COMMUNITIES IN SOCIAL NETWORKS: DETECTION, HETEROGENEITY AND EXPERIMENTATION

by

Heather Mathews

Department of Statistical Science
Duke University

Date: _____
Approved:

Alexander Volfovsky, Supervisor

Galen Reeves

Amy H. Herring

James Moody

Dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the Department of Statistical Science
in the Graduate School of
Duke University

2022

ABSTRACT

COMMUNITIES IN SOCIAL NETWORKS: DETECTION, HETEROGENEITY AND EXPERIMENTATION

by

Heather Mathews

Department of Statistical Science
Duke University

Date: _____
Approved:

Alexander Volfovsky, Supervisor

Galen Reeves

Amy H. Herring

James Moody

An abstract of a dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the Department of Statistical Science
in the Graduate School of
Duke University

2022

Copyright © 2022 by Heather Mathews
All rights reserved

Abstract

The study of network data in the social and health sciences frequently concentrates on understanding how and why connections form. In particular, the task of determining latent mechanisms driving connection has received a lot of attention across statistics, machine learning, and information theory. In social networks, this mechanism often manifests as community structure. As a result, this work provides methods for discovering and leveraging these communities to better understand networks and the data they generate.

We provide three main contributions. First, we present methodology for performing community detection in challenging regimes. Existing literature has focused on modeling the spectral embedding of a network using Gaussian mixture models (GMMs) in scaling regimes where the ability to detect community memberships improves with the size of the network. However, these regimes are not very realistic. As such, we provide tractable methodology motivated by new theoretical results for networks with non-vanishing noise by using GMMs that incorporate truncation and shrinkage effects.

Further, when covariate information is available, often we want to understand how covariates impact connections. It is likely that the effects of covariates on edge formation differ between communities (e.g. age might play a different role in friendship formation in communities across a city). To address this issue, we introduce a latent space network model where coefficients associated with certain covariates can depend on latent community membership of the nodes. We show that ignoring such structure can lead to either over- or under-estimation of covariate importance to edge formation and propose a Markov Chain Monte Carlo approach for simultaneously learning the latent community structure and the community specific coefficients.

Finally, we consider how community structure can impact experimentation. It

is evident that communities can act in different ways, and it is natural that this propagates into experimental design. As a result, this observation motivates our development of community informed experimental design. This design recognizes that information between individuals likely flows along within community edges rather than across community edges. We demonstrate that this design improves estimation of global average treatment effect, even when the community structure of the graph needs to be estimated.

Dedicated to my grandpa, John Mathews

Acknowledgements

This entire dissertation is based on the importance of connection. Before diving into the theory behind social connection, how we model it, how we study it, and more, I would like to share my deep appreciation for the connections in my life that have made this work possible.

First and foremost, thank you to my advisor, Alexander Volfovsky, for his constant patience, guidance, and encouragement throughout this entire program. It is hard to put into words how happy I am to have worked with him over the years. I thank him for introducing me to the world of networks and community detection. Further, not only is he incredibly talented in his field, but he is also a wonderful advisor. The impact of a mentor can be quite great, and I will forever be grateful for Alex and everything that he has done for me.

Thank you to Galen Reeves for being my co-advisor throughout this program. It has been an honor to work with such amazing and talented faculty here. I would like to share my appreciation for Amy Herring and Jim Moody for being on my committee. Also, thank you to Chantal Gil for taking the time to chat with me and support me as well. Thank you to those of the Duke Statistical Science department who supported me over the years.

I want to give an incredibly special thank you to my parents. I am so grateful for their endless support, for always being there for me, for making me laugh when I wanted to cry, and for showing me so much love. I have to especially thank my mom for all of the many, many phone calls.

I am very thankful for all of the friendships that I have made here at Duke with a special thank you to Vittorio, Evan, Ezinne, Michele, Federico, Jordan, Alessandro, and Jesper. They all have been so supportive and have filled my time here with

many wonderful memories including pasta dinners, video game nights, late night work sessions, and so much more. Thank you to my lab groups over the years for providing positive environments to practice talks, share ideas, and collaborate.

One incredibly special aspect about this PhD was that I got to make amazing friends all around the world. Thank you to Jeremias and Jean in particular for the extra support, friendship, and guidance. I appreciate all the video chats across many time zones that helped me through.

I want to also thank those that inspired me to pursue a PhD. This all started with my high school statistics teacher, Brian Meeron, who encouraged me to major in mathematics and statistics. I want to also share my deep appreciation for John Bailer who mentored me throughout my time at Miami University and encouraged me to pursue a PhD. Also, thank you to Olga Brezhneva who fostered a great deal of the love and appreciation I have for mathematics.

Thank you to my friends from back home, especially AJ Tatro, Ricardo Treviño, Kirk Shedlock, Christy Loke, Andrea Reeves, Carly Francis, and my family, especially my Aunt Terry, Aunt Jeanne, and my grandparents for all of their support and love. A huge thank you to my little healthcare team: Eliza Sholtz Russ, Alex Hussey, and Christine Pavey. Thank you to Whitney Simmons, Katie Austin, and the Tone It Up community for providing supportive and inspiring workout communities, especially during the pandemic. Thank you to Taylor Swift for coming out with so many albums over the past 5 years and to the Starbucks that opened up on 9th street for all the extra energy needed to finish this.

I am also so grateful to have spent a summer at Netflix. Being an intern with them was such an amazing opportunity that allowed me to grow as a data scientist. Specifically, thank you to Ted, Vickie, Sui, Martin, and Brittany.

Graduate school has been quite the adventure. I have learned that I am capable

of much more than I thought, however I could not have done it alone. I am incredibly lucky to have so many amazing people in my life who have supported me throughout this program. I truly cannot express how much appreciation I have for them. Not only have they all helped me to grow as a statistician and researcher, but also as a person.

Contents

Abstract **iv**

Acknowledgements **vii**

List of Figures **xiv**

List of Tables **xviii**

1 Introduction **1**

1.1 Network Notation 3

1.2 The Stochastic Blockmodel 4

2 Gaussian Mixture Models for Stochastic Block Models with Non-Vanishing Noise **6**

2.1 Introduction 6

2.2 Spectral Theory 6

2.2.1 Distributions of Eigenvalues 7

2.2.2 Relation between the Spiked Wigner Model and the SBM . . . 8

2.2.3 Recoverability of Communities 12

2.2.4 Information in Eigenvectors 14

2.3 Novel Contribution 18

2.4 Problem Formulation 18

2.5 Theory and Methodology 21

2.5.1 Gaussian Approximation of Eigenvectors 21

2.5.2 Proposed Method 24

2.6 Experimental Results 26

2.6.1 Numerical Simulations 26

2.6.2 Real-World Data Analysis 28

2.7 Discussion 29

3 Latent Community Adaptive Network Regression 30

3.1 Introduction 30

3.1.1 The AddHealth Data 32

3.1.2 Notation and Model Setup 34

3.2 Extending the AMEN Model 35

3.2.1 The Model 36

3.2.2 Relationship between $\tilde{\beta}$ in Eq (3.5) and β in Eq (3.6) 37

3.3 Bayesian Computation 38

3.3.1 Priors 40

3.3.2 Computation 40

3.3.3 Posterior Inference and Summarization 44

3.3.4 Extending to the Censored Binary Likelihood 45

3.4 Simulations and Computational Considerations 46

3.4.1 Binary Likelihood with Community Dependent and Independent Covariates 47

3.4.2 Initializing U 49

3.4.3 Effect of Censoring 54

3.5 Application to AddHealth Data 55

3.5.1 School A 57

3.5.2 School B 58

3.6 Discussion 62

4 Community Informed Experimental Design 63

4.1 Introduction 63
4.1.1 Classical Assumptions and Designs 66
4.1.2 Our Approach: Community Interference Assumptions and Design 68
4.2 Methods 70
4.3 Bias Reduction 74
4.4 Cost of Misclustered Nodes 78
4.5 Simulations 80
4.5.1 Two Communities 81
4.5.2 Community Level Average Treatment Effect 83
4.5.3 Varying Number of Communities 85
4.5.4 No Community Structure nor Interference 86
4.5.5 Potentially Mismatched Communities 87
4.6 Discussion 89

5 Conclusion and Future Work 91

5.1 Community Detection with Multiple Networks 91
5.2 Extensions for Latent Community Network Regression 93
5.3 Other Estimators for Community Informed Design 94

A Spectral Theory and SBM 95

A.1 Parameter Estimation for the SBM 95
A.2 Phase Transitions for Whitened Representation of the SBM 96
A.3 Additional Real Data Example 96

B Network Regression 99

B.1 Vectorization of $\mathrm{diag}(\tilde{\beta}_r U^T) X_r$ 99

B.2 Including Dyadic Covariates 100
B.3 General Dyadic Covariates 102
B.4 Bivariate Probit Likelihood 104
B.5 Comparison of Censored Binary Formulations 105
B.6 Covariate Information for AddHealth 106
B.7 Trace plots for AddHealth Networks 107
B.8 Dyadic Posterior Credible Intervals for AddHealth Networks 107
B.8.1 School A 108
B.8.2 School B 110
B.9 Covariate Assisted Clustering 113

C Experimentation **114**
C.1 Expected Cost of Misclustering 114
C.2 Exposure Probabilities with Community Interference Under 3-Net . . 117
C.3 Variance of Difference in Means Estimator 118
C.4 Algorithms 122
C.5 Generative Model from Ugander et al 122
C.6 Additional Figures 124

Bibliography **127**

Biography **136**

List of Figures

1.1 Network generated from ER vs SBM 5

2.1 Distribution of eigenvalues for spiked Wigner model 9

2.2 Recovery of the signal in a spiked Wigner Model 9

2.3 Matrix representation of 3 SBM networks with varying signal 10

2.4 Distribution of eigenvalues for 3 community SBM 11

2.5 Threshold for misclustering rate with 2 community SBM 13

2.6 Comparison of separation on first eigenvector for SBMs with community balanced vs unbalanced degree 15

2.7 Eigenvectors of 3 community SBM 16

2.8 Example of how eigendecomposition is invariant to node order (up to sign change) for a network 17

2.9 Performance of k-means on SBM for community label recovery 17

2.10 Aligned eigenvectors for SBMs with varying signal levels using the whitened representation 19

2.11 Visual representation of phase transitions in the eigenvectors 23

2.12 Visual representation of covariance convergence as d grows with N . . 25

2.13 Simulation results for GMM community detection 27

3.1 95% CI for β estimates when fitting AMEN model on AddHealth network. 33

3.2 Running separate AMEN models on school network partitioned by initial estimated latent communities. 33

3.3 Simulation performance for community dependent row and column covariates under different models 48

3.4 Simulation performance for community independent row and column covariates under different models 48

3.5 Estimating U using different initialization techniques 52

3.6 Median ACF for $U_i\tilde{\beta}$ across $i \in \{1, ..., 150\}$. 53

3.7 Computational performance for community adaptive network regression 54

3.8 95% Credible intervals for row and column covariates from censored community dependent model. 55

3.9 School A: 95% CI for row and column covariates with community dependent model, $K = 2$. 58

3.10 School B: 95% CI for row and column covariates with community dependent model, $K = 2$. 59

3.11 School A: Comparing 95% CI $\sum_{k=1}^{2} \frac{\tilde{\beta}_k}{2}$ vs. AMEN with multiplicative effects of dimension 2 for row and column covariates. 60

3.12 School B: Comparing 95% CI $\sum_{k=1}^{2} \frac{\tilde{\beta}_k}{2}$ vs. AMEN with multiplicative effects of dimension 2 for row and column covariates. 61

4.1 Ideal scenario for GATE estimation 64

4.2 Neighborhood interference potential outcomes 65

4.3 Desirable potential outcomes under community interference 69

4.4 Example clusterings under different experimental designs 72

4.5 Simulated exposure probabilities under various designs 78

4.6 Relative bias for 2 equally sized community SBM 83

4.7 Relative bias for 2 equally sized community SBM with homogenous neighborhood interference 84

4.8 Relative bias for dis-assortative community interference 84

4.9 Relative bias for community level treatment effects for 2 equally sized community SBM 85

4.10 Relative bias for SBM with varying K values 86

4.11 Relative bias for network with no community interference 87

4.12 Relative bias for various schools from the middle school data set . . . 89

5.1 Plots of eigenvectors of sums and differences of two adjacency matrices 93

A.1 MSE plots for estimating p and Q 96

A.2 Empirical demonstration of phase transitions for recoverability for the whitened representation of the SBM 97

A.3 Misclustering rates for various algorithms implemented on 4 middle schools 98

B.1 Comparing censored binary models 105

B.2 Convergence diagnostics for AddHealth 107

B.3 School A: 95% CI for dyadic grade covariates with community dependent model, $K = 2$ 108

B.4 School A: 95% CI for dyadic smoking covariates with community dependent model, $K = 2$ 109

B.5 School A: 95% CI for dyadic drinking covariates with community dependent model, $K = 2$ 109

B.6 School A: 95% CI for dyadic race covariates with community dependent model, $K = 2$ 110

B.7 School B: 95% CI for dyadic grade covariates with community dependent model, $K = 2$ 111

B.8 School B: 95% CI for dyadic smoking covariates with community dependent model, $K = 2$ 111

B.9 School B: 95% CI for dyadic drinking covariates with community dependent model, $K = 2$ 112

B.10 School B: 95% CI for dyadic race covariates with community dependent model, $K = 2$ 112

C.1 Comparing observed and expected cost from mispaired nodes 117

C.2 Relative bias for equal sized two community simulation using one hop max algorithm 124

C.3 Relative RMSE for experiment with equal sized two community simulation 124

C.4 Example of a middle school network colored by estimated community label 125

C.5 Combining recoverability thresholds and bias due to misclustering . . 126

List of Tables

3.1 Models considered for community adaptive network regression simulations 47

3.2 Comparing percent of students with a particular response by estimated latent community 56

Chapter 1

Introduction

Our world has become increasingly driven by networks (Dunbar et al., 2015; Farine and Whitehead, 2015; Grinberg et al., 2019; Stadtfeld et al., 2019). Whether they are the social networks we form on and offline, those generated by computers, airplanes, and more, we are constantly impacted by connection. In this work we focus on exploring social networks as they provide relevant insight into how humans interact with one another and how information propagates between them. These tasks can help identify harmful groups of people, at risk individuals, and heterogenous behaviors in connection formation.

Working with network data is not a trivial task. With their presence comes the need to understand the fundamentals of networks from a mathematical perspective to accurately analyze the data they generate. The most common complexity that networks introduce is their strong correlation structure that violates most assumptions underlying standard statistical techniques. As such, there has been the need for development of new methods and theory to accommodate them. Further, the way in which connections form is often intricate and thus understanding this process is a fundamental goal of network analysis. In particular, this work concentrates on illuminating the latent mechanisms that drive connections between individuals and exemplifying the substantial impact of them. When only the network structure is available, deriving meaningful conclusions about connection can be challenging. However, often these latent mechanisms that contribute to network formation can be inferred from the network itself (Fortunato, 2010; Girvan and Newman, 2002; McPherson et al., 2001). Community detection aims to uncover these latent structures

and forms the basis of this work. To motivate the importance of latent communities, note that for social networks, homophily is extremely prevalent (Igarashi et al., 2005; Shrum et al., 1988). That is, individuals tend to be friends with people who are similar to them and cluster together to form 'communities'. These communities allow us to identify and to better understand meaningful subgroups of individuals.

Chapter 2 presents the fundamentals of spectral theory needed to understand community detection algorithms and provides a novel approach to community detection. We present methodology using Gaussian Mixture Models on a lower dimensional spectral embedding of a network. We show how our method allows for community label recovery in more challenging (and more realistic) regimes than what is typically explored in the literature.

In Chapter 3, we consider having access to an observed network and nodal covariate information. As mentioned, network analysis often aims to understand why people connect. Combining covariate information and community detection provides an opportunity to understand these processes in more depth. To do this, we derive an extension of the Additive and Multiplicative Effects Network regression model (Hoff et al., 2013) where we allow for simultaneous estimation of community memberships and community dependent coefficients. This allows recovery of not only community labels but also of the extent to which covariates impact these communities differently. This model is applied to a high school data set that provides meaningful insights as to how different communities of high school students are driven to connect in different ways.

Chapters 2 and 3 demonstrate how community structure tends to motivate heterogenous behaviors between groups of people. In Chapter 4, this notion is extended to a causal setting where we consider how communities can be influenced by interventions in different ways. Companies heavily rely on A/B testing to make important business

decisions. Consider testing a new feature on an app. Rather than simply deploying the new feature to all users, an experiment is conducted where some users are exposed to the new feature and some are not. After the experiment, an outcome of interest is measured. If the difference in outcomes between the two groups of users is significant in favor of the treatment, the new feature is deployed at scale. While a seemingly simple idea, actually estimating the impact of a new feature can be quite complex when users are part of a network and can influence one another. While much work has been done on designing methods aiming to capture and account for interference patterns, most assume homogenous interference (Sussman and Airoldi, 2017; Toulis and Kao, 2013). However, as mentioned, this is not always a reasonable assumption. Chapter 4 presents methodology for experimental design that leverages community structure in order to improve causal inference in network settings where community level peer interference is likely. This allows for better estimation of treatment effect as well as access to community level treatment effects.

1.1 Network Notation

Throughout this work, let a network for N individuals be represented by A, a $N \times N$ binary adjacency matrix where $A_{ij} = 1$ if i and j are connected and 0 otherwise. Let $A_{ii} = 0$ for all i since we are not interested in self loops. Further, a network can be either undirected or directed. In an undirected network, A is symmetric and thus $A_{ij} = A_{ji}$ (edges are reciprocated). However, for a directed network, if $A_{ij} = 1$, A_{ji} can be 0 or 1. Also, we often are interested in how many neighbors a node has which is described as the degree of a node, $d_i = \sum_{j=1}^{N} A_{ij}$.

Consider when A is random such that there is some underlying distribution for connections in the network. To build intuition, consider a simple network model, the

Erdos-Renyi (ER) graph (Erdös and Rényi, 1959) in which

$$A_{ij} \sim Bern(p)$$

independently for all $i, j \in \{1, ...N\}$ with p being the probability of connection. Under this model, each pair of nodes has the same probability of being connected; however this assumption of uniform connection probability is not realistic. As such, we introduce a slightly more flexible version of the ER model: the Stochastic Blockmodel (SBM) (Holland et al., 1983) .

1.2 The Stochastic Blockmodel

The SBM is a popular probabilistic network model that allows for connection probabilities to vary based on latent community membership. While the notion of community structure has been discussed briefly, the SBM allows us to formally define how communities relate to network generation. For a SBM with K communities,

$$A_{ij}|U_i, U_j \sim Bern(U_i^T Q U_j)$$

where U is an $N \times K$ membership matrix such that each row contains one 1 indicating which community a node belongs to. Let Q be a $K \times K$ probability matrix describing the relations between communities. For example, if node i belongs to community 1 and node j belongs to community 2, then Q_{12} represents the probability that node i is connected to node j. For a SBM, the network is said to be assortative if the diagonal elements of Q are greater than the off diagonal elements. Assortativity is closely tied to homophily as this implies that individuals in the same community are more likely to be connected. A network is said to be dis-assortative if the diagonal

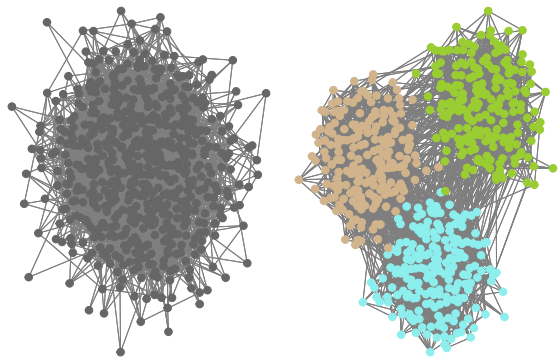

Figure 1.1: Example of a network generated from an ER model (left) versus a 3 community SBM (right).

elements of Q are less than the off diagonal, implying that individuals are more likely to be connected with those who are not in their own community. Note that if all entries of Q are equal, this reduces to the ER model. Figure 1.1 illustrates examples of an ER network (left) versus a SBM (right) where the SBM has nodes colored by true community labels. However, in practice, often these true labels are not known and only the network is available. The goal of community detection is to then recover the labels, U, from the observed network, A.

Chapter 2

Gaussian Mixture Models for Stochastic Block Models with Non-Vanishing Noise

2.1 Introduction

As mentioned, community detection tasks have received a lot of attention across statistics, machine learning, and information theory with a large body of work concentrating on theoretical guarantees of spectral based algorithms (usually implemented on the SBM). This chapter introduces the fundamentals of spectral theory that are needed to understand and motivate these types of algorithms. We then provide novel, tractable methodology for community detection in challenging regimes. This work is driven by new theoretical results for networks that leverage the asymptotic normal behavior of the scaled eigenvectors of a network.

2.2 Spectral Theory

Before discussing spectral theory applied to a network specifically, first consider a $N \times N$ real symmetric matrix, W. This matrix can be represented using its eigendecomposition: $W = V\Lambda V^T$ where $V = \{V_1, ..., V_N\}$ is the set of N orthonormal eigenvectors of W and Λ is the $N \times N$ diagonal matrix of eigenvalues such that $\Lambda_{ii} = \lambda_i$ for $i \in \{1, ..., N\}$. The set of eigenvectors, V, satisfy the following: $WV = V\Lambda$. Notice that W can be rewritten in terms of its eigenvectors and eigenvalues, and thus it is intuitive that those vectors associated with larger eigenvalues (in magnitude) carry more information about W. As a result, it is possible that certain eigenvectors can

illuminate the underlying potential 'signal' that exists in W. Since eigenvalues can be associated with how much information an eigenvector provides, it is imperative to understand their behavior. This ties to the notion of community detection where we aim to extract the community labels from the observed network.

2.2.1 Distributions of Eigenvalues

Consider redefining W as a $N \times N$ random matrix where entries are distributed identically and independently (iid). Since W is random, the eigenvalues of W are functions of random variables. However, these eigenvalues are not independent of one another and exhibit an interesting behavior as they do not get too big nor get too close to one another (Livan et al., 2018). Due to this correlation structure, standard statistical approaches are again deemed invalid (similar to the problems encountered with networks in general). However, the distribution of eigenvalues has been studied in detail for some classes of matrices. For example, let W have all off diagonal entries iid $N(0, \sigma^2/2)$ and diagonal entries iid $N(0, \sigma^2)$, yielding what is formally defined as a Gaussian Wigner matrix. In particular, it is known that for a $N \times N$ symmetric Gaussian Wigner matrix, the eigenvalues follow a semi-circle distribution (Livan et al., 2018) as illustrated in the bottom panel of Figure 2.1. While under this formulation W is essentially a noise matrix, understanding its spectral properties yields key insights for considering matrices that are a combination of noise and signal.

The Spiked Wigner Model

The spiked Wigner model formalizes this 'signal plus noise' notion and becomes useful due to the extensive study of its spectral properties in the information theory and statistical physics literature (Benaych-Georges and Nadakuditi, 2011; El Alaoui et al.,

2020; Lelarge and Miolane, 2019; Miolane, 2019; Reeves et al., 2019) as well as due to its close ties to the stochastic blockmodel (explored in the following section). Consider a spiked Wigner model of the form:

$$Y = \sqrt{\frac{\theta}{N}} F F^T + W \tag{2.1}$$

where $F \sim \mathbb{P}_0$ is a N dimensional signal vector, θ represents the strength of the signal, and W is a symmetric $N \times N$ standard Gaussian Wigner matrix. Let F encode important information that we want to recover. When only Y is observed, recovering this information can be challenging. One popular approach is to utilize the eigen-decomposition of Y. Note that for small values of θ, Y will be dominated by W and thus the eigenvalue distribution of Y will be similar to that of W. This insinuates that there is not enough signal to overcome the noise. However, as θ grows, there is a threshold at which recoverability of the signal in F is possible. This corresponds to when the largest eigenvalue(s) in magnitude of Y escape the semi-circle 'bulk' as illustrated in the top panel of Figure 2.1.

When this occurs, information related to F can be recovered through the associated leading eigenvector of Y as shown in Figure 2.2. While this particular example has a spike of rank $r = 1$, theory has been generalized for higher rank signal matrices when $r \leq N$ (Benaych-Georges and Nadakuditi, 2011) which allow us to easily connect the spiked Wigner model to the SBM for multiple communities.

