambda_j \left[\frac{\theta_j^2}{2\sigma^2} + \delta_j\right]\right\} \tag{9.35}$$

$$\propto \lambda^{r+\frac{1}{2}-1} \exp\left\{-\lambda_j \left[\frac{\theta_j^2}{2\sigma^2} + \delta_j\right]\right\} \tag{9.36}$$

Using the following definition of the gamma distribution with scale parameter α and rate parameter β, we have that

$$f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \tag{9.37}$$

and we find that we can estimate λ_j with Gamma $(r + \frac{1}{2}, -\frac{\theta_j^2}{2\sigma^2} + \delta_j)$.

9.6 Algorithm

Algorithm: Bayesian Ridge Exponential Random Graph Model- Gamma Prior

---

```
Require: Set the initial value for $\boldsymbol{\lambda}, \sigma^2, \boldsymbol{\gamma}$, Use ERGM to find MPLE (Maximizer to the Psuedolikelihood Function) to find initial values for $\boldsymbol{\theta}$. Denote samples of $\boldsymbol{\theta}$ in the $h$th chain, as $\boldsymbol{\theta}_h$.
while $i = 1, ..., N$ do
    while $h = 1, ..., H$ do
        1. sample $\boldsymbol{\theta}_h$ with Parallel Adaptive Direction Sampler:
            a. generate $h_1$ and $h_2$ such that $h_1 \neq h_2 \neq h$
            b. update $\boldsymbol{D}_{\boldsymbol{\tau}}^{-1}$
            c. generate $\boldsymbol{\theta}'_h$ from $\gamma(\boldsymbol{\theta}_{h_1} - \boldsymbol{\theta}_{h_2}) + \epsilon(\ddots|\boldsymbol{\theta}_h)$
            d. simulate $\boldsymbol{y}'$ from $\pi(\ddots|\boldsymbol{\theta}'_h)$
            e. update $\boldsymbol{\theta}_h \to \boldsymbol{\theta}'_h$ with the log of the probability
            $\min\left(0, [\boldsymbol{\theta}_h - \boldsymbol{\theta}'_h]^T[s(\boldsymbol{y}') - s(\boldsymbol{y})] + \log\left[\frac{\pi(\boldsymbol{\theta}'_h)}{\pi(\boldsymbol{\theta}_h)}\right]\right)$
        where $\pi(\boldsymbol{\theta}) \sim \mathscr{N}(0_p,\, \sigma^2 \boldsymbol{D_\tau})$
    2. sample $\lambda_j$ for $j = 1, 2, 3, ..$ by generating a sample from
            $\text{Gamma}\ \left(r + \frac{1}{2}, -\frac{\theta_j^2}{2\sigma^2} + \delta_j\right)$
    3. sample $\sigma^2$ by generating a sample from Inverse Gamma($\left(\frac{p}{2}, \left(\sum_{j=1}^{p} \frac{\theta_j^2 \lambda_j}{2}\right)\right)$
  end while
end while
```

9.7 Data Description

The health crisis of the COVID-19 pandemic has highlighted the need for effective local response to emerging health crises. American local health departments (LHDs) have been a subject of study for years [61]. In particular, the National Association of County and City health officials (NACCHO) with funding from the Centers for Disease Control and Prevention have conducted several comprehensive studies to understand the resources and composition of (LHDs).

Local health departments have a wide variety of jurisdictions at the city, county, and state levels serving from 1,000 to 10 million people. They provide a variety of services from immunizations to epidemiology and environmental health research and monitoring. Other LHDs perform regulation and inspection services for restaurants, schools, daycares, and public pools. This study provides details about the demographics of leadership, characteristics of staffing resources, and factors affecting emergency preparedness including staffing and funding. These studies provide an interesting look at the future challenges facing LHDs. Similar to the general population, the age of top executives is increasing, indicating that a significant changeover in leadership could occur in the coming years [61]. This has prompted a closer understanding of the connections between LHDs and their ability to share information. The following network found in the book [45] and R package [44] shows the 1283 LHDs as nodes with edges created by communication between the leadership of each LHD. Each node has attributes to indicate which state it is located in, whether or not the LHD provides tobacco use prevention programming or HIV screening, the size in millions of the serviced area, and the number of years the current leader of the LHD has been in their position.

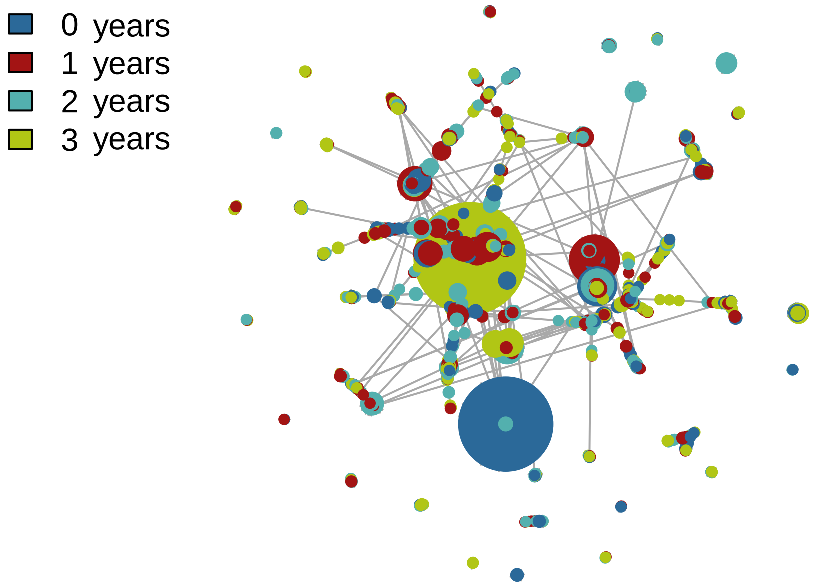


Figure 9.1: Size of node determined by the size of population served, the color of node determined by years

This data set has missing values for 27 nodes which poses problems for the subsequent models. While there are multiple options to address this issue, the simplest in this situation is to exclude these nodes. This is the option taken here.