2.2.2 Relation between the Spiked Wigner Model and the SBM

Recall that the goal of community detection reduces to the idea of decoupling signal from the noise in a network, identical to the goal of recovering F from Y in the

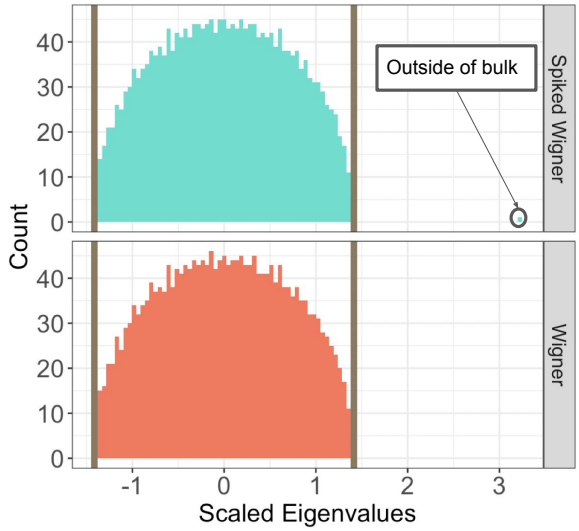

Figure 2.1: The top panel shows the distribution of eigenvalues for the spiked Wigner model where the arrow points at the 'signal' eigenvalue. The bottom panel shows the eigenvalues for just the Wigner matrix, W.

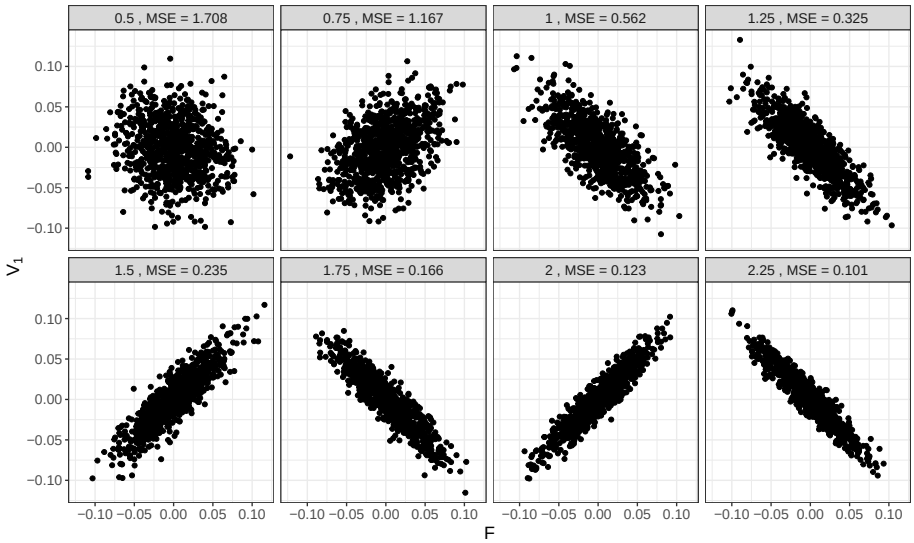

Figure 2.2: Across the panels, we adjust θ to change how challenging it is to recover F. The mean squared error between the true signal F and the leading eigenvector is also recorded in the panel labels (formally, the MSE is $\frac{1}{N}\sum_{i=1}^{N}(F_i - \hat{V}_{1i})^2$ where V_1 is the leading eigenvector of Y). As θ increases, the problem becomes easier (up to a sign change) and MSE decreases.

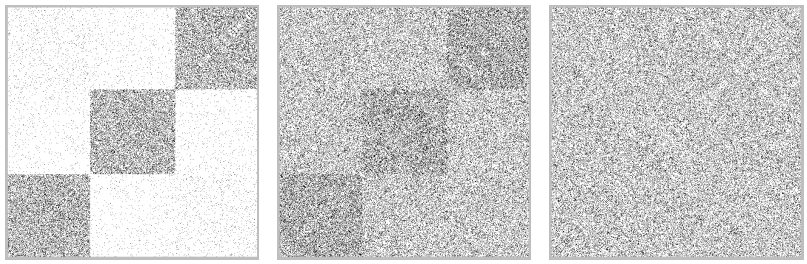

Figure 2.3: Adjacency matrix representation of 3 networks. Each matrix represents a network drawn from a 3 community SBM; each with varying connection probability matrices, Q. As we move left to right, the network becomes less assortative; thus providing less signal as those probabilities of within and between community connection become closer.

previous section. Figure 2.3 shows 3 SBMs with different connection probabilities. From left to right, the probability of within and between community edges becomes closer and thus the communities are less distinguishable, similar to the altering of θ in the previous section. Formally, the similarity between the spiked Wigner model and the SBM has been investigated (Lelarge and Miolane, 2019; Miolane, 2019) and the asymptotic equivalence between these models has been demonstrated in Reeves et al. (2019) through the use of mutual information. To illustrate these connections, recall:

$$A_{ij}|U_i, U_j \sim Bern(U_i^T Q U_j)$$

and thus $E[A|U, Q] = UQU^T$. Note that given U and Q, the entries of A are independent. If UQU^T is known and A is observed, then the residual matrix can be expressed as:

$$E = A - UQU^T$$

and thus $A = UQU^T + E$ (note the similarity to Eq (2.1)). Further, $Var[E|U, Q] = Var[A|U, Q]$. Ideally, the residual matrix would reflect the behavior of a Wigner

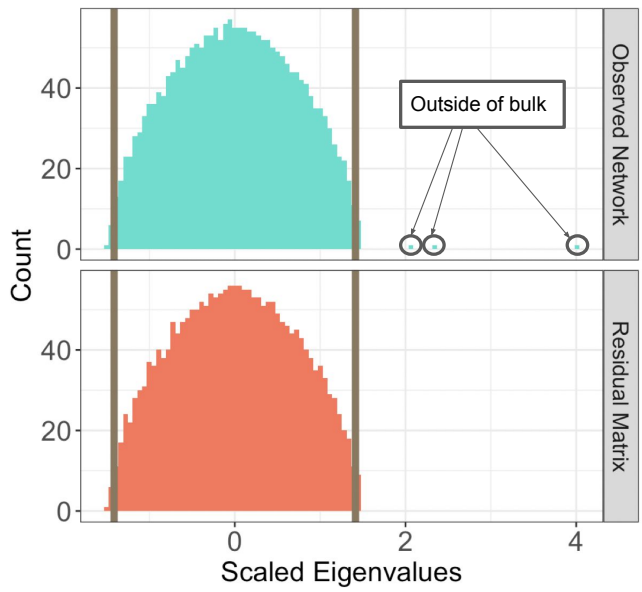

Figure 2.4: Distribution of eigenvalues for stochastic blockmodel with $K = 3$ communities for the observed scaled network, A (top panel) and for the residual matrix (bottom panel). This mirrors the behavior of the spiked Wigner model shown in Figure 2.1.

matrix (then UQU^T would reflect a spike and the models would be equivalent). Due to the underlying and non identical Bernoulli distribution of A, this exact equivalence does not quite hold. However, it is shown that this residual matrix behaves in a similar manner to a Wigner matrix, and its eigenvalues follow a semi-circle distribution (Nadakuditi and Newman, 2012). This is illustrated in Figure 2.4 where the eigenvalues of E are denoted in the bottom panel, and the eigenvalues of the scaled observed network, A, are in the top panel. Compared to Figure 2.1, the connection between the SBM and spiked Wigner model is evident. Again, when enough signal is present in the network, eigenvalues escape the semi-circle 'bulk'. Depending on how strong

the signal is, this determines the feasibility of community label recovery.

2.2.3 Recoverability of Communities

We now briefly discuss thresholds for community label recovery. When using real data, we rarely have access to ground truth communities to test the performance of an algorithm. As a result, it is crucial to understand when recoverability of labels is possible. There are different levels of recovery in community detection and corresponding notions of phase transitions for these regimes. For almost exact recovery, a vanishing fraction of labels is allowed to be misclustered where misclustered means that a node's estimated community label does not match its true community label. For partial recovery, a constant fraction of labels can be misclustered. When we are known to be in an 'impossible' regime, this implies that no algorithm performs better than random guess (where random guess can be defined as either putting all nodes in one community or uniformly randomly assigning labels to nodes). Throughout this work, random guess will be defined as the former since this reflects the true worst case scenario. While exact recovery (learn all labels correctly) is desirable, this regime is not typically realistic. As such, we focus on problems in the partial and almost exact range of recoverability (Abbe, 2017). Much work has been done for developing thresholds that relate recoverability to the eigenvalues and degree of a network to determine how much signal is needed to achieve different levels of recovery (Abbe et al., 2015; Abbe and Sandon, 2015; Decelle et al., 2011a,b; Krzakala et al., 2013; Massoulié, 2014; Mossel et al., 2018; Nadakuditi and Newman, 2012). As a simple example, consider a 2 equally sized community SBM. Decelle et al. (2011b) show that recoverability, asymptotically in N, is not possible if the difference between the expected within and between community degree ($N \times a - N \times b$) is less than $2\sqrt{c}$ where $c = N \times (a + b)/2$. An example demonstrating this behavior in a finite setting

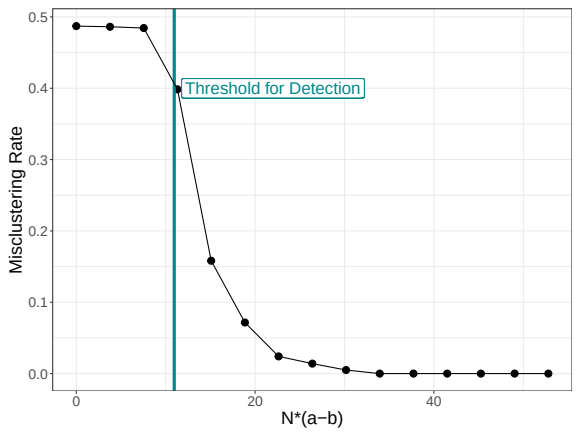

Figure 2.5: For each point, a 2 community equally sized SBM is generated with $N = 1000$. Total average degree is held constant at $d = 30$. The values of a and b vary to maintain overall average degree of 30 while also creating differences in within and between expected degree. The y axis is the miscustering rate using k-means on the top 2 eigenvectors of the observed network, A. The misclustering rate is averaged over 10 runs of the algorithm. The vertical line indicates the threshold for detection.

is given in Figure 2.5 where a clear phase transition can be seen. This figure shows the difference in expected within and between community degree versus misclustering rate (the number of misclustered nodes divided by N). As the difference in within and between community degree increases, recovery becomes feasible. To obtain the estimated labels used for calculating the misclustering rate, a k-means algorithm is implemented which is discussed in the next section.

It is also important to note that in Caltagirone et al. (2017) these thresholds are defined for a 2 community asymmetric SBM in terms of the signal to noise ratio, a function of the network eigenvalues. Further, Reeves et al. (2019) show how to extend recoverability bounds to multiple asymmetric communities when side information is available. Note that given these thresholds, the goal then becomes to develop community detection algorithms that can actually recover community labels for the network regimes corresponding to the theoretical bounds.

2.2.4 Information in Eigenvectors

Eigenvectors have been briefly discussed, and this section demonstrates how they are utilized in the community detection task. While eigenvalues carry information about the recovery of signal, they themselves cannot detect communities; however the eigenvectors can (Athreya et al., 2016; Newman, 2006a; Rohe et al., 2011). As recoverability thresholds exist in relation to the eigenvalues, a threshold also exists for the corresponding eigenvectors (Benaych-Georges and Nadakuditi, 2011; Miolane, 2019).

To build intuition as to how eigenvectors carry information, consider the leading eigenvector of a SBM. Results in spectral theory show that this eigenvector is correlated with the degree of nodes. To be explicit, the maximal eigenvalue is related to the degree of A which means that the corresponding eigenvector maximizes this quantity (Krivelevich and Sudakov, 2003):

$$\frac{\sum_{i=1}^{N} d_i}{N} \leq \lambda_{max}(A) \leq max(d_i).$$

If there is heterogeneity in degree distribution across communities, this causes community separation in the eigenvector as shown in the left plot of Figure 2.6. Note that the values of the leading eigenvector differ for the brown community as compared to the other two communities. In contrast, the right plot of Figure 2.6 shows values for the leading eigenvector of a SBM where degree is the same for each community and thus this eigenvector is uninformative.

Notice that one vector is not sufficient to completely separate all 3 communities for either SBM. When degree is not heterogenous across all communities, it becomes imperative to consider multiple vectors. While the interpretability of information in other eigenvectors is not as straightforward, it has been shown that multiple vectors

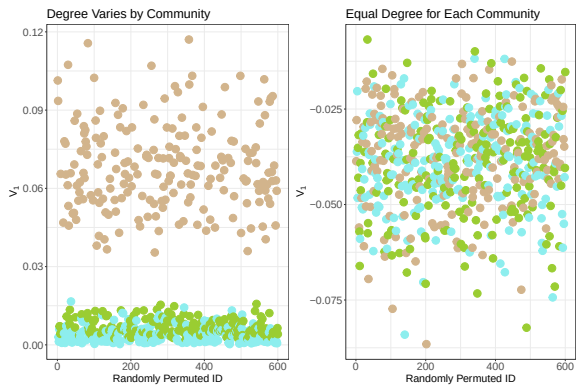

Figure 2.6: Comparison of separation on first eigenvector for SBMs with community balanced vs unbalanced degree. The x-axis is a randomized ordering of node ID and the y-axis is the V_1 value for each node. This shows how when degree is homogenous across communities, the leading eigenvector is not informative for community detection.

carry community information (Krzakala et al., 2013). Fig 2.7 illustrates this idea as multiple eigenvectors are considered. This figure plots the top 3 eigenvectors of a 3 community SBM with near equivalent expected degree across communities (which is evident when investigating separation along the leading eigenvector). Considering 3 eigenvectors shows clear separation between the 3 communities in this multi-dimensional space. This information gain from multiple eigenvectors thus motivates much of this chapter as we leverage the spectral decomposition of a network to detect communities.

The benefit of this lower dimensional embedding is that each node is now represented by a K dimensional object that presumably carries important information about each node. Further, the eigendecomposition provides an embedding that is invariant to the order of nodes in the network as demonstrated in Figure 2.8 (up to a sign change), a property crucial for implementation on real world data.

Typically K is not known a priori and must be estimated. A common approach for estimating K is to consider how many eigenvalues escape the bulk. For example, from the top plot of Figure 2.4, without any knowledge of the underlying SBM, it is clear

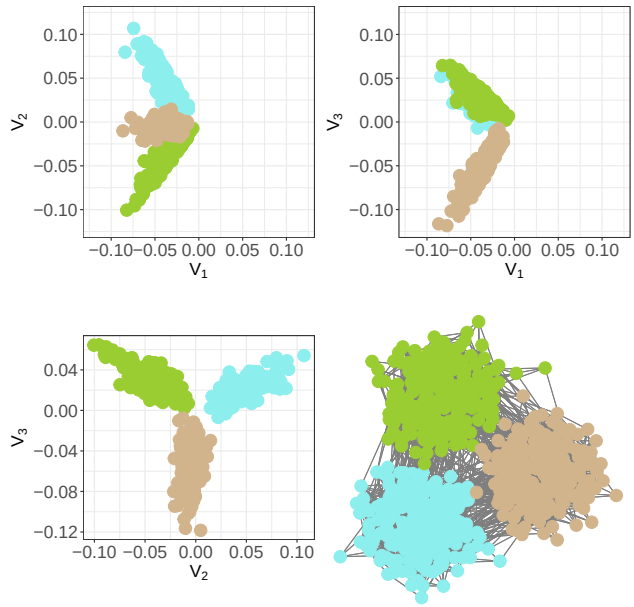

Figure 2.7: Example of a network generated from a 3 community SBM where each color indicates a different community. Plots show various eigenvectors plotted against one another to exemplify how these top 3 eigenvectors carry community structure information.

that $K = 3$ would be a good estimate since 3 eigenvalues fall outside the bulk. Given K, a standard clustering algorithm for community detection is to implement k-means on the top K eigenvectors of A or a version of A (Rohe et al., 2011). K-means is a distance based algorithm that iteratively clusters nodes into one of K communities based off of closest distance to centroids. The fundamental motivation is that nodes with similar $V_{i\cdot}$ are closer in latent space and thus more likely to share community membership. As demonstrated in Figure 2.9, overall, k-means is able to recover the majority of community labels correctly. However, the algorithm struggles to distinguish nodes where there is high overlap in the V values for some individuals from varying communities. This shows how spectral embeddings provide substantial information for community recovery, however more advanced algorithms may be able to extract more information.

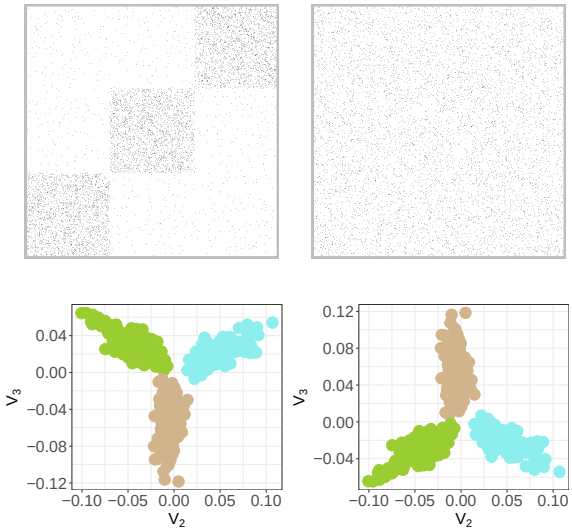

Figure 2.8: Comparison of eigendecomposition for a SBM with and without nodes ordered by community. The upper left plot shows the adjacency matrix for the SBM with nodes ordered by community (and the block structure is evident). However, the upper right plot shows the same network, however the order of the nodes has been shuffled and the block structure is not evident upon visual inspection of the adjacency matrix. However, the lower two figures show a plot of 2 eigenvectors for each version of the network respectively. This illustrates the invariance (up to a sign change) of the eigendecomposition to the order of nodes in a network.

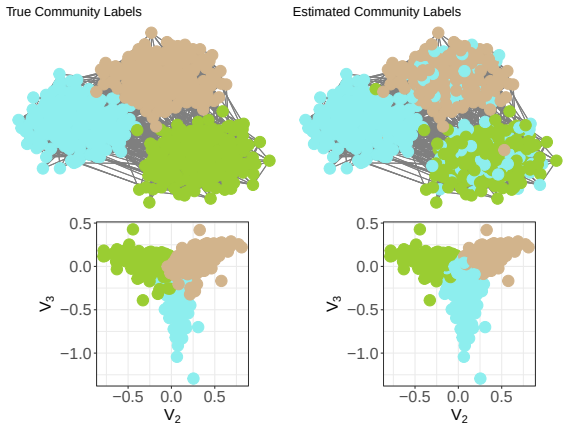

Figure 2.9: Example of k-means performance on a 3 community SBM. The left panel shows the observed network (top) and two scaled eigenvectors (bottom), colored by true community labels. The right panel shows the same network and eigenvectors, except colored by the estimated community labels from running k-means to illustrate how clustering on spectral embeddings is capable of recovering labels.

2.3 Novel Contribution

Given the vast theory supporting spectral embeddings for community detection, we consider further extensions of such methods. In current work, spectral embeddings of a network have been modelled using Gaussian mixture models (GMMs) in scaling regimes where the ability to detect community memberships improves with the size of the network. However, these settings are not very realistic as these asymptotic regimes do not exist in practice. More commonly, we encounter large, sparse networks where these theoretical results no longer hold. As shown in the sections above, these more difficult regimes are incredibly important for implementing algorithms in practice. Through the use of truncation and shrinkage effects, we present a procedure for community detection using GMMs on spectral embeddings in the non-vanishing noise regime. This method is then empirically validated using both simulated and real-world data.

2.4 Problem Formulation

As shown in Section 2.2.4, the community detection problem can be quite simple when there is heterogenous expected degree across communities since the task typically reduces to detecting the difference in degrees between communities. As such, we shift focus to the degree-balanced SBM (DB-SBM) in which each community has the same expected average degree, d. Under the DB-SBM, the community detection task is slightly more challenging. In this section, we provide a formal problem formulation that describes how all the relevant information can be captured in a $(K-1)$ dimensional embedding of the eigenvectors of the adjacency matrix. We then develop a novel GMM representation for mid- to high-noise regimes that is able to appropriately

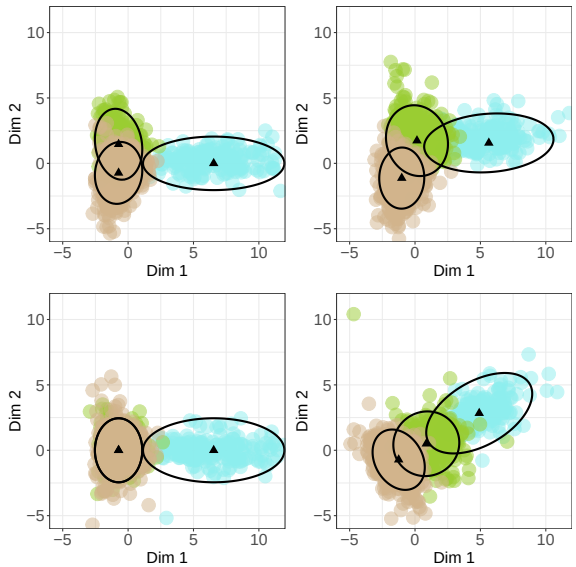

Figure 2.10: Aligned eigenvectors from graphs generated from four different SBMs. The models in the top row have two eigenvalues of R that are greater than one while the bottom row only has one eigenvalue greater than one. In the left column R is diagonal, while in the right column there is a rotation by an orthogonal matrix. Each ▲ and ellipse are based on the mean and covariance of GMM components from Section 2.5.

quantify the uncertainty about the labels of the individual nodes. In Figure 2.10, we display our proposed eigenvector embedding and the mean and covariance matrices we derive in Section 2.5 for our GMM across models with both high- (lower row of plots) and low-noise (upper row of plots) levels. Section 2.6 provides an empirical validation of our method and a comparison to other state-of-the-art algorithms.

Our approach is described in the context of a general latent space model for networks that includes the degree balanced SBM as a special case. Let $X = (X_1, \ldots, X_N)^T$ be an $N \times s$ matrix of latent variables whose rows are drawn independently from a distribution P on $\mathbb{R}^s$ with mean zero, identity covariance, and bounded support. Conditional on X, the entries of the adjacency matrix A are drawn independently

according to

$$A_{ij}|X_i, X_j \sim \text{Bern}\left(\frac{d}{N} + \frac{\sigma}{N} X_i^T R X_j\right) \tag{2.2}$$

where d is the expected degree of each node in the network, R is a symmetric $s \times s$ matrix that describes how the probability of an edge depends on the latent variables, and $\sigma = \sqrt{d(N-d)/N}$ is a scaling factor that ensures that the signal-to-noise ratio is invariant to the choice of d. The tuple (N, P, d, R) is valid only if $d/N + (\sigma/N)x^T R\tilde{x}$ is between zero and one for all $x, \tilde{x}$ in the support of P.

The assumption that P has zero mean ensures that the model is degree-balanced in expectation. Specifically, $E[A_{ij} \mid X_i] = d/N$, and thus the expected degree of node i is independent of X_i. The assumption that P has identity covariance is without loss of generality since any linear transformation of the latent variables X_i can be absorbed into the model parameter R.

When the support of P is finite, this model reduces to the degree balanced SBM where each node is assigned to exactly one of K possible communities, independently with probability vector $p = (p_1, \ldots, p_K)$. Specifically, we let X_i be supported on a set of K points $\{\mu_1, \ldots, \mu_K\}$ in dimension $s = K - 1$ satisfying the moments constraints:

$$\sum_{k=1}^{K} p_k \mu_k = 0, \qquad \sum_{k=1}^{K} p_k \mu_k \mu_k^T = I. \tag{2.3}$$

We leverage a unique specification of P given in Reeves et al. (2019, Remark 1) as a function of label probabilities p.

We note that an alternative representation for the labels, used in previous work (Rohe et al., 2011), is to associate the k-th label with the k-th standard basis vector in $\mathbb{R}^K$. An explicit mapping between these various representations is

provided in Reeves et al. (2019).

2.5 Theory and Methodology

In this section, we present a general method for inference in degree-balanced networks. This method is based on a Gaussian approximation for the spectral embedding of the adjacency matrix. Since the leading eigenvector of the adjacency matrix is correlated with the degree of the nodes and thus does not provide any information about the memberships, we consider the spectral embedding of the normalized adjacency matrix $\frac{1}{\sigma}(A-(d/N)11^T)$. The projection of this matrix onto the space of rank-s matrices can be expressed as $V\Lambda V^T$ where $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_s)$ contains the largest eigenvalues (in magnitude), in decreasing order, and $V = (V_1, \ldots, V_N)^T$ is an $N \times s$ matrix with orthonormal columns corresponding to the eigenvectors. We note that the representation of the eigenvectors is not unique.

2.5.1 Gaussian Approximation of Eigenvectors

In the context of community detection, the basic principle underlying spectral clustering is that the rows of the leading eigenvectors are correlated with the latent variables. Applying standard clustering techniques, such as k-means from Section 2.2.4, directly on the points $V_1, \ldots, V_N \in \mathbb{R}^s$ provides a partition of the nodes in the network and can be used to estimate community memberships. More generally, a principled approach to inference is to formulate a joint model for the eigenvectors and the latent parameters. This applies in the general latent space model as well as in the specialized case of community detection where the parameter space is finite. This builds upon recent work (Athreya et al., 2016; Reeves et al., 2019; Suwan et al., 2016), which provides both theoretical and empirical support for the use of GMMs.

To describe our approach, we introduce the scaled eigenvectors $Y_1, \ldots, Y_N \in \mathbb{R}^s$ according to

$$Y_i = \sqrt{N}\,\mathrm{diag}(r_1, \ldots, r_s)V_i, \tag{2.4}$$

where $r_1 \geq r_2 \geq \ldots r_s$ are the eigenvalues of R. The results in Athreya et al. (2016, Theorem 4.8) can be used to characterize the asymptotic distribution of the Y_i in the vanishing-noise regime where P is fixed and the model parameters (N, d, R) scale to infinity. Adapted to the setting of this work, this result suggests the following approximation:

$$JY_i \sim \mathcal{N}\left(RX_i, \tilde{\Sigma}(X_i)\right), \tag{2.5}$$

where J is an orthogonal matrix that aligns the eigenvectors with the latent variables and the covariance is given by

$$\tilde{\Sigma}(x) = E_{X_0 \sim P}\left[\nu(x, X_0)X_0X_0^T\right] \tag{2.6}$$

$$\nu(x, \tilde{x}) = 1 + \left(\frac{N-2d}{N\sigma}\right)x^T R\tilde{x} - \frac{1}{N}\left(x^T R\tilde{x}\right)^2. \tag{2.7}$$

The matrix J depends on the eigenspace of R as well as the particular choice of eigenvectors used in the eigen-decomposition of A. In the proposed method described below, this matrix is estimated from the data.

It is important to emphasize that the approximation in Eq (2.5) is adapted from the vanishing-noise regime where the eigenvalues of R scale with N. As a consequence, some important aspects of the moderate to high noise regimes are not captured. In particular, it is well known that an eigenvector is uninformative about latent structure unless its associated eigenvalue exceeds a threshold. For our regimes in

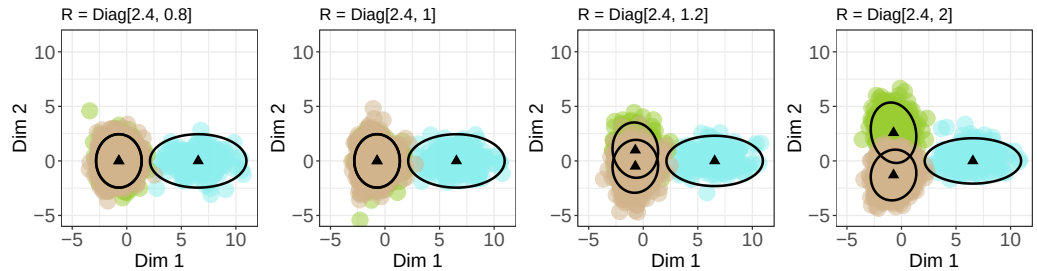

Figure 2.11: The set up for this figure is similar to Figure 2.10, however this figure emphasizes the phase transition when the magnitude of a value of R is below 1. r_1 stays constant (and high at 2.4) while r_2 varies from below to above 1 moving left to right. The blue community separates from the other 2; however distinguishing between the brown and green communities is not possible until r_2 is above 1.

particular, if any $|r_i| < 1$ where $i \in \{1, ..., s\}$, recoverability of all communities is not possible (Benaych-Georges and Nadakuditi, 2012, 2011). This is shown empirically in Figure 2.11.

We propose a Gaussian approximation for the scaled eigenvectors Y_i that incorporates both truncation and shrinkage effects via the mean and variance of Y_i. Let $\bar{R}$ and $\underline{R}$ be the symmetric $s \times s$ matrices obtained by applying the mappings $r \mapsto \max(|r|, 1)$ and $r \mapsto \min(|r|, 1)$, respectively, to the eigenvalues of R. Our approximation is given by

$$JY_i \sim \mathcal{N}\left((\bar{R}^2 - I)^{1/2} X_i, \Sigma(X_i)\right), \tag{2.8}$$

where again J is an orthogonal matrix and the covariance is

$$\Sigma(x) = (I - \bar{R}^{-2})^{1/2} \tilde{\Sigma}(x) (I - \bar{R}^{-2})^{1/2} + \bar{R}^{-1} \underline{R}^2 \bar{R}^{-1}. \tag{2.9}$$

The term $(\bar{R}^2 - I)^{1/2}$ provides the truncation and shrinkage to the eigenvalues of R.

This approximation follows from a leave-one-out argument combined with asymptotic properties of spiked Wigner matrices (Benaych-Georges and Nadakuditi, 2012). Note that any direction corresponding to an eigenvalue of magnitude less than one does not provide any information about X_i.

Proposition 1. *The matrices* $\Sigma(x)$ *and* $\tilde{\Sigma}(x)$ *satisfy*

$$\Sigma(x) = \tilde{\Sigma}(x) + O(\lambda_{\min}^{-2}(R)) \tag{2.10}$$

$$\Sigma(x) = \underline{R}^2 + O(\sigma^{-1} + N^{-1}) \tag{2.11}$$

$$\tilde{\Sigma}(x) = \underline{R}^2 + O(\sigma^{-1} + N^{-1}). \tag{2.12}$$

Proof Sketch. These results follow straightforwardly from the assumption that P has bounded support and the fact that $\nu(x, \tilde{x}) = 1 + O(\sigma^{-1} + N^{-1})$. □

Proposition 1 shows that as R increases, the approximation in Eq (2.8) converges to the vanishing-noise approximation given in Eq (2.5). Proposition 1 also shows that if either d or $N - d$ increase with N, then the covariance does not depend on X_i and is given by $\underline{R}^2$ as demonstrated in Figure 2.12. Interestingly, this result establishes a connection between the analysis of spectral methods and the information-theoretic analysis of dense networks given in Reeves et al. (2019), which also involves a GMM with common covariance across the mixtures.

2.5.2 Proposed Method

Let $\mathbb{P}_0(x, z)$ be the distribution of the pair (X_0, Z_0) where $X_0 \sim P$ and Z_0 is conditionally Gaussian:

$$Z_0 \sim \mathcal{N}\left((\bar{R}^2 - I)X_0, \Sigma(X_0)\right), \tag{2.13}$$

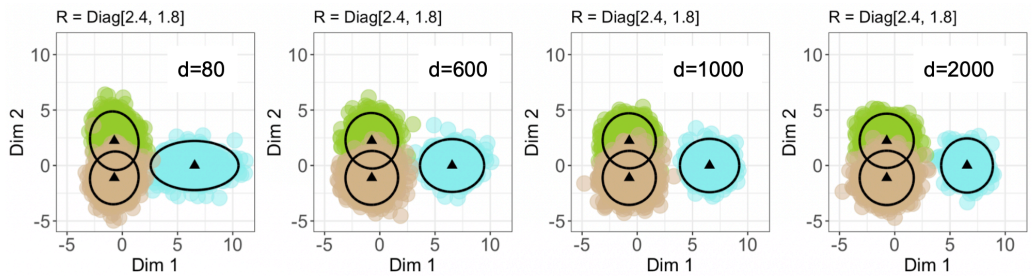

Figure 2.12: Similar to Figure 2.11, we now study how the covariance is impacted by the parameters of the SBM. This figure shows how the covariance becomes independent of X_i as d grows. Note that R is fixed and $N = 4000$.

and $\Sigma(x)$ is defined by Eq (2.9). Our method has three components:

1. (Spectral embedding) Let V be the eigenvectors of the rank-s projection of the normalized adjacency matrix $\frac{1}{\sigma}(A - (d/N)11^T)$ and let $Y_1, \ldots, Y_N$ be given by Eq (2.4).

2. (Alignment via maximum likelihood) Let J^* be a solution to the optimization problem

$$\max_J \prod_{i=1}^{n} \mathbb{P}_0(JY_i), \tag{2.14}$$

   where the maximum is over all orthogonal matrices J and $\mathbb{P}_0(z)$ is the marginal of $\mathbb{P}_0(x, z)$ with respect to x. For an SBM, this is the marginal over a GMM.

3. (Classification) For $i = 1, \ldots, N$ output the the posterior $\mathbb{P}_0(x \mid z)$ evaluated on the rotated data $z = (J^*Y_i)$. For an SBM, the posterior is represented by the probability vector $\hat{p}_i = (\hat{p}_{i1}, \ldots, \hat{p}_{iK})$.

Besides the eigenvalue decomposition, the potentially computationally challenging step in our method is the optimization with respect to an orthogonal matrix. For convenience, this optimization can be carried out over a restricted set of orthogonal

matrices belonging to the set $\{J : J\,\mathrm{diag}(r_1, \ldots, r_s)J^T = R\}$. In the simulations that follow, we obtain an approximate solution by searching over a set of representative orthogonal matrices.

2.6 Experimental Results

In this section, we study the behavior of the degree-balanced SBM, parameterized by (N, p, d, R) from Section 2.4.

2.6.1 Numerical Simulations

We generate a network of $N = 5000$ nodes and $K = 3$ communities with probability vector $p = (0.1, 0.3, 0.6)$. The support of P is defined according to (Reeves et al., 2019, Remark 1), which yields

$$\mu_1 = \begin{pmatrix} 3 \\ 0 \end{pmatrix}, \quad \mu_2 = \begin{pmatrix} -1/3 \\ 2\sqrt{5}/3 \end{pmatrix}, \quad \mu_2 = \begin{pmatrix} -1/3 \\ -\sqrt{5}/3 \end{pmatrix}.$$

The adjacency matrix is generated according to Eq (2.2) with average degree $d = 15$ and

$$R = J\,\mathrm{diag}(r_1, r_2)J^T, \qquad J = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

We compare four different methods:

- (GMM) The method described in Section 2.5.2 using the approximation in Eq (2.8).

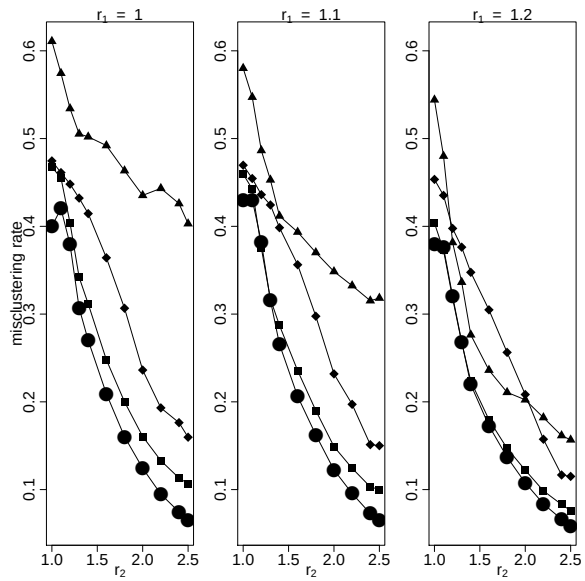

Figure 2.13: Misclustering rate across varying regimes. Each point is the average misclustering rate over 100 independent networks. Methods are given by: GMM: ●, Low-Noise GMM: ■, Uninformed GMM: ♦, k-Means: ▲.

- (Low-Noise GMM) This is the version of the method described in Section 2.5.2 where $\mathbb{P}_0(x, z)$ corresponds to the low-noise approximation in Eq (2.5).

- (Uninformed GMM) This method fits a GMM to the the rows of the eigenvectors associated with the $K - 1$ largest eigenvalues (in magnitude) of the scaled network.

- (k-Means) The k-means algorithm is applied to the same selected eigenvectors as the uninformed GMM.

We study the regimes where $(\tilde{r}_1, \tilde{r}_2) \in \{1, 1.1, 1.2\} \times \{1, \ldots, 2.6\}$ and $(r_1, r_2) = \text{sort}(\tilde{r}_1, \tilde{r}_2)$. Performance is assessed using the misclustering rate. For the GMM methods, we use the maximum a posteriori estimate of the community memberships. Following the usual convention in the literature, this metric is optimized over permutations of the estimated labels to mitigate the effects of label switching.

Figure 2.13 shows the misclustering rate across the different regimes. Each point

in the figure is the average misclustering rate over 100 networks generated for each set of parameters. As r_1 and r_2 increase, we see an improvement across all methods. When the eigenvalues of R are close to one, there is almost no correlation between the eigenvectors of the adjacency matrix and the community structure and so all methods perform poorly. Two interesting phenomena can be observed in the figure: first, the relative advantage of our proposed approach to other approaches appears to increase as the larger eigenvalue of R grows (when $r_2 = 2.5$ we have a nearly 50% improvement over the next competing method). Second, as the eigenvalues of R grow, the performance of k-means improves and surpasses that of the uninformed GMM. An additional experiment demonstrating the phase transitions that occur with respect to $|r_i|$ is available in Appendix A.2.

2.6.2 Real-World Data Analysis

In this section, we apply our method to an email network from a large European research institution (Leskovec et al., 2007; Yin et al., 2017). In these data, an undirected edge exists between person i and person j if either one or both had sent an email to the other. Only communication between individuals within the institution is considered and each person belongs to one of 42 known departments which can be treated as ground truth communities. The smallest 40 communities are combined, yielding a total of 3 ground truth communities. This leads to an approximate degree balanced network with the average in each community being around 30. We provide two types of analysis for this data: one based on oracle model parameters and one based on estimated ones. Consider that an oracle provides us with the true values of R and p (which we can compute based on our ground truth community information). The misclustering rate of our approach with oracle R and p is 0.2.

In practice, since one rarely has access to true R and p values, we also estimate $\hat{R}$ and $\hat{p}$ based on a 10% sample from the true communities. This process is described in Appendix A.1. Using the estimated values of p and R we apply the method of Section 2.5 and achieve a misclustering rate of 0.3 where we have accounted for using 10% of the data to learn the model parameters. These results compare favorably to the performance of k-means which achieves a misclustering rate of 0.368. An additional real world data implementation is given in Appendx A.3.

2.7 Discussion

In this chapter, we proposed a Gaussian mixture model representation of a projection of the adjacency matrix that can be leveraged for community detection in the non-vanishing noise regime. In contrast to prior work, we used a model for the joint distribution between the eigenvectors and the latent community structure that includes truncation and shrinkage effects. We demonstrate empirically that this novel representation is able to improve on the performance of community detection in moderate to high noise regimes.

Chapter 3

Latent Community Adaptive Network Regression

3.1 Introduction

Chapter 2 focused on the recovery of community labels when only the observed network is available. However, now consider the impact of community structure on inference when we also have nodal covariate information. This chapter considers more complex mechanisms for connection such as the interactions between nodal attributes and community structure. The inferential task in the social and health sciences often involves learning about the influence that individual and group level information can have on the formation of network ties, such as friendships among students within and across schools (e.g. the PROSPER (Moody et al., 2011) and National Longitudinal Study of Adolescent to Adult Health (Harris et al., 2009)). A common explanation for the formation of connections among individuals is the aforementioned notion of homophily. As we saw in the previous chapter, this idea has driven the development of community detection models that attempt to identify important and similar groups of individuals in the network. However, this notion also drives more general network regression models that postulate a direct link between observed individual attributes or covariates and the observed network interactions (Hoff, 2018a; Hoff et al., 2002; Holland and Leinhardt, 1981). Several approaches have brought the two tasks closer together by conjecturing that under some distributional assumptions, covariate information can assist in detecting communities in the network (Binkiewicz et al., 2017; Mele et al., 2019).

An important observation that motivates the development in this chapter is that the differences between communities within networks likely translate to differences in the impact of certain covariates for explaining edge formation. The degree corrected stochastic blockmodel was one of the first models that allows within community heterogeneity such that communities are not just detected based on degree (Qin and Rohe, 2013). Importantly, such differences are frequently observed empirically. Examples of such behavior abound in the literature. Aukett et al. (1988) shows that prediction of same sex friendships differ between men and women: women typically base relationships on sharing emotions and discussing personal issues whereas men tend to build relationships based on activities that they do together. Staber (1993) studies how men and women form entrepreneurial relationships differently, finding that women tend to form larger networks with more male ties and more strangers. Chamberlain et al. (2007) describes classmate relationship formation between children with and without autism in a mixed environment. Bail et al. (2018) shows differential effects of opposite party exposure on polarization. For all of these examples, the community labels are known, and, given this information, differences between the communities and effects of covariates are easy to evaluate.

Unlike the aforementioned studies, community labels are frequently unavailable but differences in covariate influence are still likely to persist via latent or unobserved communities. The literature lacks methods that address the important question of varying influence of covariate information based on latent community membership of individuals in a network. In this chapter, we propose a network model that addresses this gap by leveraging learned latent community information to better describe the effects of certain covariates on friendship formation.

We motivate our development by studying the high school friendship networks of the National Longitudinal Study of Adolescent Health (AddHealth), a nationally

representative study conducted during the 1994-1995 school year. In AddHealth, high school students nominate their top 5 male and top 5 female friends. If student i nominates student j, then a directed edge exists from i to j. Covariate information on students was collected including grade, smoking status, drinking habits, club involvement, GPA, and sport involvement (Harris et al., 2009). Many of these covariates have been shown to be predictors of friendship formation (Hoff et al., 2013; Kiuru et al., 2010). Given the high school setting, it is however probable that latent communities that are not associated with observed covariates are influencing friendships as well. While the communities are not predicted by observed covariates, the ties within them likely are influenced by covariates in different ways.

3.1.1 The AddHealth Data

We motivate the development of this chapter by studying friendship formation in one AddHealth school. We first analyze the data using a flexible but not community driven network model called the Additive and Multiplicative Effects Network (AMEN) model (Hoff et al., 2013). In this model covariates are linearly associated with the probability of edge formation, with positive coefficients meaning that there is a higher probability of a friendship. Covariate information for both the sender and the receiver of the ties can influence the formation of a directed edge. Figure 3.1 presents 95% posterior credible intervals for sender (labeled row) and receiver (labeled column) effects.

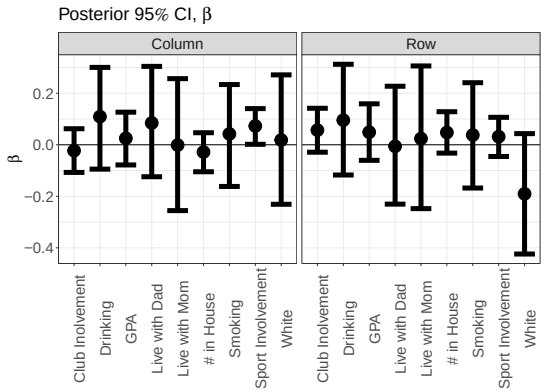

Figure 3.1: 95% CI for β estimates when fitting AMEN model on AddHealth network.

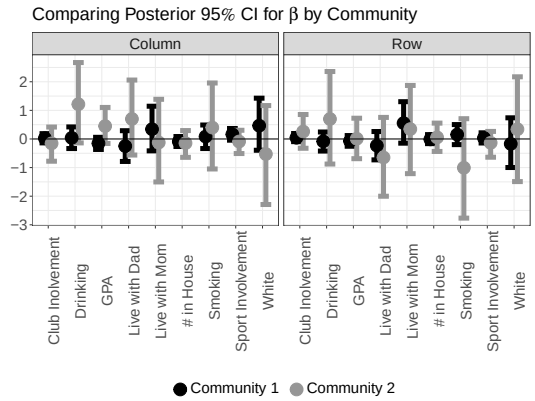

Figure 3.2: Running separate AMEN models on school network partitioned by initial estimated latent communities.

The results indicate that being involved in sports potentially increases individuals' sociability (row effect) and popularity (column effect). While the model appears to fit fairly well according to posterior predictive checks, standard community detection tools detect communities that are not directly correlated with any predictor. As such, we cannot rule out that the covariates have different effects on friendship formation within these communities. We explore this possibility by naively fitting simple AMEN models within each of the found communities. Figure 3.2 depicts the results of a naive application of community detection on the original network and then performing separate fits of AMEN within each community. We see that individuals in different estimated communities might be influenced differently by certain covariates such as being white and club involvement. However, this two-step procedure has clear downsides: partitioning the data leads to a large loss of information since only connections between people within the same estimated community are studied and the uncertainty in the community membership estimation is not propagated throughout the analysis. In the next section we propose a joint model that resolves these issues.

3.1.2 Notation and Model Setup

To fix notation, the $N \times N$ adjacency matrix, A, represents a directed network for N individuals. We further observe p covariates associated with these individuals as senders (row covariates) and as receivers (column covariates). For notational simplicity, assume that the number of row and column covariates is the same (however this is not required). Below we generalize the AMEN model that incorporates covariate coefficient estimation along with latent multiplicative and nodal random effects (Hoff et al., 2013). Specifically, we consider the generative model where $A_{ij} = 1_{Z_{ij}>0}$ (that is, the observed edge A_{ij} is a function of a latent unobserved strength of connection Z_{ij}) where:

$$Z_{ij} = \beta_0 + \sum_{l=1}^{p} (x_{rli}\beta_{rl} + x_{clj}\beta_{cl}) + U_i^T \Lambda V_j + a_i + b_j + \epsilon_{ij}. \tag{3.1}$$

In matrix form, this relational latent representation of Y can be written as:

$$Z = \beta_0 1 1^T + \sum_{l=1}^{p} (X_{rl}\beta_{rl} + X_{cl}\beta_{cl}) + U\Lambda V^T + a1^T + 1b^T + E \tag{3.2}$$

where E is an $N \times N$ matrix of possibly correlated errors. The flexibility of this model is in the specification of the random effects. Row (a) and column (b) additive effects capture sociability and popularity of nodes, respectively (this generalizes the degree corrected stochastic blockmodel) (Qin and Rohe, 2013). Since a node's popularity and sociability are likely correlated, we specify them jointly,

$$\begin{pmatrix} a_i \\ b_i \end{pmatrix} \overset{i.i.d}{\sim} N(0, \Sigma_{ab}) \text{ with } \Sigma_{ab} = \begin{bmatrix} \sigma_a^2 & \sigma_{ab} \\ \sigma_{ab} & \sigma_b^2 \end{bmatrix}$$

where σ_a^2, σ_b^2 represent the variance among row and column means and σ_{ab} represents the covariance between row and column effects. In the standard AMEN model, $U\Lambda V^T$

captures potential higher order latent dependencies between nodes. For example, if there are homophilous latent communities in A and node i and j belong to the same community, $U_i^T \Lambda V_j$ would increase Z_{ij}. In the analysis of Section 3.1.1, latent multiplicative dimension is equal to 2, that is, $U_i, V_j \in \mathbb{R}^2$. We note that the model is overparametrized when Λ is included without a restriction on the elements of U_i and V_j. As we are about to introduce such a restriction we maintain such a notation. Furthermore, this formulation showcases how the AMEN framework generalizes both the SBM and the latent distance model (Hoff, 2008). This also allows the AMEN model to capture information about triads without requiring the explicit inclusion of those statistics as in an ERGM. To capture the likely reciprocity between directed friendships, one can introduce dyadic dependence in Z by letting ϵ_{ij} and ϵ_{ji} be correlated:

$$\begin{pmatrix} \epsilon_{ij} \\ \epsilon_{ji} \end{pmatrix} \overset{i.i.d}{\sim} N(0, \Sigma_\epsilon) \text{ where } \Sigma_\epsilon = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

where ρ captures this dyadic correlation (Hoff, 2018a). The parameters of the standard AMEN model in Eq (3.1) are estimated using a Markov Chain Monte Carlo (MCMC) algorithm. In this chapter, we extend the AMEN model by considering the connection between the coefficients β and the multiplicative effects U, V.