9.8 Results

To demonstrate the effectiveness of the Ridge Bayesian exponential random graph model, we choose three network statistics. The first is the count of the number of edges, the second is the counts of homophily for the continuous variables for the population of the districts in millions, and the last is gwesp with 0.7 as the decay value. Chapter 1 describes the theory of these network statistics in more detail.

Both models were run with the same formula with the network statistics listed above. The initial settings were set to be the same at a relatively small number of the main iterations at 500, after 200 iterations of burn-in on 10 chains. The results in Table 9.2, 9.3, 9.4 show estimates for $\boldsymbol{\theta}$ that are similar for each model and both methods produce fairly small standard deviations though the new ridge model has a slightly smaller spread of estimates.

Table 9.2: Estimates for $\boldsymbol{\theta}$ by BERGM

Result for BERGM

Network Statistic	**Mean**	**SD**	**Quantiles**				
			2.5%	25%	50%	75%	97.5%
θ_1 (edges)	-7.086	0.108	-7.298	-7.155	-7.087	-7.011	-6.882
θ_2 (nodecov(population))	0.126	0.127	-0.122	0.042	0.119	0.206	0.378
θ_3 (gwesp)	2.352	0.287	1.881	2.145	2.323	2.521	2.986

Table 9.3: Estimates for $\boldsymbol{\theta}$ by Ridge BERGM with Cauchy prior

Result for Penalized Ridge

Network Statistic	**Mean**	**SD**	**Quantiles**				
			2.5%	25%	50%	75%	97.5%
θ_1 (edges)	-7.104	0.121	-7.358	-7.1880	-7.10	-7.024	-6.867
θ_2 (nodecov(population))	0.133	0.130	-0.111	0.0503	0.130	0.211	0.417
θ_3 (gwesp)	2.350	0.263	1.920	2.1603	2.32	2.501	2.938

We see here that all models converge to similar values, but both the BERGM and Ridge BERGM with Cauchy prior have a larger value for θ_3 than the result generated by the Ridge BERGM with Gamma prior.

Table 9.4: Estimates for $\boldsymbol{\theta}$ by Ridge BERGM with Gamma prior

Result for Penalized Ridge							
Network Statistic	**Mean**	**SD**	**Quantiles**				
			2.5%	25%	50%	75%	97.5%
θ_1 (edges)	-6.999	0.121	-7.239	-7.081	-6.995	-6.917	-6.770
θ_2 (nodecov(population))	0.103	0.109	-0.121	0.035	0.104	0.173	0.311
θ_3 (gwesp)	1.937	0.216	1.556	1.787	1.926	2.073	2.401

As seen in plots in Figures 9.2, 9.3, and 9.4 the process has produced centered distributions though the BERGM plot is closer to being unimodal, the trace of the estimates is stable, and the lag decreases very quickly and maintains a negligible level through the estimating process.

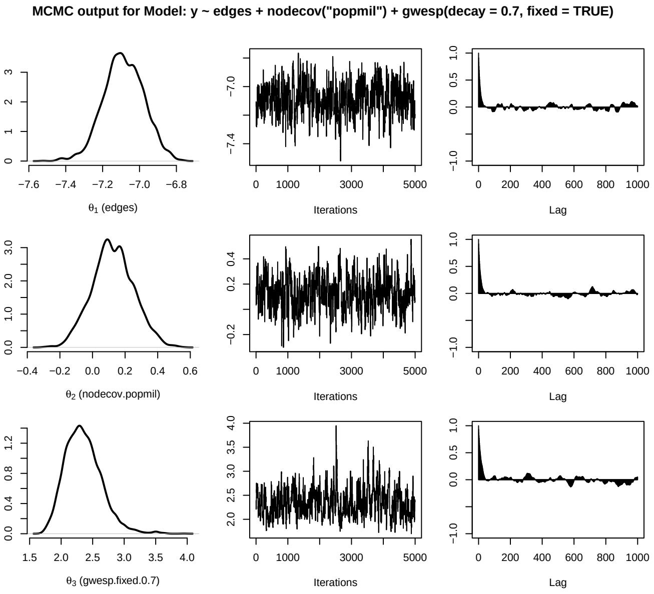


Figure 9.2: MCMC plot for BERGM

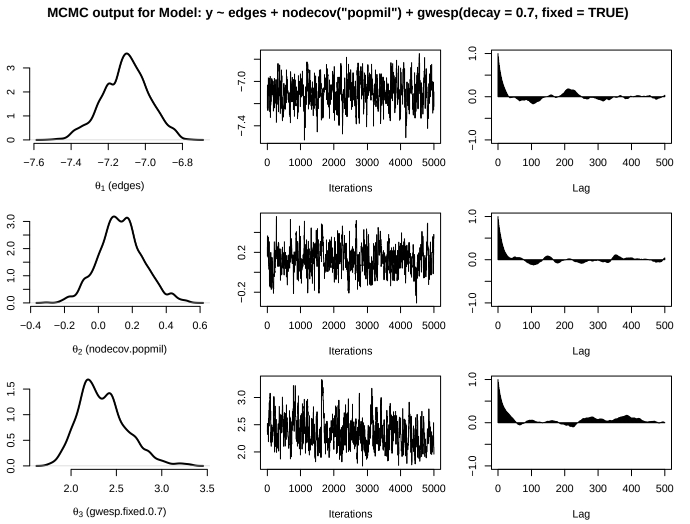


Figure 9.3: MCMC plot for Ridge with Cauchy prior

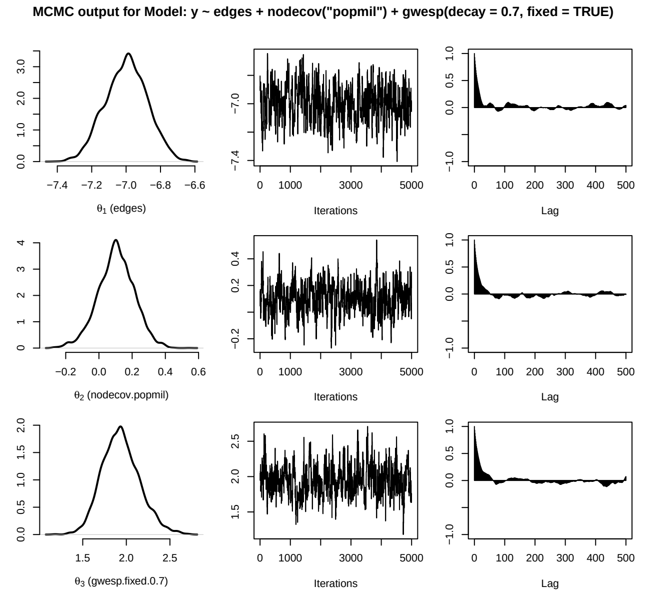


Figure 9.4: MCMC plot for Ridge with Gamma prior

To better weigh the abilities of these models we introduce the VIF function for exponential random graph models.

9.8.1 VIF function

The standard definition for Variance Inflation Factor (VIF) in the general regression model is

$$VIF_i = \frac{1}{1 - R_i^2}. \tag{9.38}$$

This quantity aids in understanding the degree of multicollinearity between variables since when $R_i^2 = 0$ or the variables are not correlated, the VIF value will be 1.

Duxbury has translated this concept to the context of exponential random graph models [18]. The following algorithm is adapted from this work to this context.

Algorithm: VIF calculation for Bayesian Exponential Random Graph Models

---

1. Generate values for $\boldsymbol{\theta}$ through BERGM/BALERGM/BARERGM
2. Simulate a large number of networks from the values found for$\boldsymbol{\theta}$
3. Count the number of each network statistic in each network creating a distribution
4. Use the distribution generated in step 3 to calculate $\boldsymbol{R}^2$
5. Calculate VIF

$$\mathbf{VIF} = \frac{1}{1 - \boldsymbol{R}^2}$$

Using the estimates found by the model in question, the simulate function generates a number of networks. For each of these networks, the network statistics are counted. This creates a list of numbers for each network statistic which can be used in the calculation of the correlation between the network statistics. For each variable y,

$$R_y^2 = r_{xx}^T R_{xx}^{-1} r_{xy}^2 \tag{9.39}$$

where x is all other network statistics, and R is the correlation matrix (excluding the edges term). Here r_{xx} is the correlation vector between all statistics except for the variable of interest y and the edges term which is excluded through this process. Once this value is found, the VIF calculation follows

$$VIF_y = \frac{1}{1 - R_y^2}. \tag{9.40}$$

The typical benchmark that VIF values greater than 5 indicate highly correlated variables is not appropriate in the context of ERGM. As found by [18], threshold values of 20 for concerning and 100 for severe collinearity.

Table 9.5: VIF results for Bayesian adaptive Ridge BERGM with both Gamma and Cauchy priors and BERGM

	Ridge: Gamma Prior			**Ridge: Cauchy Prior**			**BERGM**		
	Estimate	SD	VIF	Estimate	SD	VIF	Estimate	SD	VIF
θ_2 [a]	0.103	0.109	1.182162	0.133	0.130	48.11192	0.126	0.127	53.29528
θ_3 [b]	1.937	0.216	1.182162	2.350	0.263	48.11192	2.352	0.287	53.29528

---

[a] (nodecov(population))
[b] (gwesp)

In Table 9.5, we see the Bayesian Ridge exponential random graph model with Gamma prior effectively penalizing the estimation of the θ for the gwesp network statistic reducing the collinearity in the model.

9.9 Goodness of Fit

We can confirm the accuracy of the estimates produced by the new ridge BERGM by using Bayesian Goodness of Fit Diagnostics produced in [11]. Using the estimates for $\boldsymbol{\theta}$ found in Section 9.7, 100 graphs are generated, and their characteristics are plotted as grey histograms. The true network's values in red track with the generated values indicating effective estimations.