3.2 Extending the AMEN Model

When latent communities exist, connections in different communities may be influenced by covariates in different ways. We present a model in which latent community detection is combined with network regression by allowing for community dependent coefficients. We concentrate on discrete communities, considering the previously defined multiplicative variables to be K-dimensional binary vectors where $U_{ik} = 1$ if

unit i is in community k (assume $\sum_{k=1}^{K} U_{ik} = 1$) as a sender and similarly V_{ik} for a receiver, where $\Lambda \in \mathbb{R}^{K \times K}$ represents the relationships between communities. One of the important upsides of this formulation is the interpretability of the parameters due to the discrete community memberships of each individual. We discuss a natural extension to the mixed membership and general latent space model that loses this interpretability in Chapter 5.

3.2.1 The Model

In order to allow for community dependent coefficients we re-index them as a function of U and V. That is, let

$$Z_{ij} = \beta_0 + \sum_{l=1}^{p} (x_{rli}\beta_{rlf(U_i)} + x_{clj}\beta_{clf(V_j)}) + U_i^T \Lambda V_j + a_i + b_j + \epsilon_{ij}. \tag{3.3}$$

where $f(U_i) = \sum_{k=1}^{K} k \times U_{ik}$. To facilitate computation we derive a matrix formulation for Eq (3.3). To allow each latent community to have its own covariate coefficient, let $\tilde{\beta}_r$, $\tilde{\beta}_c$ denote $p \times K$ matrices of coefficients for row and column covariates respectively. Altering Eq (3.1) to accommodate the community dependent covariate coefficients yields the following:

$$Z = \beta_0 11^T + \sum_{l=1}^{p} (\mathrm{diag}(\tilde{\beta}_{rl} U^T) X_{rl} + X_{cl}\, \mathrm{diag}(\tilde{\beta}_{cl} V^T)) + U \Lambda V^T + a1^T + 1b^T + E. \tag{3.4}$$

To see that this corresponds to Eq (3.3), consider the following three community example. Nodes 1-4 belong to communities 1, 2, 1, and 3 respectively.

$$\tilde{\beta}_{rl} U^T = \begin{bmatrix} \tilde{\beta}_{rl,1} & \tilde{\beta}_{rl,2} & \tilde{\beta}_{rl,3} \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\operatorname{diag}(\tilde{\beta}_{rl}U^T)X_r = \begin{bmatrix} \tilde{\beta}_{rl,1} & 0 & 0 & 0 \\ 0 & \tilde{\beta}_{rl,2} & 0 & 0 \\ 0 & 0 & \tilde{\beta}_{rl,1} & 0 \\ 0 & 0 & 0 & \tilde{\beta}_{rl,3} \end{bmatrix} \begin{bmatrix} x_1 & x_1 & x_1 & x_1 \\ x_2 & x_2 & x_2 & x_2 \\ x_3 & x_3 & x_3 & x_3 \\ x_4 & x_4 & x_4 & x_4 \end{bmatrix} =$$

$$\begin{bmatrix} \tilde{\beta}_{rl,1}x_1 & \tilde{\beta}_{rl,1}x_1 & \tilde{\beta}_{rl,1}x_1 & \tilde{\beta}_{rl,1}x_1 \\ \tilde{\beta}_{rl,2}x_2 & \tilde{\beta}_{rl,2}x_2 & \tilde{\beta}_{rl,2}x_2 & \tilde{\beta}_{rl,2}x_2 \\ \tilde{\beta}_{rl,1}x_3 & \tilde{\beta}_{rl,1}x_3 & \tilde{\beta}_{rl,1}x_3 & \tilde{\beta}_{rl,1}x_3 \\ \tilde{\beta}_{rl,3}x_4 & \tilde{\beta}_{rl,3}x_4 & \tilde{\beta}_{rl,3}x_4 & \tilde{\beta}_{rl,3}x_4 \end{bmatrix}.$$

To construct the above for column covariates, replace $\operatorname{diag}(\tilde{\beta}_{rl}U^T)X_r$ with $X_c \operatorname{diag}(\tilde{\beta}_{cl}V^T)$. Dyadic covariates can be represented similarly, and this discussion is relegated to Appendix B.3.

3.2.2 Relationship between $\tilde{\beta}$ in Eq (3.5) and β in Eq (3.6)

In this section we describe the way in which our model generalizes AMEN and what to expect from fitting the simpler model when the truth is community dependent. For the purpose of this example only, consider a setting where continuous interactions between units are observed, community memberships of units are known, there is no reciprocal correlation, and we are only concerned with a single potential row predictor. In this case, there are two possible model formulations:

$$\text{community dependent model:} \qquad Z = \operatorname{diag}(\tilde{\beta}U^T)X + E, \tag{3.5}$$

$$\text{independent model:} \qquad Z = X\beta + E, \tag{3.6}$$

where $\epsilon_{ij} \sim N(0,1)$ and $p = 1$. Note that this can easily be extended to a correlated error structure by using the quantities described in Section 3.3.2 as well as be extended

to include multiple row and column covariates. Let $Z^{(vec)}$ be the vectorized version of Z and $X^{(vec)}$ the vectorized version of X (Kolda and Bader, 2009). We can write the OLS estimates of β and $\tilde{\beta}$ as

$$\hat{\beta} = \frac{\sum_{i=1}^{N^2} x_i z_i}{\sum_{i=1}^{N^2} x_i^2} \ , \quad \hat{\tilde{\beta}}_k = \frac{\sum_{i \in c_k} H_{ik} z_i}{\sum_{i \in c_k} H_{ik}^2},$$

where c_k is the set of all indices for individuals in community k, $H = \{1_N \otimes [I_N \circ X]U\}$, where "$\otimes$" denotes the Kronecker product and "$\circ$" denotes the Hadamard product. The numerator of the expression for $\hat{\beta}$ can be rewritten as $\sum_{i=1}^{N^2} x_i z_i = \sum_{k=1}^{K} \sum_{i \in c_k} x_i z_i = \sum_{k=1}^{K} \sum_{i \in c_k} H_{ik} z_i$. Combined with the identity $X^{(vec)} = \sum_{k=1}^{K} H_{\cdot k}$, we can express $\hat{\beta}$ in terms of $\hat{\tilde{\beta}}$:

$$\hat{\beta} = \frac{(\sum_{k=1}^{K} \hat{\tilde{\beta}}_k (\sum_{i \in c_k} H_{ik}^2))}{\sum_{i=1}^{N^2} x_i^2} = \frac{n \sum_{k=1}^{K} S_k^2 \hat{\tilde{\beta}}_k |c_k|}{N^2 S^2} \tag{3.7}$$

where $S_k^2 = \frac{1}{|c_k| N} \sum_{i \in c_k} H_{ik}^2$ is the sample variability of X for each observed community, and $S^2 = \frac{1}{N^2} \sum_{i=1}^{N^2} x_i^2$ is the overall variability. The connection between the two models is clear — the community independent model represents a weighted average of the community dependent coefficients. As such, we can translate between the community dependent and independent models, making over-specification less problematic. We demonstrate this behavior empirically for the full model in Section 3.4.

3.3 Bayesian Computation

We rarely observe continuous interactions between individuals and rather have binary or ranked values. This makes maximum likelihood inference much more challenging, but tractable Bayesian inference is feasible. In this section, we derive a Metropolis within Gibbs sampler for the parameters of the model proposed in Eq (3.4). Our derivations generalize those presented in Hoff (2018a) by incorporating community

dependence. In the remainder of this chapter we consider the sender and receiver communities to be the same (that is $U = V$). This is a reasonable assumption for much of social science data. Generalizing to $U \neq V$ is a simple exercise that follows the same steps outlined below. The Markov Chain Monte Carlo sampler for sampling from the posterior $p(\tilde{\beta}, Z, a, b, U, \Lambda, \rho | A, X)$ is given by:

1. Update Λ: Sample $\Lambda^{(s+1)} \sim N(V_\Lambda m_\Lambda, V_\Lambda)$ where V_Λ, and m_Λ depend on $U^{(s)}$, $\tilde{\beta}^{(s)}$, $Z^{(s)}$, $a^{(s)}$, $b^{(s)}$, and $\rho^{(s)}$.

2. Update U and Z:

   (a) With uniform probability, randomly select a unit i to update. Propose new membership $U^\star_{i\cdot}$.

   (b) Compute the acceptance probability

   $$r = \min\left(1, \frac{P(U^\star_{i\cdot}|A, U^{(s)}_{-i}, a^{(s)}, b^{(s)}, \tilde{\beta}^{(s)}, \Lambda^{(s+1)}, \rho^{(s)})}{P(U_{i\cdot}|A, U^{(s)}_{-i}, a^{(s)}, b^{(s)}, \tilde{\beta}^{(s)}, \Lambda^{(s+1)}, \rho^{(s)})}\right),$$

   which does not depend on any Z.

   (c) Sample $w \sim uniform(0, 1)$ and if $w < r$ then we accept and set $U^{(s+1)}_{i\cdot} = U^\star_{i\cdot}$, otherwise $U^{(s+1)}_{i\cdot} = U^{(s)}_{i\cdot}$. If the proposal is accepted, sample a new Z directly: $\{Z^\star_{i\cdot}, Z^\star_{\cdot i}\} \sim Normal(\mu_z, \Sigma_z)$ where the parameters depend on $U^{(s+1)}$, $\Lambda^{(s+1)}$, $\rho^{(s)}$ $\tilde{\beta}^{(s)}$, $Z^{(s)}_{-i}$,$a^{(s)}$,$b^{(s)}$.

3. Update ρ: Sample $\rho^{(s+1)}$ using a Metropolis Hastings update. This is based off of the full conditional that depends on $U^{(s+1)}$, $\Lambda^{(s+1)}$, $\rho^{(s)}$, $\tilde{\beta}^{(s)}$, and $Z^{(s)}$.

4. Update $\tilde{\beta}$: Sample $\tilde{\beta}^{(s+1)}$ from $N(Vm, V)$ where V, m depend on $U^{(s+1)}$, $Z^{(s)}$, $a^{(s)}$,$b^{(s)}$, $\rho^{(s+1)}$

5. Update Z (again): Sample $T = \text{lower tri}(Z)|\text{upper tri}(Z^{(s)})$ and set $\text{lower tri}(Z^{(s+1)}) = T$ and then sample $T = \text{upper tri}(Z)|\text{lower tri}(Z^{(s+1)})$ and set $\text{upper tri}(Z^{(s+1)}) = T$.

6. Update a, b: Sample $\{a, b\}^{(s+1)}$ from a multivariate normal where the parameters depend on $U^{(s+1)}$, $\Lambda^{(s+1)}$, $\rho^{(s+1)}$, $\tilde{\beta}^{(s+1)}$, $\Sigma_{ab}^{(s)}$, and $Z^{(s+1)}$.

7. Update Σ_{ab}: Sample $\Sigma_{ab}^{(s+1)}$ from a conjugate Inverse-Wishart distribution that depends on $Z^{(s+1)}$, $a^{(s+1)}$ and $b^{(s+1)}$.

3.3.1 Priors

Normal priors are placed on β related coefficients:

$$\beta_0 \sim N(\mu_{\beta_0}, \sigma^2_{\beta_0}),\ \tilde{\beta}_{rl}, \tilde{\beta}_{cl} \sim N(\mu_{\tilde{\beta}}, \Sigma_{\tilde{\beta}}).$$

Note that $\mu_{\tilde{\beta}}$ is a K dimensional vector and $\Sigma_{\tilde{\beta}}$ is a $K \times K$ matrix. For Λ, a normal prior is also used, $vec(\Lambda) \sim N(\mu_\Lambda, \Sigma_\Lambda)$ where μ_Λ is a K^2 vector (if made symmetric, $K(K-1)/2$). For U, assume that each person has equal probability of belonging to community $k \in \{1, ..., K\}$. As such, place a uniform distribution over the canonical basis for U. The remaining priors for variance and covariance parameters are default choices (Gamma and Inverse-Wishart) as in Hoff (2018a). An arc sine prior is used for ρ and an Inverse-Wishart$(I_2, 4)$ is used for Σ_{ab}.

3.3.2 Computation

This section provides the calculations needed to sample the posterior of the parameters of interest.

Updating $\tilde{\beta}$, a, b

For ease of notation, let $p = 1$ for the derivations below. In order to derive full conditionals for $\tilde{\beta}_r, \tilde{\beta}_c$, define

$$\tilde{Z}_r = Z - \beta_0 1 1^T - X_c \operatorname{diag}(\tilde{\beta}_c V^T) - U\Lambda V^T - a1^T - 1b^T = Diag(\tilde{\beta}_r U^T) X_r + E$$
$$\tilde{Z}_c = Z - \beta_0 1 1^T - \operatorname{diag}(\tilde{\beta}_r U^T) X_r - U\Lambda V^T - a1^T - 1b^T = X_c Diag(\tilde{\beta}_c V^T) + E.$$

Throughout, the computational complication in the full conditionals comes from the correlation between Z_{ij} and Z_{ji}. To address this, we can create decorrelated versions of these as

$$\tilde{Z}_{c*} = s \times \tilde{Z}_c + t \times \tilde{Z}_c^T \quad \text{and} \quad \tilde{Z}_{r*} = s \times \tilde{Z}_r + t \times \tilde{Z}_r^T. \tag{3.8}$$

where $s = \frac{\{(1+\rho)^{-1/2} + (1-\rho)^{-1/2}\}}{2}$ and $t = \frac{\{(1+\rho)^{-1/2} - (1-\rho)^{-1/2}\}}{2}$. It is possible to then vectorize each of the objects in Eq (3.8) to perform a standard conjugate update, but this will require inverting $N^2 \times N^2$ matrices. Instead, we are able to leverage the Kronecker structure of the predictors to describe the full conditional of $\tilde{\beta}_c$ and $\tilde{\beta}_r$ in terms of the following, easy to compute objects:

$$H_c = s \times \{[I_N \circ X_c]V \otimes 1_N\} + t \times \{1_N \otimes [I_N \circ X_c]V\}, \tag{3.9}$$

$$H_r = s \times \{1_N \otimes [I_N \circ X_r]U\} + t \times \{[I_N \circ X_r]U \otimes 1_N\}. \tag{3.10}$$

The full conditionals for $\tilde{\beta}_r$ and $\tilde{\beta}_c$ are given by:

$$\tilde{\beta}_r | X_c, X_r, U, V, \Lambda, Z, \tilde{\beta}_c, a, b \sim N(V_r m_r, V_r) \text{ and}$$
$$\tilde{\beta}_c | X_c, X_r, U, V, \Lambda, Z, \tilde{\beta}_r, a, b \sim N(V_c m_c, V_c).$$

where

$$
\begin{aligned}
V_c &= (H_c^T H_c + \Sigma_{\tilde{\beta}_c}^{-1})^{-1} & V_r &= (H_r^T H_r + \Sigma_{\tilde{\beta}_r}^{-1})^{-1} \\
m_c &= vec(\tilde{Z}_{c*})^T H_c + \mu_{\tilde{\beta}_c} \Sigma_{\tilde{\beta}_c}^{-1} & m_r &= vec(\tilde{Z}_{r*})^T H_r + \mu_{\tilde{\beta}_r} \Sigma_{\tilde{\beta}_r}^{-1}.
\end{aligned}
\tag{3.11}
$$

Updating (a, b) follows similarly by computing $\tilde{Z}_a$ and $\tilde{Z}_b$, applying the transformation in Eq (3.8) and proceeding as above.

Update Λ

To update Λ we leverage the conjugacy of the normal prior again. Write the de-meaned and decorrelated data as

$$
\tilde{Z}_\Lambda = Z - \beta_0 1 1^T - \text{diag}(\tilde{\beta}_r U^T) X_r - X_c \,\text{diag}(\tilde{\beta}_c V^T) - a 1^T - 1 b^T \text{ and}
$$
$$
\tilde{Z}_{\Lambda *} = s \times \tilde{Z}_\Lambda + t \times \tilde{Z}_\Lambda^T.
$$

We can define the counterpart to Eq (3.9):

$$
H_\Lambda = s \times (V \otimes U) + t \times (U \otimes V).
$$

The full conditional of Λ is given by

$$
\Lambda | X_c, X_r, U, V, \tilde{\beta}_c, \tilde{\beta}_r, Z, a, b \sim N(V_\Lambda m_\Lambda, V_\Lambda) \tag{3.12}
$$

where $m_\Lambda = vec(\tilde{Z}_{\Lambda *})^T H_\Lambda + \mu_\Lambda \Sigma_\Lambda^{-1}$ and $V_\Lambda = (H_\Lambda^T H_\Lambda + \Sigma_\Lambda^{-1})^{-1}$.

Updating U and Z

Note that the update for different sending and receiving communities can be derived analogously. Updating large numbers of discrete parameters is notoriously associated with poor mixing of MCMC. Given the size of the parameter space, we propose

to update only a small portion of U at a time. We observe that this allows for better overall mixing. In the original AMEN model, when U is a continuous object in $\mathbb{R}^d$ the update can naturally be implemented as a Gibbs step conditional on Z (Hoff, 2018a). While updating $U|Z, \cdots$ is still possible using a Metropolis step when U takes on discrete values, the autocorrelation in the Zs induces an unacceptably high autocorrelation in the Us, thus again inhibiting mixing. We propose to sample $U|Y, \cdots$ marginializing out the Zs. This requires an evaluation of the bivariate probit likelihood. Details on the explicit form of this marginal bivariate probit likelihood are provided in Appendix B.4. In order to facilitate the remaining updates (such as those in the previous section) we must re-sample the Zs — this must now be done marginally as well since the marginalization of the Zs in the U update changes the state space, meaning we cannot condition on previous Zs.

We introduce the following update: for a randomly selected unit i we propose $U^\star_{i\cdot}$ uniformly from $\{1, ..., K\}$. If $U^\star_{i\cdot}$ is accepted then $Z_{i\cdot}$ and $Z_{\cdot i}$ are sampled from the bivariate truncated normal distribution unconditional on previous Zs. If $U^\star_{i\cdot}$ is not accepted, then the Zs can be sampled using a Gibbs step. The acceptance probability for the $U^\star_{i\cdot}$ is

$$\min\left(1, \frac{P(U^\star_{i\cdot}|A, U_{-i}, a, b, \tilde{\beta}, \Lambda, \rho)}{P(U_{i\cdot}|A, U_{-i}, a, b, \tilde{\beta}, \Lambda, \rho)}\right).$$

We note that the acceptance ratio has had Z marginalized out. We can then write

$$P(U^*_{i\cdot}|A, U_{-i}, a, b, \tilde{\beta}, \Lambda, \rho) \propto P(A|U^*_{i\cdot}, U_{-i}, a, b, \tilde{\beta}, \Lambda, \rho) \times P(U^*_{i\cdot}). \tag{3.13}$$

U_{i^*} is independent of all other rows of U. The first part requires the evaluation of the bivariate probit likelihood where pairs (A_{ij}, A_{ji}) are independent of all other pairs of observations. For the second component, recall that a uniform prior is used for U,

and the new sample is drawn from a symmetric proposal distribution. If $U_{\cdot}$ is not accepted then the state space for the Zs does not change, and they can be updated conditional on the other Zs.

Other Parameters

The updates for the other parameters, Σ_{ab} and ρ, proceed in the standard way: Σ_{ab} leverages the conjugacy of the Inverse-Wishart distribution, while ρ is updated via a Metropolis Hastings step (that is conditioned on the current value of Z).

3.3.3 Posterior Inference and Summarization

If there is no prior information about the communities and a uniform (symmetric) prior is placed on the community labels captured by U then the posterior will also be symmetric (Stephens, 2000). This is problematic since this leads to multiple posterior modes, all with equal probability, and thus potential label switching as different modes are explored. In our setup, this means that we should consider inference on $U\tilde{\beta}$ rather than $\tilde{\beta}$ directly. We thus use a post-processing procedure suggested in Stephens (2000) which has become a popular way for dealing with label switching (Fraley and Raftery, 2002; Hurn et al., 2003; Krivitsky et al., 2009). This is done by first obtaining the 95% posterior credible intervals for each individuals' $U_i\tilde{\beta}$. Then, to investigate the differences between community coefficients, clustering using k-means is performed on the posterior means of $U\tilde{\beta}$ to identify which cluster the posterior mean belongs to. Call that cluster $\hat{k}_i$ for unit i. Then the posterior credible intervals are calculated for $\tilde{\beta}_k$ by averaging the credible intervals of $U_j\tilde{\beta}$ for units j for which $\hat{k}_j = k$. This step may not be necessary if a non-uniform prior is placed over U as community labels could be identified by community proportions since the posterior would no longer be

symmetric.

3.3.4 Extending to the Censored Binary Likelihood

Many data collection procedures lead to the observed network, A, being censored by limiting the number of friends each person can nominate. Denote the maximal number of friends a person can nominate by m.

To accommodate this type of data collection, we introduce a new random effect, h_i. This parameter can be interpreted as an offset of the latent variable that measures censoring: for an individual who is not censored, this is set to 0 while for a censored individual, it is negative. This can be related to standard censored binary regression models when there is a continuous latent outcome. For example, if we set all of the $h_i = h < 0$ for censored individuals and $h_i = 0$ otherwise, this is equivalent to having a heterogeneous cutoff for censored versus non censored individuals. To provide more flexibility, we allow h_i to be different for each censored individual. Eq (3.3) is extended to:

$$Z_{ij} = \beta_0 + \sum_{l=1}^{p} (\tilde{\beta}_{rl} U_i x_{rli} + \tilde{\beta}_{cl} V_j x_{clj}) + U_i^T \Lambda V_j + a_i + b_j + h_i 1_{i \in C} + \epsilon_{ij} \tag{3.14}$$

where a truncated normal prior is placed on the vector of random effects h such that $h \sim N(0, \Sigma_h) \mathbb{1}_{h<0}$ with $\Sigma_h = \sigma_h^2 I_{N \times N}$ where σ_h^2 has an inverse-gamma prior. Let 1_C be an N dimensional indicator vector that is a 1 if an individual is censored and 0 if they are not. Let $p = 1$ and decorrelate such that:

$$\tilde{Z} = Z - \beta_0 11^T - \text{diag}(\tilde{\beta}_r U^T) X_r - X_c \,\text{diag}(\tilde{\beta}_c V^T) - a1^T - 1b^T - U \Lambda V^T \text{ with}$$
$$\tilde{Z}_* = s \times \tilde{Z} + t \times \tilde{Z}^T$$

$$H_h = s \times (1_C \otimes I_n) + t \times (I_n \otimes 1_C).$$

For censored individuals, the posterior is then:

$$h \sim N(V_h m_h, V_h^{-1}) \times \mathbb{1}\{-\infty < h < 0\} \text{ where } m_h = vec(\tilde{Z})^T H_h + \mu_h \Sigma_h^{-1} \text{ and}$$
$$V_h = H_h^T H_h + \Sigma_h^{-1}.$$

3.4 Simulations and Computational Considerations

We demonstrate the performance of our approach on synthetic data. First we compare our approach to some general alternatives including those with oracle knowledge of any underlying groups. Next we describe the initialization approach for the cluster memberships and demonstrate their importance for the mixing of the MC. We then evaluate the computational burden of the discrete communities, finding that the MH step can in fact increase efficiency in these parts of the chain. Lastly we show the performance of the proposed censored model from Section 3.3.4. Data are simulated using the model proposed in Section 3.2 with $K = 3$ equally sized communities. Let $X_r = X_c = X$. Covariates X_p are drawn from $N(0, 1)$. Coefficients, $\tilde{\beta}$, are defined in the individual simulations, and the intercept is chosen such that the observed network is approximately 30% dense. Error terms are correlated such that:

$$\Sigma_\epsilon = \begin{bmatrix} 1 & 0.9 \\ 0.9 & 1 \end{bmatrix}$$

and covariance between random effects for an individual (a_i, b_i) is:

$$\Sigma_{ab} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}.$$

3.4.1 Binary Likelihood with Community Dependent and Independent Covariates

Let $N = 150$, $\beta_r = 1$, $\beta_c = 2$, $\tilde{\beta}_r = [1, 0, -1]$, and $\tilde{\beta}_c = [0, -2, 2]$ with $U = V$. That is, the coefficients are community dependent for $X_{2,c}$ and $X_{2,r}$. Note that for both $\tilde{\beta}_r$ and $\tilde{\beta}_c$, $\sum_{k=1}^{K} \frac{\tilde{\beta}_k}{K} = 0$. Latent Z is generated as:

$$Z = \beta_0 11^T + X_{1r}\beta_r + X_{1c}\beta_c + \mathrm{diag}(\tilde{\beta}_r U^T) X_{2r} + X_{2c}\,\mathrm{diag}(\tilde{\beta}_c U^T) + a1^T + 1b^T + U\Lambda U^T + E.$$

We compare the performance of the models presented in Table 3.1. In addition to the community dependent model with varying initialization of communities, we fit standard AMEN models with varying multiplicative effects (**amen** package, Hoff et al., 2017). We also study models on partitioned data where the partition uses either the true or estimated community labels. We show that allowing the model to learn the community labels rather than using fixed estimates yields much more accurate inference. Each model runs for $150,000$ iterations.