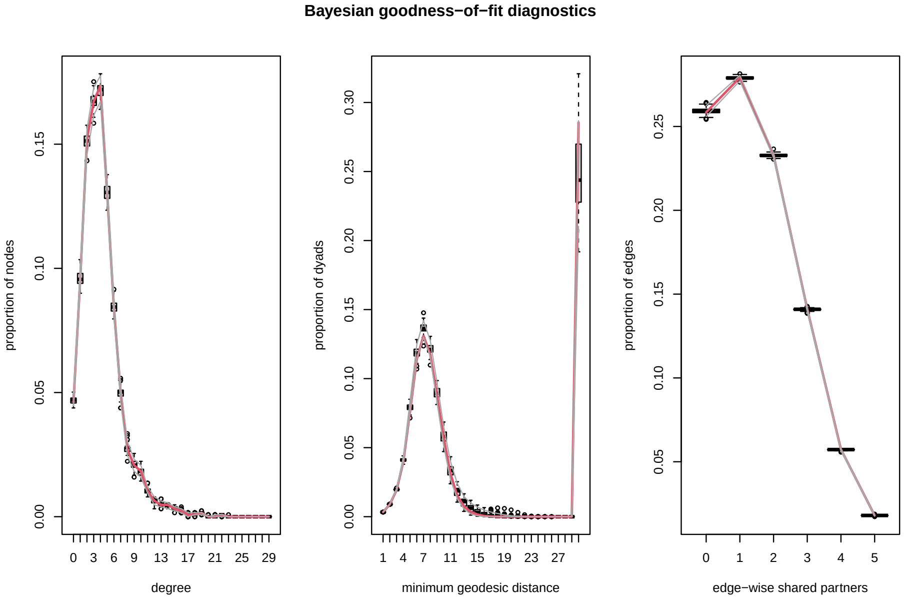


Figure 9.5: Bayesian goodness of fit diagnostics for ridge exponential random graph model

CHAPTER 10
CONCLUSIONS

The statistical family of exponential random graph models has shown to be very promising for addressing a wide range of research questions across many disciplines. Even though the modern iteration of these models is only a couple of decades old, it has become a mainstay for analyzing relational data. Despite these successes, ERGMs are plagued by degeneracy issues leaving many important networks unanalyzed. Considering the significance of networks, advancements in exponential random graph models have significant potential to shape many fields. Bayesian models, such as BERGM by Caimo and Friel et al., as well as BALERGM and BARERGM proposed in this study, effectively address the challenges of intractability and degeneracy in network analysis. Through the utilization of MCMC methods, these models circumvent the computational complexities associated with calculating the normalizing constant, ensuring efficient estimation. Furthermore, the incorporation of prior distributions aids in mitigating degeneracy, enhancing convergence and stability of the models.

While Bayesian analysis provides significant advancements to these crucial models, the five novel models presented in this study offer valuable contributions. Specifically, they address the current limitations of the BERGM model, which lacks parameter selection capabilities and strategies to handle multicollinearity. Given the frequent need for such tasks, especially in the analysis of large-scale network data, these new models provide practical advantages to fill these gaps.

The integration of Bayesian ridge and Lasso models within the network analysis framework presents a potent approach for comprehending and exploring intricate networks. By combining the strengths of these two regularization techniques, we can overcome the limitations inherent in each method and capitalize on their respective benefits. This integration yields a more comprehensive and robust modeling framework that offers enhanced insights into the underlying structures and dynamics of complex networks.

The Bayesian adaptive ridge exponential random graph model (BARERGM) provides a flexible and robust approach for handling multicollinearity and stabilizing parameter estimates. By incorporating a Gaussian prior distribution on the regression coefficients, the ridge penalty introduces shrinkage effects that effectively handle correlated nodal covariates. This leads to more reliable parameter estimates, improved predictive performance, and better generalizability of the model.

On the other hand, the Bayesian adaptive lasso exponential random graph model (BALERGM) offers an automatic variable selection mechanism, emphasizing the most relevant network parameters while diminishing the influence of less significant ones. By applying a l^1 penalty, the Lasso model promotes sparsity in parameter estimates, resulting in a more parsimonious model that aids in identifying influential factors governing network behavior. Furthermopre, the BALERGM promises to provide the possibility of fine-tuning the estimations for each parameter with individualized standard deviations allowing for a faster and more effective algorithm. This not only enhances interpretability but also improves computational efficiency by reducing the number of variables considered. The simulation and application results show that the Bayesian Adaptive Lasso ERGM is a significant improvement to the BERGM model in both improving the acceptance rate and ability to select parameters while maintaining the goodness of fit. This model provides a more accurate and reliable estimate with faster convergences, increasing the effectiveness of the answers sought in various research contexts.

The combination of Bayesian ridge and lasso models within the network analysis framework provides a comprehensive and flexible modeling approach. Researchers can leverage the strengths of both techniques to address various challenges encountered in network analysis, such as multicollinearity, variable selection, and model complexity. All the examples presented in this study were implemented using the R programming language. The code for the algorithms and examples is readily available to researchers and practitioners upon request.

Several future directions can be explored to advance the field of network analysis and improve the performance of Bayesian models like BARERGM and BALERGM. Some potential areas of focus include: (1) Extension to dynamic networks: Currently, BERGM Ridge primarily focuses on static networks. Extending the model to dynamic networks would enable the analysis of evolving network structures and relationships over time. This could involve incorporating time-varying parameters or considering temporal dependencies in the network dynamics. (2) Incorporating additional regularization techniques: further exploration can involve integrating other regularization techniques such as the Elastic Net, which combines both Lasso and ridge penalties. This would offer a more flexible approach to address various challenges in network analysis, such as variable selection, multicollinearity, and model stability.

REFERENCES

[1] Rahim Alhamzawi and Haithem Taha Mohammad Ali, *The Bayesian adaptive lasso regression*, Mathematical Biosciences **303** (2018), 75–82.

[2] M. A. Alkhamisi and G. Shukur, *A monte carlo study of recent ridge parameters*, Communications in Statistics - Simulation and Computation **36** (2007), no. 3, 535–547.

[3] David F Andrews and Colin L Mallows, *Scale Mixtures of Normal Distributions*, Journal of the Royal Statistical Society: Series B (Methodological) **36** (1974), no. 1, 99–102.

[4] Stéphanie Baggio, Victorin Luisier, and Cristina Vladescu, *Relationships between social networks and mental health*, Swiss Journal of Psychology **76** (2017), no. 1, 5–11.

[5] Eric Balkanski and Yaron Singer, *Approximation guarantees for adaptive sampling*, Proceedings of the 35th International Conference on Machine Learning (Jennifer Dy and Andreas Krause, eds.), Proceedings of Machine Learning Research, vol. 80, PMLR, 10–15 Jul 2018, pp. 384–393.

[6] Kendra R. Becker, Monika M. Stojek, Allan Clifton, and Joshua D. Miller, *Disordered eating in college sorority women: A social network analysis of a subset of members from a single sorority chapter*, Appetite **128** (2018), 180–187.

[7] P. J. Brown and J. V. Zidek, *Adaptive multivariate ridge regression*, The Annals of Statistics **8** (1980), no. 1, 64–74.

[8] A. Caimo and N. Friel, *Bayesian model selection for exponential random graph models*, Social Networks **35** (2013), no. 1, 11 – 24.

[9] Alberto Caimo, Lampros Bouranis, Robert Krause, and Nial Friel, *Statistical network analysis with bergm*, Journal of Statistical Software **104** (2022), no. 1, 1–23.

[10] Alberto Caimo and Nial Friel, *Bayesian inference for exponential random graph models*, Social Networks **33** (2011), no. 1, 41–55.

[11] Alberto Caimo and Nial Friel, *Bergm: Bayesian Exponential Random Graphs in R*, Journal of Statistical Software **61** (2014), 1–25.

[12] Alberto Caimo and Antonietta Mira, *Efficient computational strategies for doubly intractable problems with applications to bayesian social networks*, Statistics and Computing **25** (2015), no. 1, 113–125.

[13] Alberto Caimo, Francesca Pallotti, and Alessandro Lomi, *Bayesian exponential random graph modelling of interhospital patient referral networks*, Statistics in Medicine **36** (2017), no. 18, 2902–2920.

[14] Sourav Chatterjee and Persi Diaconis, *Estimating and understanding exponential random graph models*, Annals of Statistics **41** (2013), no. 5, 2428–2461.

[15] Feixiong Cheng, Junfei Zhao, Yang Wang, Weiqiang Lu, Zehui Liu, Yadi Zhou, William R. Martin, Ruisheng Wang, Jin Huang, Tong Hao, and et al., *Comprehensive characterization of protein–protein interactions perturbed by disease mutations*, Nature Genetics **53** (2021), no. 3, 342–353.

[16] Raj S. Chhikara and Leroy Folks, *The inverse gaussian distribution: Theory, methodology, and applications*, CRC Press, 1988.

[17] A. P. Dempster, Martin Schatzoff, and Nanny Wermuth, *A simulation study of alternatives to ordinary least squares*, Journal of the American Statistical Association **72** (1977), no. 357, 77–91.

[18] Scott W. Duxbury, *Diagnosing multicollinearity in exponential random graph models*, Ph.D. thesis, Jul 2018, p. 491–530.

[19] Hanan Duzan and Nurul Sima Mohamad Shariff, *Ridge regression for solving the multicollinearity problem: Review of methods and models*, Journal of Applied Sciences **15** (2015), 392–404.

[20] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani, *Least angle regression*, The Annals of statistics **32** (2004), no. 2, 407–499.

[21] P. Erdös and A. Rényi, *On random graphs I*, Publicationes Mathematicae Debrecen **6** (1959), 290.

[22] Jianqing Fan, Yang Feng, and Yichao Wu, *Network exploration via the adaptive LASSO and SCAD penalties*, The Annals of Applied Statistics **3** (2009), no. 2, 521 – 541.

[23] Jianqing Fan and Runze Li, *Variable selection via nonconcave penalized likelihood and its oracle properties*, Journal of the American statistical Association **96** (2001), no. 456, 1348–1360.

[24] Ildiko E. Frank and Jerome H. Friedman, *A statistical view of some chemometrics regression tools*, Technometrics **35** (1993), no. 2, 109–135.

[25] Ove Frank and David Strauss, *Markov Graphs*, Journal of the American Statistical Association **81** (1986), no. 395, 832–842.

[26] Nial Friel, Anthony Pettitt, R. Reeves, and E. Wit, *Bayesian Inference in Hidden Markov Random Fields for Binary Data Defined on Large Lattices*, Journal of Computational and Graphical Statistics **18** (2009), 243–261.

[27] Wenjiang J. Fu, *Penalized regressions: The bridge versus the lasso*, Journal of Computational and Graphical Statistics **7** (1998), no. 3, 397–416.