Table 3.1: Table of different models considered in simulations

ID	joint?	Mult. Eff	Community?	Comm. coef.
a	✓	None	NA	✗
b	✓	$\mathbb{R}^3$	NA	✗
c	✓	$\{0,1\}^3$	NA	✗
d	✗	NA	Estimated before	✓
e	✗	NA	Oracle	✓
f	✓	$\{0,1\}^3$	Estimated before	✓
g	✓	$\{0,1\}^3$	Oracle	✓
h	✓	$\{0,1\}^3$	Estimated during	✓
i	✓	$\{0,1\}^3$	Estimated during	✓(all)

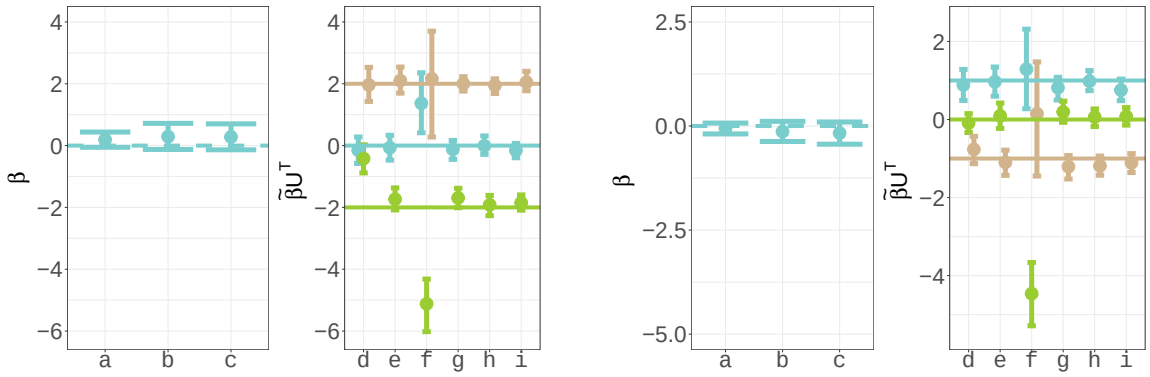

Figure 3.3: Dashed horizontal line indicates the average $\sum_{k=1}^{K} \tilde{\beta}_k / K$ for dependent covariates and solid horizontal lines are equal to the true $\tilde{\beta}$ (Figure on the left are column covariates, figure on the right are row covariates).

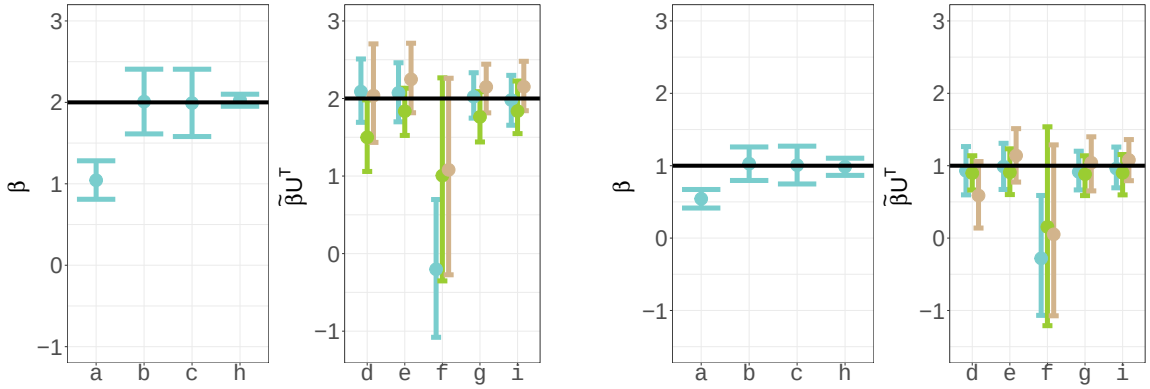

Figure 3.4: Solid horizontal lines are equal to the true β (Figure on the left are column covariates, figure on the right are row covariates).

Figures 3.3 and 3.4 plot the 95% posterior credible intervals for the coefficients for column and row covariates, respectively. The first three models can only estimate community independent coefficients while the remaining models have varying levels of adaptability. We note that model `h`, which is the full model that learns the community membership of units, has oracle knowledge of which covariates are community dependent and hence is grouped with the first three in Figure 3.4. The AMEN model with multiplicative dimension of 3 (model `b`) yields posterior credible intervals that

contain the true value of the coefficients for independent covariates (those associated with β_r and β_c). The standard AMEN models (`a`, `b`, `c`) yield posterior credible intervals that capture the average of the community dependent $\tilde{\beta}$, exhibiting the loss of information when community dependence is not considered. That is, we might incorrectly conclude that X_{2r} and X_{2c} do not appear to influence connections in A when indeed two communities are impacted by X_2 substantially and one community is not. The results for models `d` (separate models for pre-estimated communities) and `f` (full model that uses pre-estimated communities) demonstrate how using the initial estimate of community structure is not sufficient. For both of these, the posterior credible intervals fail to capture the true coefficient values. When U is known, the posterior $\tilde{\beta}$ can be approximated by the data with partitioning as in model `e`, but any cross community information is unknown (e.g. cross community dyadic effects). Nonetheless, this performs nearly as well on both community dependent and community independent covariates as the oracle model that does not partition the data (model `g`). Lastly consider two versions of our proposed model: model `h` assumes prior knowledge of which covariates are community dependent and which are not, while model `i` does not have this knowledge and fits a community dependent coefficient for each covariate. The two models yield very similar results. Importantly, model `i` yields nearly equivalent intervals across communities for X_1 while still identifying the varying coefficients associated with X_2. This shows how our approach can differentiate between covariates that are not dependent on community structure without being given that prior knowledge.

3.4.2 Initializing U

High dimensional discrete parameters are notoriously hard to work with. However, we saw in model `d` that some communities can be learned prior to model fitting. This

suggests that initializing the MCMC at a good estimate of communities might make it easier for the chain to move. Here we explore two approaches to initializing U and show how choosing one of these leads to an improvement in the autocorrelation of the $U_i\tilde{\beta}$s.

- **Spectral Methods.** We have discussed spectral methods in great detail throughout this work. Here we concentrate on a normalized graph Laplacian embedding (Rohe et al., 2011). Specifically, let $L = D_{\tau,c}^{-1/2} A D_{\tau,r}^{-1/2}$ be the normalized Laplacian where $D_{\tau,r} = D_r + \tau_r I_N$, $D_{\tau,c} = D_c + \tau_c I_N$ (these are adjusted diagonal matrices that capture the degree of each node: $D_{r,ii} = \sum_{j=1}^{n} A_{ij}$, $\tau_r = \frac{\sum_{i=1}^{N} D_{r,ii}}{N}$ and $D_{c,ii} = \sum_{i=1}^{N} A_{ij}$, $\tau_c = \frac{\sum_{i=1}^{N} D_{c,ii}}{N}$). Since networks are directed in this chapter, the singular value decomposition forms the basis for clustering (similar to the eigen-decomposition). Compute the K left singular vectors of L and cluster them using k-means clustering. Initialize U to be the matrix corresponding to those clusters. A covariate assisted version of this process is described in Appendix B.9.

- **Residual Clustering.** An alternative approach is to initialize U using residuals obtained from performing probit regression (this is in effect a feasible GLS estimate of U). The procedure is as follows: fit a simple probit regression model to the observed data,

$$P(vec(A) = 1|X) = \Phi(vec(X)\beta),$$

  and write the residuals in matrix form. Take the singular value decomposition of this residual matrix and apply clustering methods on the top K left singular vectors, similar to in the previous method.

We note that this is simply an initialization for the communities rather than a true estimate for them. In particular, both methods can actually be used in the more general setting where individuals can belong to different row and column communities (this can be done by simply clustering the left and right singular vectors in the above procedures).

Comparing Initializations

For a directed network that we study in detail in Section 3.4 we plot the true community structure (left panel of Figure 3.5), the estimated structure using a spectral approach (middle panel of Figure 3.5) and the residual clustering (right panel of Figure 3.5). In this case it appears that the spectral method yields an estimate that is slightly closer to the true community structure than the residual clustering approach. We note that in applications there is no direct access to the ground truth and so the inherent nonidentifiability of the order of individuals in a network means that such visualizations require an additional post-processing step to provide a visual indication of community structure (Chan and Airoldi, 2014). Nonetheless, visualizations such as this can help through the analysis pipeline by identifying mode changes or mixing properties of the chain.

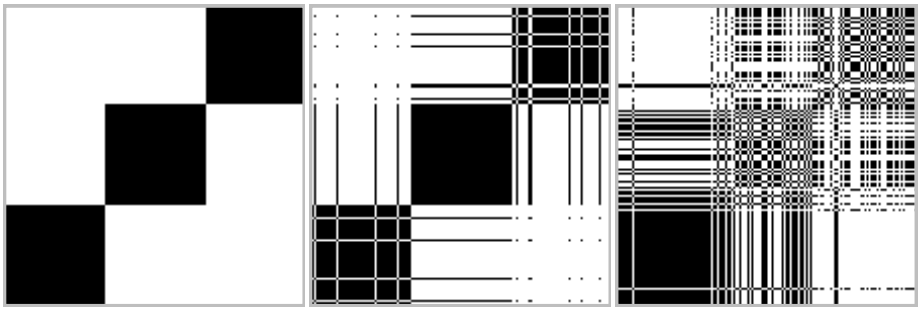

Figure 3.5: From left to right, true UU^T from the model , $\hat{U}\hat{U}^T$ where $\hat{U}$ contains membership estimates using non covariate assisted spectral clustering, $\hat{U}\hat{U}^T$ where $\hat{U}$ contains member estimates from clustering on probit regression residuals.

Comparing Convergence from Different Initialization

Here we evaluate the effect of initialization of U on the induced autocorrelation on $U\tilde{\beta}$ in the simulations from Section 3.4. Specifically we study two runs of the Markov Chain on the same data set, one initializing U uniformly at random and one with the spectral clustering proposed in Section 3.4.2. Both chains are run for $150,000$ iterations and Figure 3.6 provides summaries of the autocorrelations for the $U_i\tilde{\beta}$'s at different lags (i.e. each plot summarizes the estimates of the lag$-t$ autocorrelation across all 150 units). We see a substantial difference in the rate of decay of the ACF with the spectral decomposition exhibiting less autocorrelation and lower lags than the uniform initialization. Similar behavior is observed for other functions of the posterior.

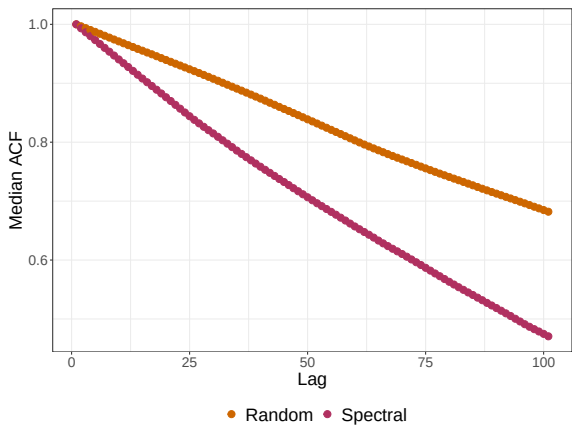

Figure 3.6: Median ACF for $U_i\tilde{\beta}$ across $i \in \{1, ..., 150\}$.

Computational Advantages

One of the important advantages of the latent space model of Hoff et al. (2002) is its flexibility in capturing different nodal behaviors (Hoff, 2008). In this section we address whether restricting the latent multiplicative positions to be discrete introduces a trade-off between computation and accuracy of estimates. We generate data from Eq (3.1), incorporating two non-community-dependent covariates and setting the latent variables to be independent samples from a 3-dimensional normal distribution $U_i \sim N_3(0, I_3)$. We fit the correctly specified AMEN model, an under specified AMEN model without multiplicative effects, and our proposed model with 3 communities and the proposed update schema. Performance is measured using effective sample size divided by a measure of accuracy for each coefficient, $\beta_{err} = (\hat{\beta} - \beta_{true})^2$ where $\hat{\beta}$ indicates the estimated posterior mean. This allows us to visualize the trade-off between speed and estimation quality. Note that higher ESS/β_{err} is desirable.

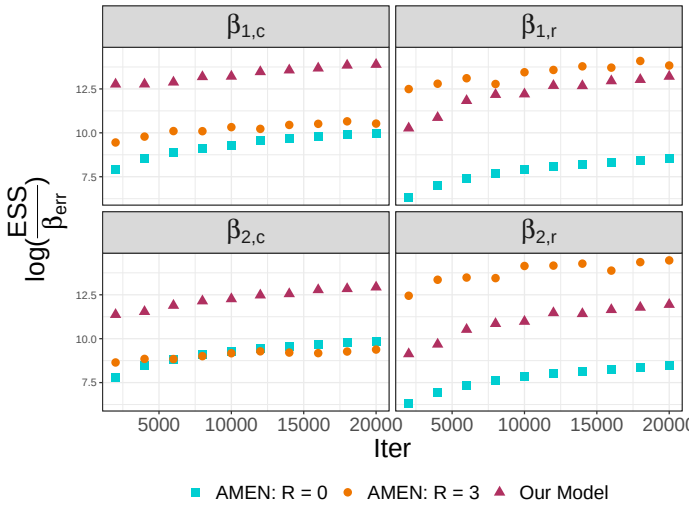

Figure 3.7: Plots showing number of iterations vs log of effective sample size (ESS) divided by β_{err}. Each point is an average over 10 runs. There are 10 different iteration levels.

Figure 3.7 plots these tradeoffs for the three models. We see that AMEN with no multiplicative effects performs the worst as expected since it is under specified and unable to account for excess variance. Our model, on the other hand, performs on par with the correctly specified AMEN model, even slightly outperforming it for two of the parameters. The move from continuous multiplicative effects to binary ones affords a computational improvement (especially with the marginal updates we propose). This figure showcases that this improvement comes with little loss in accuracy in estimating parameters even when the multiplicative effects are misspecified.

3.4.3 Effect of Censoring

Lastly, we demonstrate the performance of the proposed model when censoring of data occurs (as it does for the AddHealth data we analyze in Section 3.5). Let $N = 150$ with $K = 3$ equally sized communities. The covariates are distributed as in Section 3.4 with coefficients given by

$$\tilde{\beta}_c = \begin{bmatrix} 1 & 1 & 1 \\ 0 & -1.5 & 1.5 \end{bmatrix}, \tilde{\beta}_r = \begin{bmatrix} -1 & -1 & -1 \\ 0.5 & 0 & -0.5 \end{bmatrix}$$

where individuals are allowed to have a maximum of 15 friends (this creates a network where about about 10% of connections are censored). We fit model `i`, which includes community-based coefficients for each covariate, and Fig 3.8 plots 95% credible intervals for each of the coefficients. We see that while censoring increases the uncertainty about the estimates, the posterior is still able to capture the mean behavior, even in the presence of censoring. The similarity in inference between the specification of the censored binary likelihood in Hoff et al. (2013) and in Section 3.3.4 is demonstrated in Appendix B.5.

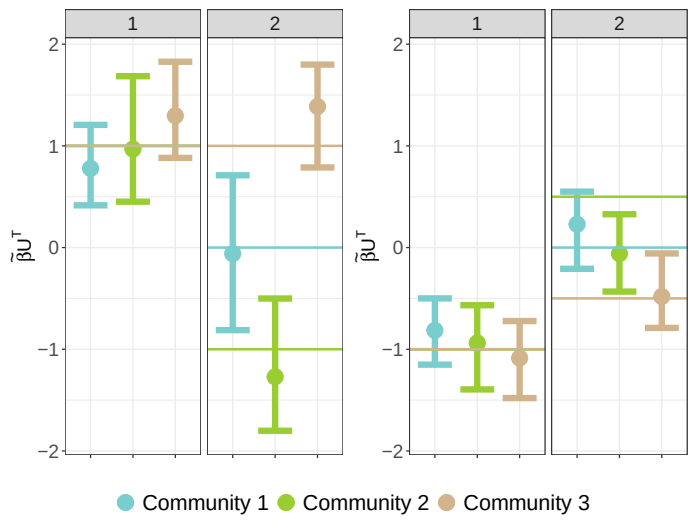

Figure 3.8: 95% Credible intervals for row and column covariates from censored community dependent model.

3.5 Application to AddHealth Data

We come back to study the friendship formation behavior of American high school students. These data were collected as part of the National Longitudinal Study

of Adolescent Health (AddHealth) dataset (Harris et al., 2009). This nationally representative study of adolescents during the 1994-1995 school year includes survey information on school and home activities and behaviors as well as information on the top five male and top five female friends for each student. Previous literature has demonstrated the importance of individual and pairwise measurements of smoking and drinking behaviors, race and extracurricular involvement on same-sex friendship formation (Hoff et al., 2013). It has further been postulated that home composition (such as living with one's mother or father) can influence in-school behavior and so we include these indicators in our analysis (Dunn et al., 2001). All covariates (described in detail in Appendix B.6) are incorporated into our analysis as potentially community informed sender, receiver and dyadic effects. Missing covariate values are imputed before the analysis using a semi-parametric Gaussian copula approach (Hoff, 2007, 2018b; Hollenbach et al., 2018).

We illustrate the flexibility of our modeling framework by studying the male-male networks of two different schools. School A is a public, rural high school in the American South with 205 male students while School B is a public, suburban school in the American Northeast with 292 males. Both schools are majority white — 80% and 92% respectively — and approximately the same fraction of each school nominated the maximal number of friends (86 and 123, respectively).

Table 3.2: Comparing percent of students with a particular response by estimated latent community

	School A		School B	
	Community 1	Community 2	Community 1	Community 2
Smoke	33.94%	23.96%	38.30%	17.88%
Drink	33.94%	25.00%	36.17%	37.75%
Smoke and Drink	22.02 %	12.5%	25.53%	15.89%
Live with Mom	90.08%	85.42%	95.74%	94.03%
Live with Dad	77.98%	79.17%	91.49%	84.77%
White	73.39%	86.46%	95.75%	89.40%

3.5.1 School A

We first present results for a school where no community structure appears to moderate the effect of covariates on friendship formation. That is, while communities are identified, community dependent coefficients are estimated to be nearly identical to community independent ones. Figure 3.9 presents the community dependent coefficients for both row- and column-effects. While some variability is present, the credible intervals for the community based coefficients are largely overlapping and the conclusions for the two communities are nearly indistinguishable. We see that those who are involved in sports are more likely to send and receive connections, while living in a larger household appears to reduce ones popularity. Dyadic covariate results are presented in Appendix B.8 and showcase that being in the same grade increases the probability of connection regardless of community membership. The only differences we observe between communities relate to the effect of race on friendship formation. While the covariate distribution matches across communities (as demonstrated in the first two columns of Table 3.2), the largest difference between them is in their racial composition. This could contribute to this slight difference in the effect of race on friendship formation.

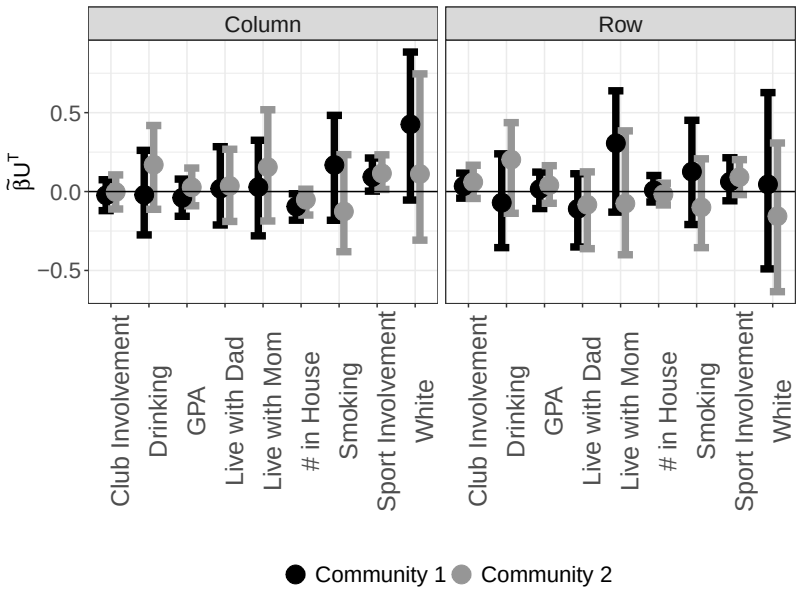

Figure 3.9: School A: 95% CI for row and column covariates with community dependent model, $K = 2$.

3.5.2 School B

The students in this school split into two communities, where each community exhibits substantively different friendship-formation behavior. The communities have 141 and 151 students respectively, and importantly (as with School A), the community structure appears latent and cannot be predicted by any combination of covariates (see last two columns of Table 3.2). Because the school is predominantly white, race is not included in the analysis (this is the only deviation from the model formulation between the two schools).

Figure 3.10 presents the 95% posterior credible intervals for the community dependent row and column coefficients (plots for dyadic covariates are in Appendix B.8). We see immediate differences between the two communities: in community 1 smoking and sport involvement increase popularity and drinking increases sociability, but these effects are not present for community 2. On the other hand, GPA appears to increase

sociability, but only in community 2. While being in the same grade increases the probability of friendship for both communities, the effect is stronger for community 1. Lastly, sharing drinking and smoking behavior increases the probability of friendship for community 1 but not for community 2.

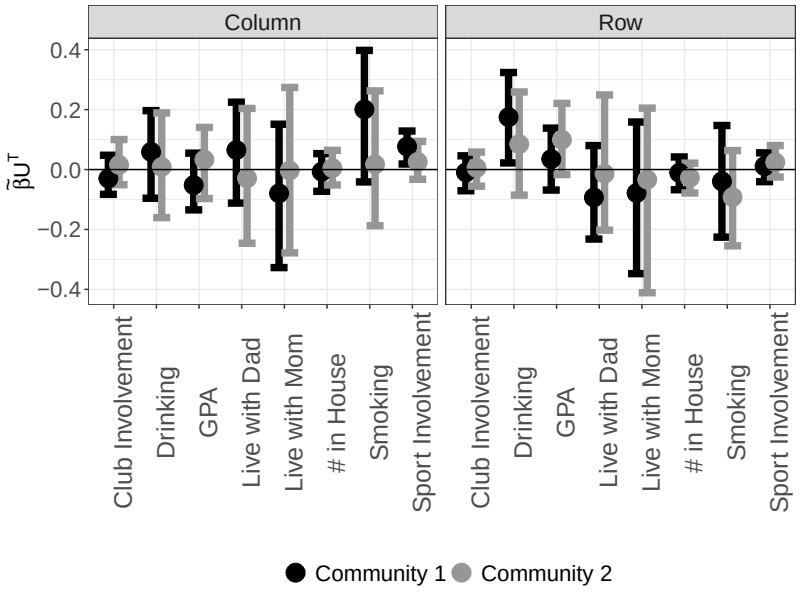

Figure 3.10: School B: 95% CI for row and column covariates with community dependent model, $K = 2$.

Comparison with community independent model Figures 3.11 and 3.12 summarize the similarity between the average community weighted coefficients and the output of the less flexible standard AMEN model with 2-dimensional multiplicative effects for Schools A and B, respectively. We see that generally the two models produce similar results when averaging over the communities. This is consistent with the theoretical claims made in Section 3.2.2 and the simulations in Section 3.4. Importantly, for networks where no community dependence is present, this suggests that estimating the more complex model does not have to come at the expense of high quality average inference.

However, when community structure is present, the average effect can be zero when the effect is non-zero for one or both communities. For example, in School A we observe a negative effect of household size on popularity but this effect is not present in either model when averaging over the communities. Similarly, smoking appears to increase sociability and drinking appears to increase popularity in the averaged results for School B (Fig. 3.12), but this effect is actually only present in one of the two communities. These discrepancies can lead to significantly different policies. For example, students from larger families might need additional support for integrating into the school community, but this would not be observable if no community dependent model were available. Similarly, while drug and alcohol programs have been demonstrated to be effective in American high schools (Eisenberg, 2003), the community dependent model suggests which communities coalesce around these behaviors, which might allow for more targeted interventions.

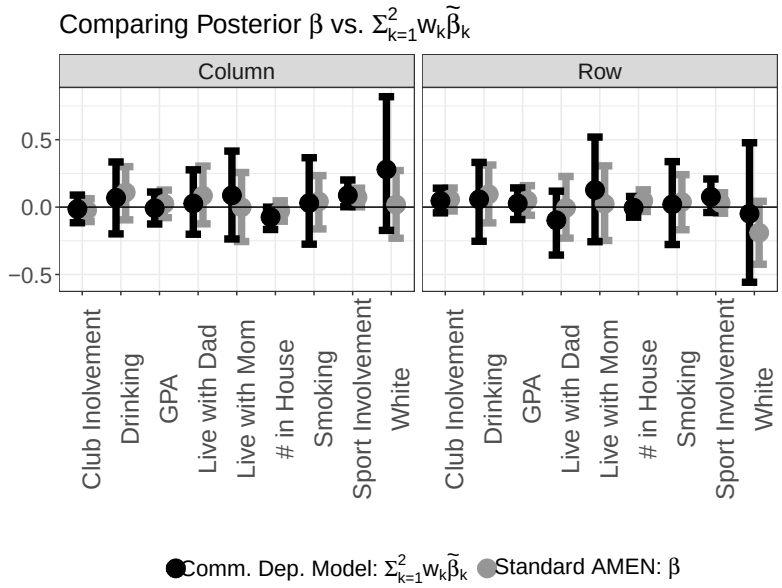

Figure 3.11: School A: Comparing 95% CI $\sum_{k=1}^{2} \frac{\tilde{\beta}_k}{2}$ vs. AMEN with multiplicative effects of dimension 2 for row and column covariates.

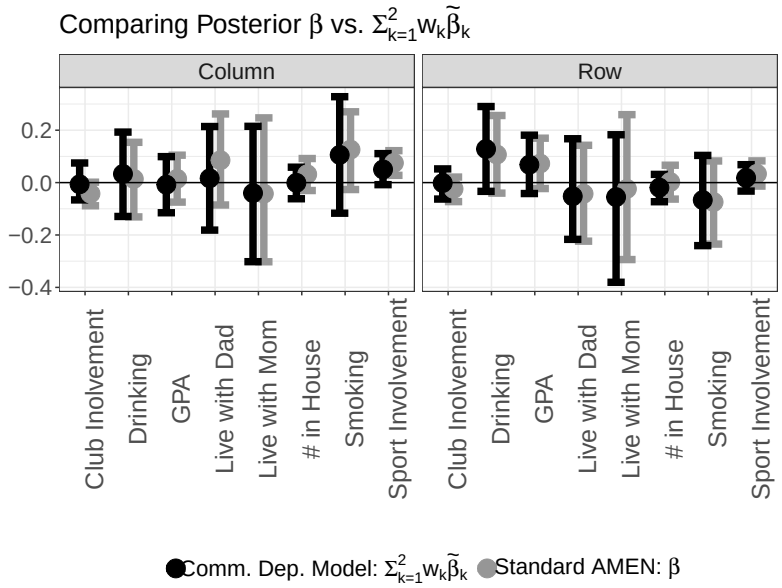

Figure 3.12: School B: Comparing 95% CI $\sum_{k=1}^{2} \frac{\tilde{\beta}_k}{2}$ vs. AMEN with multiplicative effects of dimension 2 for row and column covariates.

Convergence diagnostics As the proposed model includes a large number of parameters, convergence diagnostics must be presented to evaluate whether the Markov Chain Monte Carlo algorithm has run sufficiently long. Visual inspection of traceplots for several parameters (see plots in Appendix B.7) suggests that the MCMC is stable. Further, for each $U_j\tilde{\beta}$ we computed Geweke z-scores, with 70% and 92% of those being under 2 across all covariates and individuals for Schools A and B, respectively. We note that running the chains substantially longer would likely lead to more of the Geweke statistics to be below 2, but it is unlikely to change the estimates of the posterior. This conclusion can be evaluated heuristically: for School A there is little community dependence and substantially improved mixing in the community independent model is observed with $\sim$ %90 of Geweke z-scores under 2 (which provides estimates that are approximately the same as the community dependent model). This observation suggests that improvements to mixing can be

achieved by reducing community dependence where we do not see evidence for it.

3.6 Discussion

We have illustrated the importance of accounting for latent community structure when performing inference to determine influence of covariates on forming connections in a network. To accommodate for this, we have developed a model that efficiently incorporates community structure into the standard AMEN model by using the simultaneous estimation of memberships to allow for community dependent covariate coefficient estimation. This model provides interpretable and meaningful multiplicative effects as well as computationally efficient methods for estimating them. Additionally, the value of this model is exhibited in a real world example where accounting for community dependence allows for better understanding of how networks form in American high schools.

Chapter 4

Community Informed Experimental Design

4.1 Introduction

The previous two chapters have shown (1) that community structure is a common phenomena in networks and (2) that these communities tend to behave in different ways. We now consider how this may manifest in an experimental framework; specifically how communities may be impacted by treatment in different ways. To better understand how and why this occurs, we first present some basic terminology and concepts from causal inference and then present community informed experimental design.

Across industries, experiments provide important evaluation of new hypotheses and potential future directions (Kohavi et al., 2013; Xu et al., 2015). Whether it is testing the efficacy of a new drug or evaluating a change in the output of an algorithm, the quantity of interest must guide the experimental design. The classical experiment considers individuals in a population and a set of potential outcomes for each individual — these potential outcomes are indexed by treatments that might affect them. As such, an experiment between two alternative treatments (call them a treatment arm and a control arm) will randomly assign individuals to each arm, with the goal of quantifying a contrast between the potential outcomes of an individual had they been assigned to treatment versus had they been assigned to control. This is operationalized by first specifying several simplifying assumptions (discussed in detail below) about the causal process, specifying an estimand or quantity of interest and an estimator. A general estimand that we will consider is the global average treatment

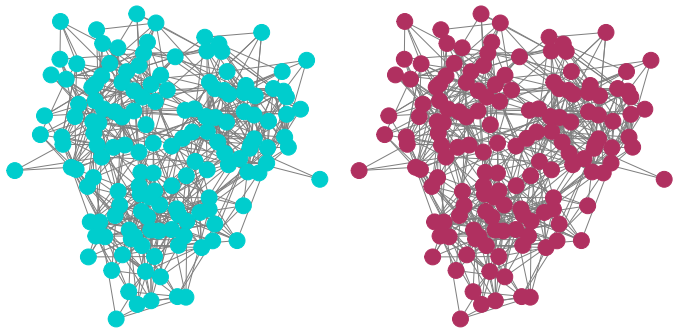

Figure 4.1: This figure shows the ideal version of the world to estimate the GATE. We would have access to two versions of the same network at the same time point: one where everyone is treated and one where no one treated.

effect (GATE):

$$\tau = \frac{1}{N}\sum_{i=1}^{N} Y_i(Z=1) - Y_i(Z=0), \tag{4.1}$$

where Z is the vector of treatment assignments for all individuals in the sample and $Y_i(Z)$ is the potential outcome for individual i under the assignment vector Z. This quantity represents the difference in outcome had everyone in the sample been treated versus not treated at the same time point as illustrated in Figure 4.1.

It is fairly clear that without additional assumptions this quantity is not estimable from data since assigning everyone to one of the arms of the experiment would not allow us to estimate anything. A further complication is that the potential outcome is indexed by the full vector of treatments — this suggests that an individual's outcome might be affected by a treatment that is assigned to someone else, a notion referred to as interference.

The GATE is of particular interest in settings where individuals are networked since interference is likely. As a result, for GATE estimation, reasonable assumptions need to be made about how interference manifests in a network. The typical assumption is that of neighborhood interference: an individual is assumed to be impacted by his or her immediate neighbors and no one else (formally defined in the next section).

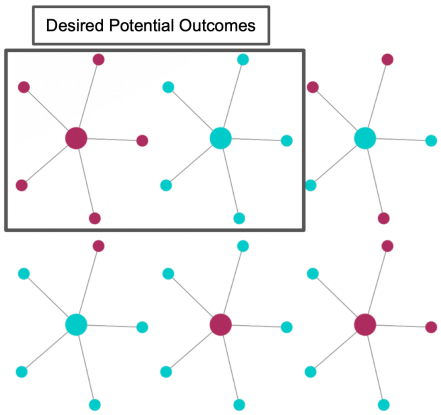

Figure 4.2: When interference occurs, there are many more possible potential outcomes in an experiment as compared to when individuals' responses are independent. For estimating the GATE, under neighborhood interference, we ideally would observe one of the outcomes where a node and all of its neighbors share the same treatment.

Examples of potential outcomes under this interference are presented in Figure 4.2.

While this assumption accounts for the basic structure of the network, it fails to account for other potentially influential features. For example, as mentioned throughout, homophily has long been demonstrated to be a driving force in many social networks (Igarashi et al., 2005; Shrum et al., 1988). As such, it is natural to believe that it is not just connections between individuals that drive interference patterns, but rather connections between similarly behaving individuals that drive interference patterns. Again, since homophilous behavior frequently leads to the formation of latent community structure in networks, a more realistic mechanism for interference is one that accounts for the fact that same-community connections are more likely to interfere with an individual's outcome than cross-community connections. We note that homophily will only affect interference patterns and not induce other cross-unit dependence in the outcomes which is known to lead to identifiability issues (Ogburn et al., 2017; Shalizi and Thomas, 2011).

Below we discuss several common simplifying assumptions and the designs they

motivate. We then formalize our own assumption that is motivated by the literature on community influence in social networks and develop an experimental design that can lead to high quality, yet easily interpretable estimates of the GATE.

4.1.1 Classical Assumptions and Designs

The most common assumption in causal inference, termed the Stable Unit Treatment Value Assumption (SUTVA) or individualistic treatment assignment, (Manski, 1995; Rubin, 1990), states that the treatment assigned to an individual can only affect their potential outcome. Under SUTVA the GATE reduces to the standard average treatment effect:

$$\tau_{ATE} = \frac{1}{N}\sum_{i=1}^{N} Y_i(Z_i = 1) - Y_i(Z_i = 0). \tag{4.2}$$

This estimand motivates a very simple design and estimation procedure: assign individuals to treatment independently, and estimate τ_{ATE} using the naive difference in means estimator,

$$\hat{\tau} = \frac{1}{N_T}\sum_{i=1}^{N} Y_i Z_i - \frac{1}{N_C}\sum_{i=1}^{N} Y_i(1 - Z_i) \tag{4.3}$$

where N_T is the number of treated individuals and N_C is the number of control individuals.

Since we know that SUTVA is likely not a viable assumption in networked populations, the literature has proposed several alternative assumptions that limit the influence of treatments across the network according to some notion of distance between individuals (Aronow and Samii, 2017; Jagadeesan et al., 2020; Sävje et al., 2017; Sussman and Airoldi, 2017; Toulis and Kao, 2013). In particular, these Network or Neighborhood SUTVA type assumptions can be written as follows: for treatment

allocation vectors Z, Z' with $g_i(Z) = g_i(Z')$ we have $Y_i(Z) = Y_i(Z')$ where the function $g_i(z)$ extracts the components of the vector Z that relate to the individuals in the network who can interfere with unit i. For example, the Neighborhood Interference assumption of Sussman and Airoldi (2017) and Awan et al. (2020) takes $g_i(Z) = (Z_i, \{Z_j : A_{ij} = 1\})$ or the simplified neighborhood exposure assumption takes $g_i(Z) = (Z_i, \sum_{\{j:A_{ij}=1\}} Z_j > 0)$. Note that this class of assumptions has been developed in order to study estimands that mimic τ_{ATE} rather than the more general GATE, and there is now a separate literature on different estimation techniques that are agnostic to the experimental design (Aronow and Samii, 2017; Sävje, 2021).

Some designs focus on only estimating the direct effect of treatment which results in the goal of minimizing the impact of interference. That is, consider how the GATE can be decomposed into a direct effect of treatment on an individual and an effect of neighborhood influence. However, in direct effect estimation, only the former is desired. When this is the goal, our type of design is sub-optimal (unless direct and interference effect can easily be decoupled which is generally not true).

To obtain high quality estimates of the GATE, network aware experimental design is crucial. A natural idea is to identify subsets of the network that are far enough apart but that represent the interference pattern of the full network. Assigning such subsets to treatment or control provides a particular view of the GATE. This approach was formalized by graph cluster randomization (GCR) (Ugander et al., 2013).

Formally, the justification for these methods rests in the notion that within a large network there are small sub-networks that are separated from each other by sufficient network distance so as to not influence the treatment effect of individuals across them. Following this, graph cluster randomization approaches partition the network into clusters using some type of clustering method (e.g. epsilon-net and one-hop max in Ugander and Yin (2020)). Letting $C(i)$ indicate the cluster that node i is assigned

to, each cluster is assigned to treatment with probability q. These methods lead to a provable reduction in bias and variance when compared to classic randomization schemes. However, these experimental designs fail to leverage additional network information.

4.1.2 Our Approach: Community Interference Assumptions and Design

The previously used network interference assumptions effectively treat every edge or connection in a network equivalently. That is, the presence of an edge implies that interference must occur. However, not all connections are created equal (Aukett et al., 1988; Bail et al., 2018; Chamberlain et al., 2007; Staber, 1993), which should be reflected in the design. Specifically, we consider the setting where network connections are formed with higher probability between similar individuals and so while connections between dissimilar individuals are possible, they are going to be deemed less strong and will not lead to interference. We will formalize this by considering networks formed according to community structure: each individual i belongs to one of K communities and edges within communities are more likely than those across communities. As in previous chapters, it is natural to represent community membership as a K dimensional binary vector U_i where $U_{ik} = 1$ if individual i belongs to community k and $\sum_{l=1}^{K} U_{il} = 1$. This framing suggests the definition of the community network interference function as $Y_i(Z) = Y_i(Z')$ if $g_i(Z) = g_i(Z')$ where

$$g_i(Z) = (Z_i, \{Z_j : A_{ij} = 1 \text{ and } U_i = U_j\}).$$

As such, interference occurs only within communities among individuals who share

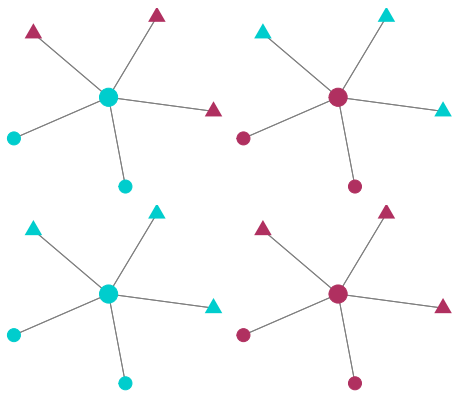

Figure 4.3: When community interference occurs, any of the shown potential outcomes are desirable. Shape of node denotes community membership and color denotes treatment. Note that in all configurations, the center node shares treatment with all same community neighbors.

connections. Examples of desirable potential outcomes are shown in Figure 4.3.

Designing a community aware mechanism for treatment assignment is crucial to capture and balance community level differences while also accurately capturing the impact of interference. In the following sections, we show how leveraging community structure, either known or estimated, improves experimental design, leading to better estimation of the GATE. As the community detection problem becomes more challenging, our methods reduce to that of standard graph cluster randomization.

Our proposed approach is as follows: identify the communities that exist in the graph and perform graph cluster randomization within these communities. There are thus two crucial steps in this design that can be notationally and conceptually confusing: (1) identifying communities that inform which units share behavior and (2) identifying design-relevant clusters which assist in randomization. Both of these steps technically use forms of clustering algorithms. However, an important distinction between these is the existence of ground truth. When we speak of community labels we are postulating that there exist true community labels that we must try to estimate — these communities inform along which connections interference will happen. As

such, the quality of estimation of these communities is important and so we discuss misclustering nodes if they are estimated to be in the wrong community. On the other hand, the clusters that are used to operationalize graph cluster randomization are purely technical artifacts and so there is no notion of ground truth or misclustering for them.

4.2 Methods

We develop a procedure that yields a high quality design for conducting experiments in networks with community structure. To illustrate this, Figure 4.4 shows examples of clusters that will be assigned to treatment or control under community informed design (left) versus standard GCR (right) on a network where nodes belong to one of two communities, denoted by the shape of the node. Recall that for networks with community interference, only within community ties are meaningful for GATE estimation. As a result, clusters of all same community nodes are desirable. Red nodes indicate that all nodes in a cluster belong to community 1 while blue indicates that all nodes in a cluster belong to community 2. Purple nodes indicate that a cluster contains nodes from both communities.

Notice that while standard GCR can produce clusters that fall into a single community, it also yields clusters that cross community boundaries. When these multi community clusters exist, individuals are less likely to share treatment with their within community neighbors. The consequences of this are explored in later sections. In contrast, community informed design guarantees that all clusters only consist of same community nodes thus capturing the true underlying interference structure. This figure also illustrates the potential of our approach even if the community structure is not informative of interference as the clusters identify potential GCR clusters.

To implement our method, community informed design can be divided into three

main parts (1) estimate community labels (2) find clusters of closely connected individuals within community level sub-graphs and (3) perform randomization at the cluster level for treatment allocation. This process is summarized in Algorithm 1 and each step is discussed in detail below.

Algorithm 1 Community Informed Design

INPUT: A, K, $f(\cdot)$, $h(\cdot)$, $s(\cdot)$
OUTPUT: Z

1: Estimate community labels, $\hat{U}$, using a community detection algorithm of choice, denoted $f(A, K)$. Specification of K a priori may not be necessary for all algorithms
2: Create a community induced sub-graph for each community based off of $\hat{U}$ that contains only within community edges
3: Run a clustering algorithm of choice, denoted $h(A, \hat{U})$, on each sub-graph independently such that each sub-graph is partitioned into clusters, $C_{\hat{U}}$. This ensures that only nodes belonging to the same community can be in the same cluster
4: Assign treatment according to some design, $s(C_{\hat{U}})$. This could denote a pairing design, randomization designs with covariate information, etc

Estimating Community Labels First consider the algorithm choice for community label recovery, $f(A, K)$. An abundance of algorithms exist in the literature for this task (Abbe, 2017; Bhattacharyya and Bickel, 2014; Blondel et al., 2008; Bruna and Li, 2017; Mathews et al., 2019; Rohe et al., 2011), any of which could be used here. However given the recent theoretical and empirical results in information theory and statistics, we implement spectral methods (Krzakala et al., 2013; Mayya and Reeves, 2019; Reeves et al., 2019). At a high level, spectral based methods produce an embedding of the observed network in a lower dimensional space using an eigen-decomposition of the adjacency matrix. There are several (nearly equivalent) approaches for doing this. We implement one from Rohe et al. (2011) that considers the normalized graph Laplacian, a summary of the network that better captures

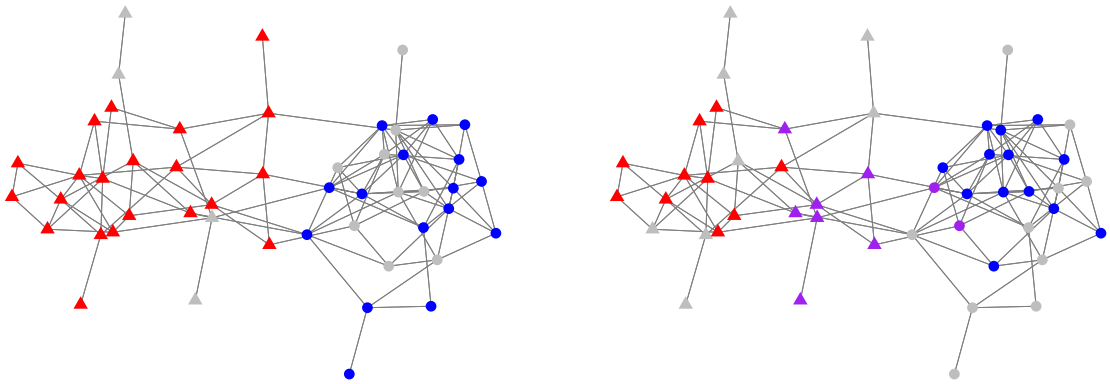

Figure 4.4: This figure shows two versions of the same network where each has a clustering design implemented. The left plot shows example cluster assignments under community informed design while the right shows standard graph cluster randomization. This network is generated from a 2 community stochastic blockmodel. The true community labels are indicated by the node shape (triangles for community 1, circles for community 2). Red nodes indicate clusters where all nodes in a cluster are in community 1 whereas blue nodes indicate all nodes in a cluster are in community 2. Purple nodes indicate clusters that have nodes from both community 1 and 2. Not all clusters are colored; hence why some nodes are gray.

clustering behavior, especially for sparse networks. This is defined as

$$L = D^{-1/2} A D^{-1/2}$$

where D is a diagonal matrix of degrees in adjacency matrix A, with $D_{ii} = \sum_j A_{ij}$ (note this is similar to the method used in Chapter 3). Using a K dimensional eigen-decomposition, we write that $L \approx V \Lambda V^T$ where again K represents the number of communities one expects to observe in the network (this value can be adaptively chosen by considering the size of the non-zero eigenvalues of L, otherwise known as identifying the eigenvalues that fall far from the 'bulk' as discussed in Chapter 2 (Krzakala et al., 2013)). After the embedding, the k-means algorithm with multiple restarts is applied to the top K eigenvectors, V. The resulting labels are used to create $\hat{U}$.

Determining Clusters After the labels are estimated, a community induced sub-graph is created for each of the K communities. As a result, each sub-graph contains only within community ties. For determining clusters for treatment, $h(A, \hat{U})$ is chosen to be the 3 net (generically epsilon net) algorithm which is implemented on each sub-graph separately. Unlike the previous step where the goal is to find ground truth communities that define interference, the goal of this step is to find sets of closely connected individuals where each cluster is far from the other clusters. At a high level, 3 net clustering generates a random ordering of all nodes in the graph. Based on this ordering, a maximal distance 3 hop independent set is created, and the nodes in this set are called seed nodes. The number of seed nodes determines the number of clusters that will exist in the experiment. Each node is assigned to its closest seed node based off of minimal graph distance, and these assignments then form clusters. Since the seed nodes are guaranteed to be network exposed to either control or treatment, this algorithm makes estimation of exposure probabilities plausible through Monte Carlo simulation. Ugander et al. (2013) demonstrate the theoretical and empirical properties of 3-net GCR which we adapt to our community informed design. An explicit outline of the implementation of the community epsilon net is in Appendix C.4.

Randomization of Clusters Finally, for this work, $s(C_{\hat{U}})$ is defined to be independent cluster level randomization where each cluster is treated with probability $q = 0.5$. However, this randomization scheme can be altered depending on the desired estimand or if the design has constraints on randomization schema.