[28] Alan E. Gelfand and Adrian F. M. Smith, *Sampling-based approaches to calculating marginal densities*, Journal of the American Statistical Association **85** (1990), no. 410, 398–409.

[29] Stuart Geman and Donald Geman, *Stochastic relaxation, gibbs distributions, and the bayesian restoration of images*, IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-6** (1984), no. 6, 721–741.

[30] Charles J. Geyer, *Markov chain monte carlo maximum likelihood*, 1991.

[31] Diane Galarneau Gibbons, *A simulation study of some ridge estimators*, Journal of the American Statistical Association **76** (1981), no. 373, 131–139.

[32] W. R. Gilks, G. O. Roberts, and E. I. George, *Adaptive direction sampling*, Journal of the Royal Statistical Society. Series D (The Statistician) **43** (1994), no. 1, 179–189.

[33] Camelia Goga and Muhammad Shehzad, *Overview of ridge regression estimators in survey sampling*, (2012).

[34] Gene H. Golub, Michael Heath, and Grace Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics **21** (1979), no. 2, 215–223.

[35] Steven M. Goodreau, Mark S. Handcock, David R. Hunter, Carter T. Butts, and Martina Morris, *A statnet Tutorial*, Journal of Statistical Software **24** (2008), no. 9, 1–26.

[36] Yves Grandvalet and Stéphane Canu, *Outcomes of the equivalence of adaptive ridge with least absolute shrinkage*, NIPS, 1998.

[37] Mark S. Granovetter, *The strength of weak ties*, American Journal of Sociology **78** (1973), no. 6, 1360–1380.

[38] Peter J. Green, *Reversible jump markov chain monte carlo computation and bayesian model determination*, Biometrika **82** (1995), no. 4, 711–732.

[39] Ester Gutiérrez-Moya, Sebastián Lozano, and Belarmino Adenso-Díaz, *Analysing the structure of the global wheat trade network: An ergm approach*, Agronomy **10** (2020), no. 12.

[40] Min Jin Ha and Shannon T. Holloway, *Ggmridge: Gaussian graphical models using ridge penalty followed by thresholding and reestimation*, 2022, R package version 1.2.

[41] Min Jin Ha and Wei Sun, *Partial correlation matrix estimation using ridge penalty followed by thresholding and re-estimation*, Biometrics **70** (2014), no. 3, 762–770.

[42] Mark S Handcock, *Assessing degeneracy in statistical models of social networks*, Tech. report, Working paper, 2003.

[43] Mark S. Handcock, David R. Hunter, Carter T. Butts, Steven M. Goodreau, Pavel N. Krivitsky, and Martina Morris, *ergm: Fit, simulate and diagnose exponential-family models for networks*, The Statnet Project, 2023, R package version 4.5.0.

[44] Jenine K. Harris, *ergmharris: Local health department network data set*, 2013, R package version 1.0.

[45] Jenine K. Harris, *An introduction to exponential random graph modeling*, SAGE, 2014.

[46] W. K. Hastings, *Monte carlo sampling methods using markov chains and their applications*, Biometrika **57** (1970), no. 1, 97–109.

[47] A.E. Hoerl, *Application of ridge analysis to regression problems*, Chemical Engineering Progress **58** (1962), no. 3, 54–59.

[48] Arthur E. Hoerl, Robert W. Kannard, and Kent F. Baldwin, *Ridge regression:some simulations*, Communications in Statistics **4** (1975), no. 2, 105–123.

[49] Arthur E. Hoerl and Robert W. Kennard, *Ridge regression: Biased estimation for nonorthogonal problems*, Technometrics **12** (1970), no. 1, 55–67.

[50] Paul W. Holland and Samuel Leinhardt, *An exponential family of probability distributions for directed graphs*, Journal of the American Statistical Association **76** (1981), no. 373, 33–50.

[51] T. C. Hsiang, *A bayesian view on ridge regression*, Journal of the Royal Statistical Society. Series D (The Statistician) **24** (1975), no. 4, 267–268.

[52] David Hunter, Mark Handcock, Carter Butts, Steven Goodreau, and Martina Morris, *ergm: A package to fit, simulate and diagnose exponential-family models for networks*, Journal of Statistical Software, Articles **24** (2008), no. 3, 1–29.

[53] David R. Hunter, *Curved exponential family models for social networks*, Social Networks **29** (2007), no. 2, 216–230.

[54] Karin Ingold and Philip Leifeld, *Structural and Institutional Determinants of Influence Reputation: A Comparison of Collaborative and Adversarial Policy Networks in Decision Making and Implementation*, Journal of Public Administration Research and Theory **26** (2014), no. 1, 1–18.

[55] Ghadban Khalaf and Ghazi Shukur, *Choosing ridge parameter for regression problems*, Communications in Statistics-theory and Methods - COMMUN STATIST-THEOR METHOD **34** (2005), 1177–1182.

[56] B. M. Golam Kibria, *Performance of some new ridge regression estimators*, Communications in Statistics - Simulation and Computation **32** (2003), no. 2, 419–435.

[57] Christina Kiel, Pedro Beltrao, and Luis Serrano, *Analyzing protein interaction networks using structural information*, Annual Review of Biochemistry **77** (2008), no. 1, 415–441, PMID: 18304007.

[58] Johan Koskinen, Garry Robbins, and Dean Lusher, *Exponential random graph models for social networks theory, methods, and applications*, Cambridge University Press, 2013.

[59] Pavel N. Krivitsky, David R. Hunter, Martina Morris, and Chad Klumb, *ergm 4: New features for analyzing exponential-family random graph models*, Journal of Statistical Software **105** (2023), no. 6, 1–44.

[60] Jerald F. Lawless and P. C. Wang, *A simulation study of ridge and other regression estimators*, Communications in Statistics-theory and Methods **5** (1976), 307–323.

[61] Carolyn J. Leep, *National profile of local health departments, 2010*, ICPSR Data Holdings (2012).

[62] Chenlei Leng, Minh-Ngoc Tran, and David Nott, *Bayesian adaptive lasso*, Annals of the Institute of Statistical Mathematics **66** (2014), no. 2, 221–244.

[63] Richard A. Levine and George Casella, *Implementations of the monte carlo em algorithm*, Journal of Computational and Graphical Statistics **10** (2001), no. 3, 422–439.

[64] Dean Lusher, Johan Koskinen, and Garry Robins, *Exponential random graph models for social networks: Theory, methods, and applications*, Cambridge University Press, 2013.

[65] Donald W. Marquardt and Ronald D. Snee, *Ridge regression in practice*, The American Statistician **29** (1975), no. 1, 3–20.

[66] Gary C. McDonald and Diane I. Galarneau, *A monte carlo evaluation of some ridge-type estimators*, Journal of the American Statistical Association **70** (1975), no. 350, 407–416.

[67] Nicolai Meinshausen and Peter Bühlmann, *High-dimensional graphs and variable selection with the lasso*, The Annals of Statistics **34** (2006), no. 3, 1436–1462.

[68] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller, *Equation of state calculations by fast computing machines*, The Journal of Chemical Physics **21** (1953), no. 6, 1087–1092.

[69] J. L. Moreno and H. H. Jennings, *Statistics of social configurations*, Sociometry **1** (1938), no. 3/4, 342–374.

[70] Martina Morris, Mark S. Handcock, and David R. Hunter, *Specification of Exponential-Family Random Graph Models: Terms and Computational Aspects*, Journal of Statistical Software **24** (2008), no. 4, 1–24.

[71] Gisela Muniz and B M Golam Kibria, *On some ridge regression estimators: An empirical comparisons*, Communications in Statistics - Simulation and Computation **38** (2009), 621–630.

[72] James Raymond Munkres and Lorraine Davis, *Topology*, Pearson Prentice Hall, 2018.

[73] Iain Murray, Zoubin Ghahramani, and David MacKay, *MCMC for doubly-intractable distributions*, 2012.

[74] J. Møller, A. N. Pettitt, R. Reeves, and K. K. Berthelsen, *An efficient markov chain monte carlo method for distributions with intractable normalising constants*, Biometrika **93** (2006), no. 2, 451–458.

[75] Hung Nguyen, Duc Tran, Bang Tran, Bahadir Pehlivan, and Tin Nguyen, *A comprehensive survey of regulatory network inference methods using single cell RNA sequencing data*, Briefings in Bioinformatics **22** (2020), no. 3, bbaa190.

[76] Trevor Park and George Casella, *The Bayesian Lasso*, Journal of the American Statistical Association **103** (2008), no. 482, 681–686.

[77] Nicholas G. Polson and James G. Scott, *On the Half-Cauchy Prior for a Global Scale Parameter*, Bayesian Analysis **7** (2012), no. 4, 887 – 902.

[78] Jialun Qin, Jennifer J. Xu, Daning Hu, Marc Sageman, and Hsinchun Chen, *Analyzing terrorist networks: A case study of the global salafi jihad network*, Intelligence and Security Informatics (Berlin, Heidelberg) (Paul Kantor, Gheorghe Muresan, Fred Roberts, Daniel D. Zeng, Fei-Yue Wang, Hsinchun Chen, and Ralph C. Merkle, eds.), Springer Berlin Heidelberg, 2005, pp. 287–304.

[79] R Core Team, *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria, 2021.

[80] M. Resnick, P. Bearman, R. Blum, K. Bauman, K. Harris, J. Jones, J. Tabor, T. Beuhring, R. Sieving, M. Shew, M. Ireland, L. Bearinger, and J. Udry, *Protecting adolescents from harm. Findings from the National Longitudinal Study on Adolescent Health.*, JAMA **278 10** (1997), 823–32.

[81] G. O. Roberts, A. Gelman, and W. R. Gilks, *Weak convergence and optimal scaling of random walk metropolis algorithms*, The Annals of Applied Probability **7** (1997), no. 1, 110–120.

[82] G.O. Roberts and W.R. Gilks, *Convergence of adaptive direction sampling*, Journal of Multivariate Analysis **49** (1994), no. 2, 287–298.

[83] Samuel F. Sampson, *A novitiate in a period of change: An experimental and case study of social relationships*, Ph.D. thesis, Cornell university, 1968.

[84] Fadil Santosa and William W. Symes, *Linear inversion of band-limited reflection seismograms*, SIAM Journal on Scientific and Statistical Computing **7** (1986), no. 4, 1307–1330.

[85] Ali Shojaie, *Link prediction in biological networks using multi-mode exponential random graph models*, 11th Workshop on Mining and Learning with Graphs, Citeseer, 2013, pp. 987–991.