Note that community informed design generalizes other discussed methods. If community labels are known, Algorithm 1 can be adjusted such that $\hat{U} = U$ and thus step 1 can be skipped. Also, when $K = 1$, Algorithm 1 reduces to standard graph cluster randomization, and further, if $K = 1$ and the number of clusters is set to N

then Algorithm 1 becomes standard independent randomization.

4.3 Bias Reduction

We study the bias of using the naive difference in means estimator of Eq (4.3) for estimating the GATE under community driven interference. In this setting, specifically let $g_i = \sum_{j \in N_i^c} Z_j$ (where N_i^c is the neighborhood of within community neighbors of i) and write the potential outcome as $Y_i(Z_i, g_i)$. The potential outcomes can be decomposed as $Y_i(Z_i, g_i) = C_i(Z_i) + B_i(g_i)$ (similar to the decomposition in Aronow and Samii (2017); Jagadeesan et al. (2020); Karwa and Airoldi (2018)) . The true GATE, τ, can be represented as:

$$\frac{1}{N}\sum_{i=1}^{N}(C_i(1) + B_i(d_i^c)) - (C_i(0) + B_i(0)) \tag{4.4}$$

where d_i^c is the number of within community neighbors of node i.

The derivation for the expectation of the estimator, $\hat{\tau}$, then follows as:

$$\begin{aligned}
E_Z[\hat{\tau}] &= E_Z\left[\frac{1}{N_T}\sum_{i=1}^{N} Y_i Z_i - \frac{1}{N_C}\sum_{i=1}^{N} Y_i(1 - Z_i)\right] \\
&= E_Z\left[\frac{1}{N_T}\sum_{i=1}^{N} Y(1, g_i) Z_i - \frac{1}{N_C}\sum_{i=1}^{N} Y(0, g_i)(1 - Z_i)\right] \\
&= E_Z\left[\frac{1}{N_T}\sum_{i=1}^{N} (C_i(1) + B_i(g_i)) Z_i - \frac{1}{N_C}\sum_{i=1}^{N} (C_i(0) + B_i(g_i))(1 - Z_i)\right].
\end{aligned}$$

Combining the above, where N_T, N_C are fixed, the bias is given by:

$$
\begin{aligned}
b &= E_Z[\hat{\tau}] - E[\tau] \\
&= \sum_{i=1}^{N} \Bigg(C_i(1)\pi_i(1) + \pi_i(1, d_i^c) B_i(d_i^c) + \pi_i(1,0) B_i(0) + \sum_{g_i \neq \{0, d_i^c\}} \pi_i(1, g_i) B(g_i) \Bigg) \\
&- \sum_{i=1}^{N} \Bigg(C_i(0)\pi_i(0) + \pi_i(0, d_i^c) B_i(d_i^c) + \pi_i(0,0) B_i(0) + \sum_{g_i \neq \{0, d_i^c\}} \pi_i(0, g_i) B(g_i) \Bigg) \\
&- \Bigg(\frac{1}{N} \sum_{i=1}^{N} (C_i(1) + B_i(d_i^c)) - (C_i(0) + B_i(0)) \Bigg)
\end{aligned}
$$

which can then be written as:

$$
b = \sum_{i=1}^{N} C_i(1) \Bigg(\pi_i(1) - \frac{1}{N} \Bigg) - C_i(0) \Bigg(\pi_i(0) - \frac{1}{N} \Bigg) \tag{4.5a}
$$

$$
+ \sum_{i=1}^{N} B_i(d_i^c) \Bigg(\pi_i(1, d_i^c) - \pi_i(0, d_i^c) - \frac{1}{N} \Bigg) \tag{4.5b}
$$

$$
+ \sum_{i=1}^{N} \sum_{g_i \neq \{0, d_i^c\}} B_i(g_i) \Bigg(\pi_i(1, g_i) - \pi_i(0, g_i) \Bigg). \tag{4.5c}
$$

This calculation mirrors the results of Karwa and Airoldi (2018) for general interference. Above, $g_i \neq \{0, d_i^c\}$ is equivalent to $g_i \in \{1, ..., (d_i^c - 1)\}$ and thus indicates the set of all possible observed g_i under our interference assumption except for those where either no within community neighbors are treated or those where all within community neighbors are treated. Anytime the assignment is different from $(1, d_i^c)$ or $(0, 0)$, estimation of the counterfactuals that contributes to the GATE suffer.

As a result, a design is needed where the probability of observing these is small. Below we show that CID leads to smaller bias than GCR. We consider $N_T = N_C = N/2$

and define the following weights as:

$$\pi_i(1) = \frac{1}{N_T} E[\mathbb{1}[Z_i = 1]] = \frac{1}{N_T} P(Z_i = 1)$$
$$\pi_i(0) = \frac{1}{N_C} E[\mathbb{1}[Z_i = 0]] = \frac{1}{N_C} P(Z_i = 0)$$

and

$$\pi_i(1, g_i) = \frac{1}{N_T} E[\mathbb{1}[Z_i = 1, G_i = g_i]] = \frac{1}{N_T} P(Z_i = 1, G_i = g_i)$$
$$\pi_i(0, g_i) = \frac{1}{N_C} E[\mathbb{1}[Z_i = 0, G_i = g_i]] = \frac{1}{N_C} P(Z_i = 0, G_i = g_i).$$

The values of the above weights are determined by the chosen design and accompanying interference assumptions.

There are three components to the bias. The first part, Eq 4.5a, can be controlled by choosing a design that treats nodes with probability $q = 0.5$. The second component, Eq 4.5b, can also be controlled by the design: under community informed design, $\pi_i^{CID}(0, d_i^c) = 0$ since each individual belongs to a cluster within their own community and must share treatment status with at least one of their within community neighbors. We can further control $\pi_i^{CID}(1, d_i^c)$ which is given in Eq (4.6). In contrast, under GCR, we will show that $\pi_i^{GCR}(0, d_i^c) \geq 0$ and $\pi_i^{GCR}(1, d_i^c) \leq \pi_i^{CID}(1, d_i^c)$ which leads to:

$$\left| \sum_{i=1}^{N} B_i(d_i^c) \left(\pi_i^{CID}(1, d_i^c) - \pi_i^{CID}(0, d_i^c) - \frac{1}{N} \right) \right| \leq$$
$$\left| \sum_{i=1}^{N} B_i(d_i^c) \left(\pi_i^{GCR}(1, d_i^c) - \pi_i^{GCR}(0, d_i^c)) - \frac{1}{N} \right) \right|,$$

suggesting that CID reduces the contribution of Eq 4.5b.

To demonstrate the above, we consider GCR and CID with the 3-net clustering

algorithm. Let d^c_{max} denote the maximal within community degree across all nodes in A. Following from Ugander and Yin (2020), the network exposure probabilities can be bounded for both designs. Tighter bounds are derived in Appendix C.2, however for simplicity, consider the worst case scenario. For community informed design, $P(Z_i = 1, G_i = d^c_i) \geq \frac{.5}{d^c_{max} \times (1+d^c_{max})}$. To obtain $\pi^{CID}_i(1, d^c_i)$, we simply divide $P(Z_i = 1, G_i = d^c_i)$ by $E[\sum_i^N \mathbb{1}[Z_i = 1]] = N/2$:

$$\frac{1}{N d^c_{max} \times (1 + d^c_{max})} \leq \pi^{CID}_i(1, d^c_i) \leq \frac{1}{2N} \leq \frac{1}{N}, \tag{4.6}$$

assuming that a node has at least 1 within community neighbor which is reasonable under assortative community structure.

Again, under CID when true communities are known, $\pi^{CID}_i(0, d^c_i) = 0$ and $\pi^{CID}_i(1, d^c_i) < 1/N$. However, we are not guaranteed that $\pi^{GCR}_i(0, d^c_i) = 0$. Further, under GCR, $P(Z_i = 1, G_i = d^c_i) \geq \frac{.5}{d_{max} \times (1+d_{max})}$. Since $d_{max} \geq d^c_{max}$, these exposure probabilities under GCR are lower than that under our design. As a result, our design yields a probability closer to $1/N$; therefore CID yields lower bias in the second term. Also note, that under CID, given the higher values of $P(Z_i = 1, G_i = d^c_i)$ and $P(Z_i = 0, G_i = 0)$, the probability of ending up with less desirable exposure is sufficiently reduced.

The last component of the bias calculation, Eq 4.5c, contributes a fairly small amount to the total bias. Note that $\pi_i(1, g_i)$ should be large when $g_i \approx d^c_i$ and small otherwise. Similarly $\pi_i(0, g_i)$ should be large when $g_i \approx 0$ and small otherwise. We show this empirically in Figure 4.5. If $B_i(g_i)$ is relatively flat in g_i, this will lead to terms canceling in Eq 4.5c. Alternatively, if $B_i(g_i)$ is increasing in g_i, only the terms associated with large g_i will contribute to the bias. While the exact ordering between GCR and CID depends on the $B_i(g_i)$, due to the $P(1, d^c_i)$ being larger for CID, we

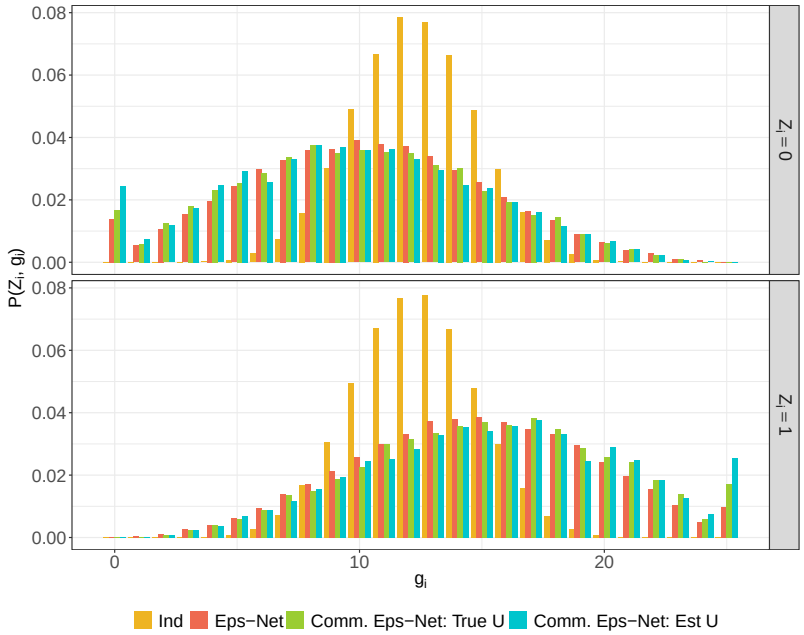

Figure 4.5: Simulated $P(Z_i, g_i)$ over 4000 simulations. This figure demonstrates how community informed design puts higher probability on more desirable potential outcomes.

expect the contribution to the bias to be smaller under CID.

4.4 Cost of Misclustered Nodes

Thus far we have abstractly discussed the existence and identification of communities. Since these communities form the foundation of CID, it is crucial to understand how community label estimation impacts the design and GATE estimation. In this section, we derive the expected bias incurred for GATE estimation that is due to incorrectly estimating the community label of individuals in the first stage of the algorithm. Define a node, i, to be misclustered when $\hat{U}_i \neq U_i$ up to permutation of the labels (estimated community label does not match the true community label). In order to explicitly derive the cost of misclustered nodes we take a model assisted approach

(Basse and Airoldi, 2018; Särndal et al., 2003). Consider the following outcome model:

$$Y_i = \alpha + \beta Z_i + [(U\Gamma U^T)_{i,} \circ A_{i,}]Z^T + \epsilon_i \tag{4.7}$$

$$= \alpha + \beta Z_i + \sum_{j=1}^{N} U_{i,}\Gamma U_{j,}^T \times A_{ij} Z_j + \epsilon_i, \tag{4.8}$$

where α is a baseline effect, β is overall direct effect, Z is the treatment indicator vector, Γ is a $K \times K$ matrix describing the effect of treated neighbors based on community membership and ϵ represents individual variability. Further, A is the adjacency matrix of the network, generated from a SBM where Q is parameterized by 2 probabilities, a and b. As a result, $A_{ij}|U_i, U_j \sim Bern(a)$ if $U_i = U_j$ or $A_{ij}|U_i, U_j \sim Bern(b)$ if $U_i \neq U_j$. We evaluate how well the GATE is estimated in a setting where $U_i = U_j$ but $\hat{U}_i \neq \hat{U}_j$.

Studying the effect of misclustered nodes can be reduced to asking how close the observed Y_i is to the actual counterfactual of interest. For example, when individual i is treated, we would want his or her true within community connections to also be treated, thus consider:

$$E\Big[Y(Z=1)_i - Y_i | Z_i = 1\Big] =$$
$$E\Big[\gamma U_i \sum_j A_{ij} \times \mathbb{1}[U_i = U_j] - \gamma U_i \sum_j A_{ij} \times \mathbb{1}[U_i = U_j] \times Z_j | \hat{U}\Big]$$

where the expectation is with respect to the model uncertainty and the design but conditioned on $\hat{U}$. Now consider the situation when the label of node i is incorrectly estimated. Formally, let T_{ij} be the event that $U_i = U_j$ and $\hat{U}_i \neq \hat{U}_j$. As such, the difference between the counterfactual and observed value for a misclassified treated

node is equal to

$$E\left[\sum_{j=1}^{N} A_{ij} Z_j \gamma U_i \mathbb{1}_{T_{ij}}\right].$$

Note that conditional on T_{ij}, A_{ij} is independent of U_i and Z_i is independent of Z_j. Hence, we can write:

$$E\left[\sum_{j=1}^{N} A_{ij} Z_j \gamma U_i \mathbb{1}_{T_{ij}}\right] = \mathbb{P}(T_{ij}) \times N \times a \times q \times \sum_{k=1}^{K} \gamma_k P_k \tag{4.9}$$

where P_k is the probability that a node belongs to community k. Recall $P[Z_j|T_{ij}] = q$ and $P[A_{ij}|T_{ij}] = a$. The probability of T_{ij} depends heavily on the network structure and algorithm f (for label recovery). However, this probability can easily be represented in terms of the sizes of each community and the number of misclustered nodes within each community. Let D_k be the number of misclustered nodes in community k and N_k be the number of nodes in true community k. For $K = 2$:

$$P(T_{ij}) = \frac{E[\sum_{k=1}^{K}(N_k - D_k) \times 2D_k]}{N^2}.$$

Details on $P(T_{ij})$ for general K with empirical validation of Eq (4.9) can be found in Appendix C.1. Note that if GCR is used instead of CID, this is equivalent to setting $P(T_{ij}) = 1/2$ under the equal two community example (since this is the worst case performance for community detection).

4.5 Simulations

We now demonstrate the performance of our proposed design coupled with the difference in means estimator in several simulations. Since community informed design generalizes GCR and independent assignment, we label the methods by the algorithm, $h(\cdot, \cdot)$, used in step 3 of Algorithm 1. Throughout, we focus on comparing

the following:

- **Ind**: Independent random assignment (No regard for network structure)
- **Eps-Net**: Epsilon net (Knows about network structure but does not know about communities, standard GCR)
- **Community Eps-Net with True** U: Perform an epsilon net on each community sub-graph (Knows about network structure and existence of communities, community informed design)
- **Community Eps-Net with Estimated** U: Perform an epsilon net on each community sub-graph by estimated community label (Knows about network structure and existence of communities, community informed design)

While the main goal of this work is bias reduction, we also consider the potential for bias/variance trade-off. Results comparing the root mean squared error (RMSE) are given in Appendix C.6. Consistently our methods maintain lower RMSE than standard GCR and independent randomization when the interference is community driven. In the main text of this section, we report the bias of the different approaches relative to the bias under independent assignment. This is defined as the average absolute bias for the difference in means estimator under a particular method divided by the average absolute bias obtained from independent randomization.

4.5.1 Two Communities

Throughout the simulation in this subsection we investigate networks with two ground truth communities. There are $N = 1600$ individuals in the SBM, split equally between the two communities. The probability of within community edges is set to 0.04, and we vary the probability of cross-community edges to demonstrate the behavior of our

randomization as the community detection problem becomes more difficult. Outcomes are generated following Eq (4.7). We consider three scenarios:

1. Community interference: The true underlying interference mechanism for outcomes is within community level interference. That is, a node is only influenced by the treatment of within community connections, and we let $\gamma = (20/50, 40/50)$.

2. Community agnostic interference: The true underlying interference mechanism for outcomes is full neighborhood level interference. That is, all neighbors influence a node equally, and we let $\gamma = (20/50, 20/50)$.

3. Anti community interference: The true underlying interference mechanism for outcomes is that individuals are only influenced by neighbors that do not share their community and thus Γ has a dis-assortative structure:

$$\Gamma = \begin{bmatrix} 0 & 20/50 \\ 40/50 & 0 \end{bmatrix}.$$

The results are presented in Figures 4.6-4.8. Unsurprisingly, our proposed approach performs exceptionally well in the first scenario when either the true labels are known or the community detection task is easy. Importantly, as the community detection problem becomes more difficult, our proposed approach reduces to the standard GCR method. When there is community agnostic interference, we do not see many gains from the community detection step of CID. The difference between GCR and CID can likely be explained by the fact that there are more within community edges and so the clusters used in CID are tighter and more representative than the ones in GCR. Lastly, we see that our approach suffers substantially when the interference

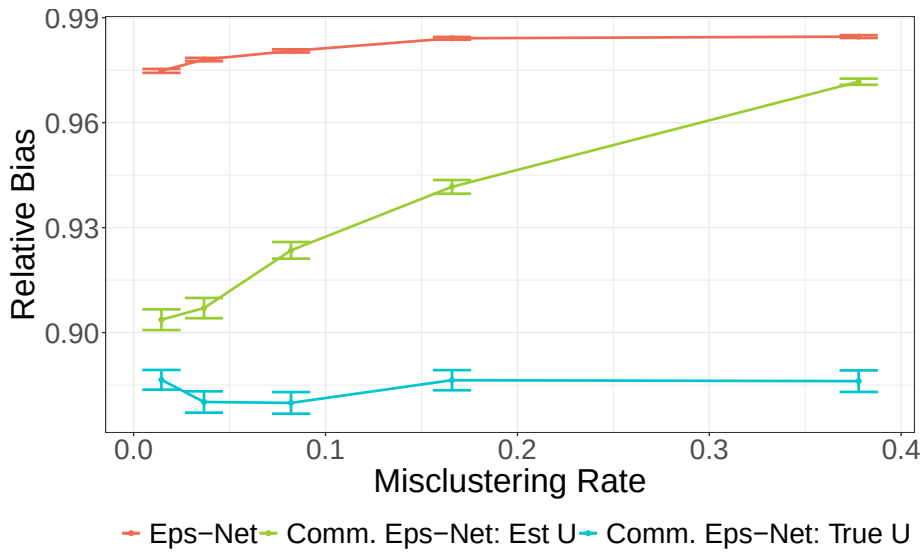

Figure 4.6: Relative bias for 2 equally sized community SBM with community interference. The x-axis is the average misclustering rate for a given regime. The y-axis is relative bias for the difference in means estimator. Standard error bars are over 2000 simulations.

mechanism is completely misspecified. In this setting, the community interference assumption is false and thus there is no reason to expect our design implementation to be successful.

4.5.2 Community Level Average Treatment Effect

Since communities are likely influenced by treatment in different ways, consider community level treatment effects:

$$\tau_k = \frac{1}{N_k} \sum_{i=1}^{N} [Y_i(Z=1) - Y_i(Z=0)] \mathbb{1}[U_{ik} = 1] \tag{4.10}$$

where $N_k = \sum_{i=1}^{N} \mathbb{1}[U_{ik} = 1]$. This quantity can be estimated using a difference in means estimator that only uses nodes in community k (whether those labels are learned or estimated). Through community informed design, these quantities are

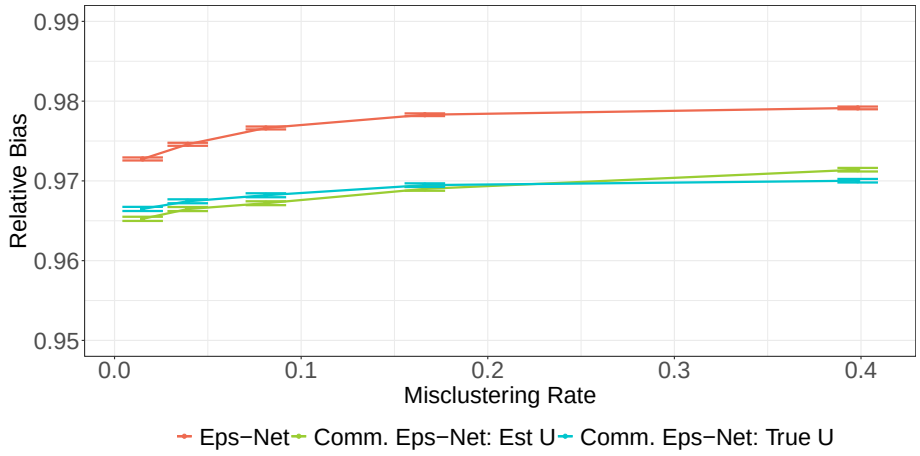

Figure 4.7: Relative bias for 2 equally sized community SBM with homogenous neighborhood interference. The x-axis is the average misclustering rate for a given regime. The y-axis is relative bias for the difference in means estimator. Standard error bars are over 2000 simulations.

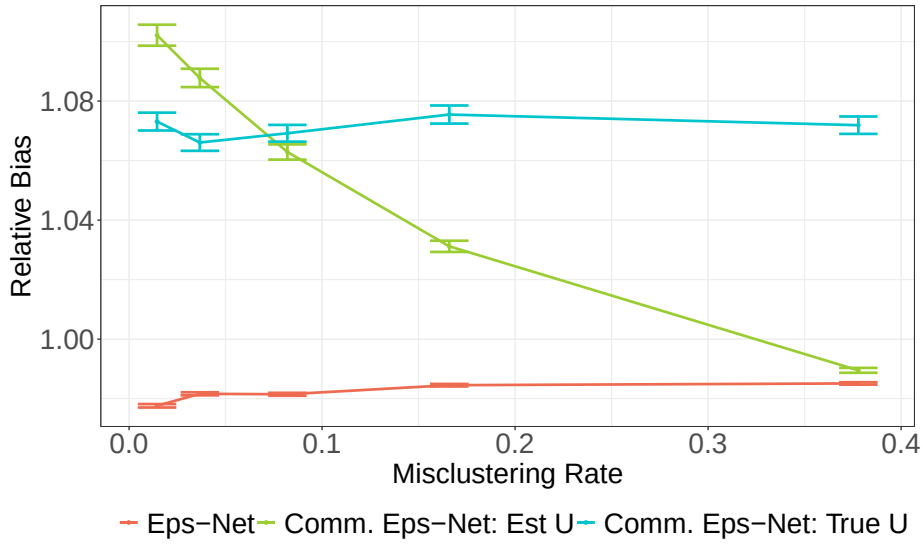

Figure 4.8: Relative bias for 2 equally sized community SBM where response is now influenced by between community neighbors. The x-axis is the average misclustering rate for a given regime. The y-axis is relative bias for the difference in means estimator. Standard error bars are over 2000 simulations.

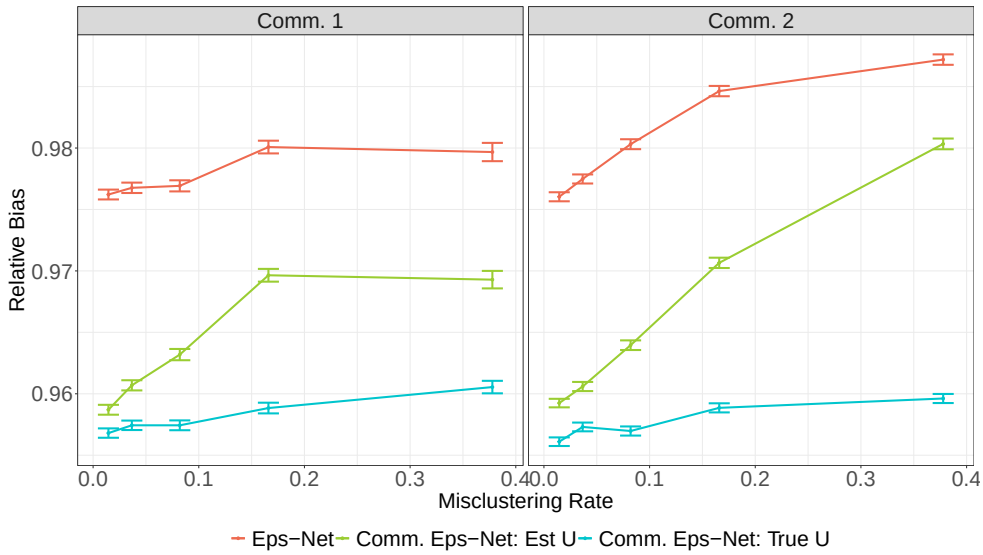

Figure 4.9: The average relative bias for community level treatment effect estimation over 2000 simulations. These community level effects are from the simulation generated in Figure 4.6.

easily obtained. However, for the independent and standard GCR epsilon net methods, community information can only be leveraged after the experiment where outcomes are partitioned by community labels. Figure 4.9 shows relative bias for estimating the community level treatment effects from the simulation set up with community interference in Section 4.5.1. From this experiment, again the benefits of community informed design are apparent.