[86] Ali Shojaie, Sumanta Basu, and George Michailidis, *Adaptive thresholding for reconstructing regulatory networks from time-course gene expression data*, Statistics in Biosciences **4** (2012), no. 1, 66–83.

[87] Ali Shojaie and George Michailidis, *Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs*, Biometrika **97** (2010), no. 3, 519–538.

[88] Victor Solo, Jean-Baptiste Poline, Martin A. Lindquist, Sean L. Simpson, F. DuBois Bowman, Moo K. Chung, and Ben Cassidy, *Connectivity in fMRI: Blind spots and Breakthroughs*, IEEE Transactions on Medical Imaging **37** (2018), no. 7, 1537–1550.

[89] Alex Stivala and Alessandro Lomi, *Testing biological network motif significance with exponential random graph models*, Applied Network Science **6** (2021), no. 1.

[90] David Strauss and Michael Ikeda, *Pseudolikelihood estimation for social networks*, Journal of the American statistical association **85** (1990), no. 409, 204–212.

[91] Cajo JF Ter Braak and Jasper A Vrugt, *Differential evolution markov chain with snooker updater and fewer chains*, Statistics and Computing **18** (2008), no. 4, 435–446.

[92] The US Burden of Disease Collaborators, *The State of US Health, 1990-2016: Burden of Diseases, Injuries, and Risk Factors Among US States*, JAMA **319** (2018), no. 14, 1444–1472.

[93] Robert Tibshirani, *Regression Shrinkage and Selection via the Lasso*, Journal of the Royal Statistical Society. Series B (Methodological) **58** (1996), no. 1, 267–288.

[94] Luke Tierney, *Markov chains for exploring posterior distributions*, The Annals of Statistics **22** (1994), no. 4, 1701–1728.

[95] Maksim Tsvetovat and Kathleen Carley, *Structural Knowledge and Success of Anti- Terrorist Activity: The Downside of Structural Equivalence*, (2005).

[96] Marijtje A.J. van Duijn, Krista J. Gile, and Mark S. Handcock, *A framework for the comparison of maximum pseudo-likelihood and maximum likelihood esti-*

mation of exponential family random graph models, Social Networks **31** (2009), no. 1, 52–62.

[97] Sara van Erp, Daniel L. Oberski, and Joris Mulder, *Shrinkage priors for bayesian penalized regression*, Journal of Mathematical Psychology (2018).

[98] George G. Vega Yon, Andrew Slaughter, and Kayla de la Haye, *Exponential random graph models for little networks*, Social Networks **64** (2021), 225–238.

[99] Matthew P. Wand, John T. Ormerod, Simone A. Padoan, and Rudolf Frühwirth, *Mean field variational bayes for elaborate distributions*, Bayesian Analysis **6** (2011), no. 4.

[100] Hansheng Wang and Chenlei Leng, *A note on adaptive group lasso*, Computational statistics & data analysis **52** (2008), no. 12, 5277–5286.

[101] Zeya Wang, Veerabhadran Baladandayuthapani, Ahmed O. Kaseb, Hesham M. Amin, Manal M. Hassan, Wenyi Wang, and Jeffrey S. Morris, *Bayesian edge regression in undirected graphical models to characterize interpatient heterogeneity in cancer*, Journal of the American Statistical Association **117** (2022), no. 538, 533–546.

[102] Nigel L. Williams and Dean Hristov, *An examination of DMO network identity using Exponential Random Graph Models*, Tourism Management **68** (2018), 177–186.

[103] Xiaofan Xu and Malay Ghosh, *Bayesian variable selection and estimation for group lasso*, Bayesian Analysis **10** (2015), no. 4, 909–936.

[104] Ming Yuan and Yi Lin, *Model selection and estimation in regression with grouped variables*, Journal of the Royal Statistical Society: Series B (Statistical Methodology) **68** (2006), no. 1, 49–67.

[105] Hui Zou, *The Adaptive Lasso and its Oracle Properties*, Journal of the American statistical association **101** (2006), no. 476, 1418–1429.

CURRICULUM VITAE

Vicki Modisette

Education

University of Louisville Louisville, Kentucky
Ph.D. in Applied and Industrial Mathematics Expected August 2023
Advisor: Dr. Dan Han

University of Louisville Louisville, Kentucky
MA in Mathematics December 2019
summa cum laude
Mentor: Dr. Xuwen Zhu

University of Louisville Louisville, Kentucky
MM in Harp Performance May 2017
summa cum laude
Mentor: Carol McClure

Union University Jackson, Tennessee
BA in Mathematics and Music May 2015
summa cum laude
Mentors: Dr. Matt Lunsford, Carol McClure

Teaching

Graduate Teaching Assistant, University of Louisville
Fall 2015-present

Adjunct Instructor, Campbellsville University
Fall 2020-present

Talks and Papers

Penalized Bayesian Inference on the Topological Characteristics of the Network
January 2023
Joint Mathematics Meetings 2023

The Theory Behind a New Adaptive Bayesian Lasso Exponential Random Graph Model
February 2022
University of Louisville

Analyzing the Infection Risk of Healthcare Workers for COVID-19 by new Adaptive Bayesian Lasso ERGM
July 2021
2021 SIAM Annual Meeting

Skills

Programming
Proficient in: R

Service and outreach

The Graduate Network in Arts and Sciences
August 2019-May 2020
Representative

Local Chapter of American Mathematical Society
August 2018-May 2020
Treasurer and Vice-president