4.5.3 Varying Number of Communities

For the following simulation, consider performance for different values of K. Let N scale with the number of communities such that there are 250 nodes in each community ($N = 250 \times K$) and let γ depend on K such that a sequence of length K is generated with values between 10/50 and 1. For generating the networks, define the SBM parameters to be $a = .04$ and $b = .03/(K-1)$. Note that N and b are dependent on K to maintain equal expected degree for the network, however slight

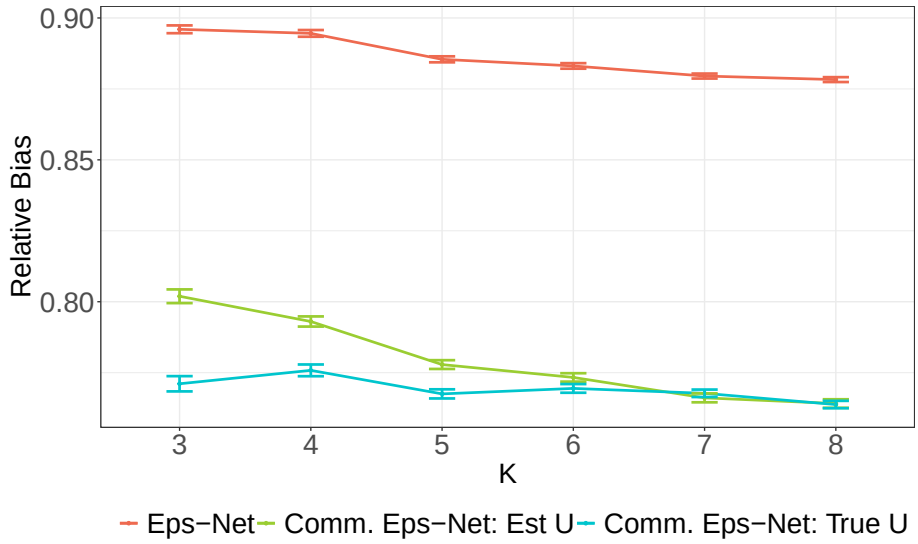

Figure 4.10: The y axis is relative bias for varying K equally sized communities with standard error bars over 2000 simulations per value of K.

variability in outcomes persists due to γ.

Figure 4.10 shows that conditioning on communities is beneficial across values of K. The community detection problem becomes easier as N increases thus the method using estimated U matches that of true U for high K.

4.5.4 No Community Structure nor Interference

Unlike the first three simulations, in this section we consider a network generative model that does not exhibit explicit community structure. Similarly, the outcome model relies on all edges in the network and so the community neighborhood interference assumption is violated. We follow the data generation procedure described in Ugander and Yin (2020) which uses small world networks and a multiplicative response model. Details of parameter specification are provided in Appendix C.5. Again, since the underlying network model does not exhibit meaningful community structure, the estimated communities used in CID are themselves not relevant and so

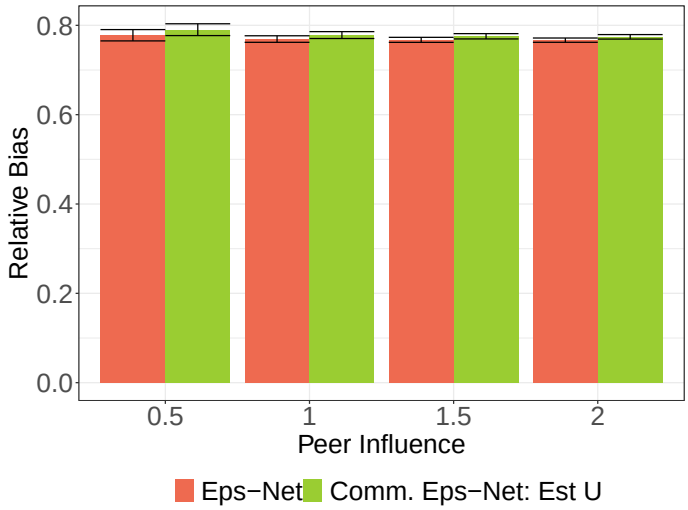

Figure 4.11: Using response model from Ugander and Yin (2020) in their section 6.2. Standard error bars are over 100 simulations. The x-axis represents their peer influence parameter that is given in Appendix C.5.

we do not expect any performance gains from using CID as opposed to standard GCR. Figure 4.11 shows the output of the two approaches relative to independent design: the performance of community informed design with epsilon net nearly matches that of the non community informed version.

4.5.5 Potentially Mismatched Communities

Much of the work on network analysis has been driven by the study of student-student networks (Hoff et al., 2013; Mayer and Puller, 2008; Rienties and Nolan, 2014; Sentse et al., 2014). While grade levels and genders represent natural communities within a network, it is not necessarily the case that these community labels are to be recoverable from observed in-school networks (Mathews and Volfovsky, 2021). Motivated by a recent collection of network-driven experiments on the impact of anti-bullying interventions (Paluck et al., 2016, 2020) we leverage the observed networks and observed grade and gender labels for individuals in those networks to study the

performance of our experimental design.

As part of the original data collection in Paluck et al. (2016), students were asked to record up to 10 friends (for the purposes of this simulation we consider undirected versions of these networks, making the observed degree slightly larger), making the problem of community detection substantially harder since the networks are censored. We study four schools with different sample sizes ($N = 126, 104, 370, 530$) and so for larger N the censoring makes the network appear sparser, again potentially affecting the performance of community detection and our proposed approach.

For each of the schools, outcomes are generated according to Eqn (4.7). Under this model, ground truth communities are defined as unique grade and gender pairs. That is, if two students share grade and gender, they share a community. Under this definition, we assume that interference in outcomes of students are only driven by the treatment of neighbors who share the same grade and gender. K is the count of unique gender-grade pairs in a particular school (ranging from 4 to 6 depending on the grades within the school). Students with NA values for grade and/or gender are dropped from the data.

For each school the within community effect, γ, is a vector of length K with values ranging from 0.4 to 1.6, and the direct effect is $\beta = 1$. As shown in Figure 4.12, leveraging community information reduces relative bias across all of the schools. However, we also note that the differences between standard GCR and community informed methods are not always the same. For example, both Schools A and B are relatively small and we see that CID with estimated community labels nearly matches CID with true labels. On the other hand, we see the smallest improvement between GCR and CID with estimated communities in School D which has the sparsest network, making the community detection problem the hardest.

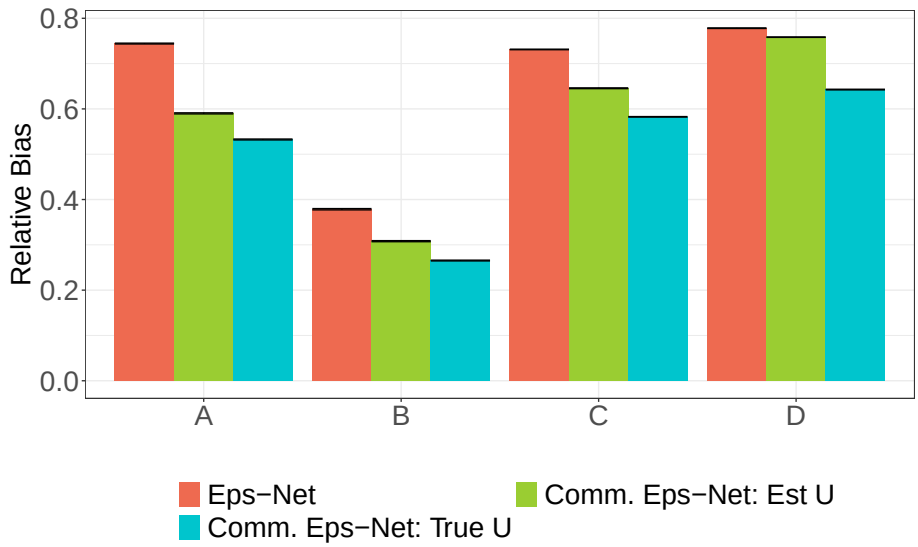

Figure 4.12: Relative bias for various schools from the middle school data set where outcomes are simulated based off of the true observed networks with standard error bars over 750 simulations.

4.6 Discussion

This chapter has proposed a new experimental design that leads to a reduction of bias of the naive difference-in-means estimator in the estimation of the global average treatment effect when interference is community driven. The approach improves on graph cluster randomization techniques by conditioning on the community structure of the graph (estimated or known) and provably reducing the fraction of low-quality randomizations. Importantly, the community interference assumption is meaningful in many applied settings, while realistic violations of it do not lead to a substantive reduction in the quality of the proposed design. In settings where the community structure must be estimated from an observed network, we demonstrate both analytically and empirically that the improvement due to CID decreases as the community detection problem becomes more difficult, but this step does not lead to performance that is worse than naive graph cluster randomization.

Network interference has been recognized as an important obstacle in experimental

design, leading to many recent advances. For example, Zhou et al. (2020) propose a cluster adaptive network testing procedure with a sequential cluster adaptive randomization and a cluster adjusted estimator for the average treatment effect. The proposed approach requires observing additional covariate information on each of the individuals in the network. While this can improve the underlying clustering of the network, such data may be unavailable at times (such as when networks are elicited prior to experimentation) or latent community structure might not be correlated with observed covariates. Another recent approach proposes a cluster based regression adjustment that improves estimation of the GATE as well as testing for interference between individuals (Karrer et al., 2021). They show how tracking exposure to treatment can be used to further reduce variance in estimating the GATE but again rely heavily on the availability of additional side information about the individuals in the study.

Chapter 5

Conclusion and Future Work

In this work, we have exemplified the importance of community structure in networks and the impact that communities have on statistical inference. We developed a novel algorithm for community detection in challenging regimes through leveraging new theory and Gaussian Mixture Models. This was followed by integrating community detection into network regression by extending the AMEN model. Through latent community adaptive network regression, it is clear that different communities of individuals can have heterogenous mechanisms for connection. We also explored how to navigate the complexities of implementing such a model. Finally, we developed a community informed design for high quality experimentation procedures when community structure is present in a network. Communities are a fundamental part of society and understanding them provides us with key insights for making informed decisions from policy implementation to product design. Additionally, there are many possible extensions to this work, a few of which are described below.

5.1 Community Detection with Multiple Networks

A natural extension of Chapter 2 is to consider when we have access to multiple networks for the same set of nodes. As mentioned in Reeves et al. (2019), access to side information can make the task of community detection more feasible. In preliminary work, we show that a single summary is generally not enough for optimizing information from two networks on the same set of nodes. To illustrate this, consider networks

generated from:

$$A_{ij}^{(s)}|X_i, X_j \sim \text{Bern}\left(\frac{d}{N} + \frac{\sigma}{N} X_i^T R^{(s)} X_j\right), \quad i < j, \tag{5.1}$$

where (s) indexes the network with $s \in \{1, ..., S\}$ and other parameters are as in Chapter 2. The proposed approach is to consider multiple linear combinations of networks and to leverage the spectra of these combinations. Consider linear combinations of the form

$$B(w) = \sum_s w_s \frac{A^{(s)} - d/n 11^T}{\sigma} \tag{5.2}$$

where w represents an S dimensional vector of weights. The goal is to consider the joint distribution of the eigenvectors of multiple linear combinations in order to better estimate community memberships. To build intuition, consider two networks where

$$R^{(1)} \propto \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \; R^{(2)} \propto \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Figure 5.1 plots two sets of eigenvectors: on the left those of the sum of the two adjacency matrices and on the right those of the difference of the two adjacency matrices. It is clear that adding the two hides the information about community 2 while subtracting the two hides the information about community 1. However, looking at both of them simultaneously allows for separating all three communities.

While more work is needed to determine the exact distributions for the multiple GMMs, approximations have demonstrated that studying multiple linear combinations of adjacency matrices provides a significant signal boost over existing methods.

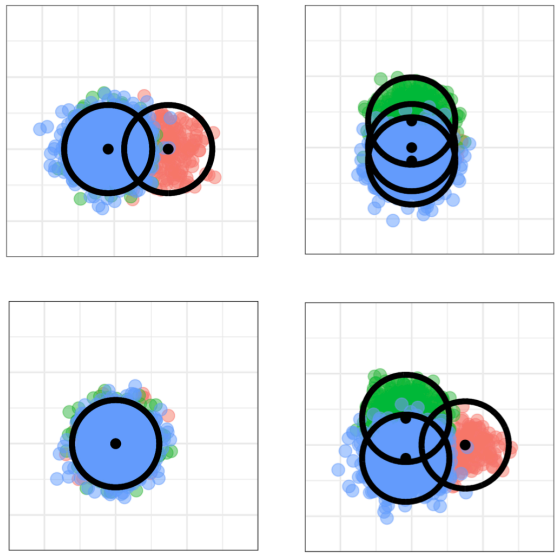

Figure 5.1: Plots of eigenvectors of sums and differences of two adjacency matrices. The points are colored according to their true community labels. The black circles represent identity covariances. Each combination individually gives us partial recovery of communities, but, together, we are able to separate all three communities.

5.2 Extensions for Latent Community Network Regression

Possible extensions for the community adaptive network regression model include allowing for U and V to be continuous for a more flexible model. For example, rather than binary values, each row of U could be defined as a vector of membership probabilities. Then, the resulting value of $\tilde{\beta}U^T$ would be a mixture of $\tilde{\beta}$. While more flexible, we lose the interpretability that is given with the binary form. Another extension would allow for time dependence in the community structure (which has been demonstrated in some social networks, e.g. Xin et al., 2017) as well as in the covariate coefficients. This in turn introduces another dependence structure that can be challenging to account for. Finally, this model can be used for prediction. Often we are interested in possible future connections between individuals. Given the Bayesian framework used, predictive posterior distributions are easily obtained and predicted

network structure can be studied.

5.3 Other Estimators for Community Informed Design

Given the recent focus on covariates in the literature, it is a natural future direction to incorporate covariate information into the community informed design procedure. This can be done by incorporating covariates into the community detection problem directly (Binkiewicz et al., 2017; Shen et al., 2022; Yan and Sarkar, 2021) or by considering community detection as a component of a network regression problem (Hoff, 2008; Mathews and Volfovsky, 2021). These approaches can further be coupled with post-hoc estimators that adjust for the potential community structure that may not have been accounted for during the design phase of an experiment. However, development of post-hoc estimators is not a trivial task due to potential (and likely) imbalance at the community level. An adjustment term or a weighted estimator might be useful if model assumptions can be made. Since Algorithm 1 in Chapter 4 involves the use of spectral methods, another extension would be to relate the cost of misclustered nodes more explicitly to recoverability thresholds which is briefly explored in Figure C.5 of Appendix C.6. Finally, this work focuses on the difference in means estimator; however much work has been done using others such as versions of Horvitz-Thompson estimators. While these can be useful, they are expensive to compute and tend to lead to high variance in practice.

Appendix A

Spectral Theory and SBM

A.1 Parameter Estimation for the SBM

Consider the case where we do not know p and Q explicitly but are given side information in the form of a small labeled network that is generated from the same SBM. Such side information is extremely likely to be available in practice as network data are rarely collected without some knowledge of underlying labels for a small fraction of individuals. In this setting, we can estimate Q and p conditional on those known labels. In this section, we employ a nonparametric histogram estimator for Q and p and show that it performs well in practice. The estimation procedure is as follows: for the labeled graph, use the empirical fraction of members of each community as the estimate $\hat{p}$ of p. To estimate $Q_{(i)}$, order the adjacency matrix according to the known labels and estimate $\hat{Q}_{(i)}$ by $\sum_{x_i=a,x_j=b} a_{ij} / \sum_{X_i=a,X_j=b} 1$. Such histogram estimators are known to be consistent for the SBM (Chan and Airoldi, 2014). Each data point in Figure A.1 represents an average over 250 estimates generated for a graph of size n with $p = \begin{pmatrix} 1/10 & 3/10 & 6/10 \end{pmatrix}$ and

$$Q = \begin{pmatrix} 0.058 & 0.010 & 0.010 \\ 0.010 & 0.022 & 0.012 \\ 0.010 & 0.012 & 0.017 \end{pmatrix}.$$

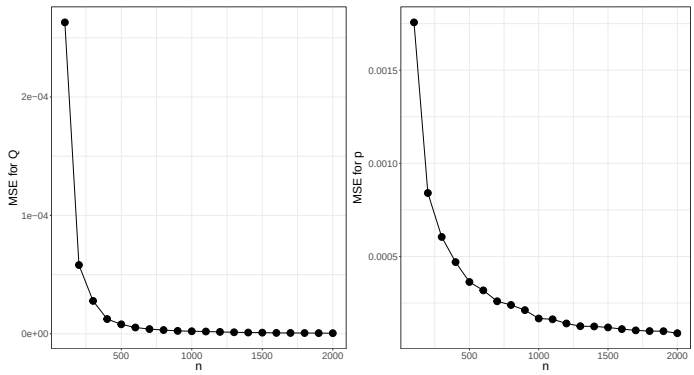

Figure A.1: MSE plots for estimating p and Q.

A.2 Phase Transitions for Whitened Representation of the SBM

To exemplify the phase transition that occurs when $|r| < 1$, Figure A.2 shows misclustering rate (using our GMM from Chapter 2) on SBMs where the values of r_1 and r_2 are varied. We can see that when $|r_i| < 1$, recoverability better than random guess is not possible.

A.3 Additional Real Data Example

We also implement our GMM clustering algorithm on various middle school networks from Paluck et al. (2016, 2020) (which is also considered in Chapter 4). In this example, we take grade level to be the ground truth communities. As a result, we consider 4 middle schools that each have 3 grade levels. Again, we make these networks undirected.

Figure A.3 compares misclustering rate for k-means and various GMM implementations on the middle school networks with the size of the networks ranging from 80 to

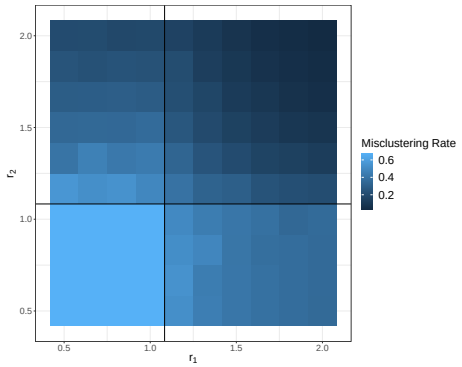

Figure A.2: Empirical demonstration of phase transitions for recoverability for the whitened representation of the SBM as used in Chapter 2. The low noise GMM with known parameters is used for this example where $N = 1000$, $K = 3$, $d = 30$, $p = [1/3, 1/3, 1/3]$.

462 students. Labels for 25% of the network are used to estimate parameters for the GMM. In the figure, the schools are ordered by N, with school A having the smallest size. For school A in particular, note that the uninformed GMM and the GMM with estimated parameters perform the worst as expected due to the small network size. As N increases, we see the benefits of the GMM become more prevalent.

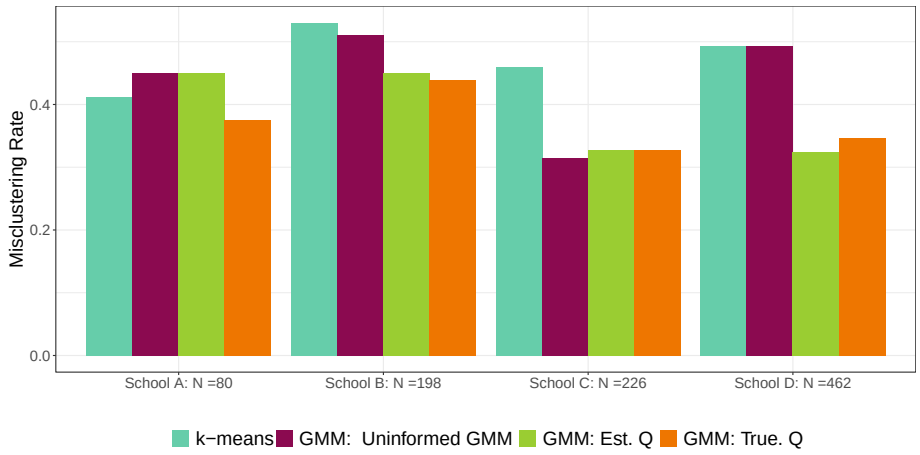

Figure A.3: Misclustering rates for various algorithms implemented on 4 middle schools (with varying N) from the middle school network data. Note that overall, GMM methods perform better than k-means (especially as N grows).

Appendix B

Network Regression

B.1 Vectorization of $\text{diag}(\tilde{\beta}_r U^T)X_r$

For deriving full conditionals needed to sample our parameters of interest, it is convenient to rewrite $\text{diag}(\tilde{\beta}_r U^T)X_r$ in vectorized form (for simplicity we consider $p = 1$)

$$\begin{aligned} vec(\text{diag}(\tilde{\beta}_r U^T)X_r) &= vec(\text{diag}(U\tilde{\beta}_r^T)X_r) \\ &= vec((I_N \circ U\tilde{\beta}_r^T 1_N^T)X_r) \\ &= vec([I_N \circ X_r]U\tilde{\beta}_r^T 1_N^T) \\ &= (1_N \otimes [I_N \circ X_r]U)vec(\tilde{\beta}_r). \end{aligned}$$

The main quantities of importance are:

$$\text{diag}(U\tilde{\beta}_r^T) = (I_N \circ U\tilde{\beta}_r^T 1_N^T)$$

and

$$(I_N \circ U\tilde{\beta}_r 1_N^T)X_r = (I_N \circ X_r)U\tilde{\beta}_r 1_N^T$$

which hold due to the structure of $U\tilde{\beta}_r^T 1_N^T$.

Note that the above is specifically for row covariates. A similar calculation is available for column covariates:

$$\text{diag}(V\tilde{\beta}_c^T) = (I_N \circ V\tilde{\beta}_c^T 1_N^T)$$

$$X_c(I_N \circ V\tilde{\beta}_c 1_N^T) = (V\tilde{\beta}_c 1_N^T)^T (I_N \circ X_c),$$

$$\begin{aligned} vec(X_c \operatorname{diag}(\tilde{\beta}_c V^T)) &= vec(X_c(I_N \circ V\tilde{\beta}_c^T 1_N^T)) \\ &= vec((V\tilde{\beta}_c 1_N^T)^T (I_N \circ X_c)) \\ &= vec(1_N \tilde{\beta}_c^T V^T [I_N \circ X_c]) \\ &= ([I_N \circ X_c]V \otimes 1_N) vec(\tilde{\beta}_c). \end{aligned}$$

B.2 Including Dyadic Covariates

Often, dyadic covariates are observed or created from row/column covariates. A dyadic covariate, X_d, is a $N \times N$ matrix where each entry $x_{d,ij}$ describes some notion of how i relates to j. For example, observe the row/column covariate, grade. A dyadic covariate can be created for grade such that $X_{d,ij} = 1$ if i and j share the same grade, otherwise it is 0.

Incorporating dyadic covariates in the standard AMEN model is simple but is more complicated when coefficients can be community dependent. To begin, consider a dyadic covariate that has a special form. Again, for notational convenience, $p = 1$ for X_r, X_c, X_d. Let X be a n-dimensional vector and define a dyadic covariate as $X_d = XX^T$. To allow for community dependence, define:

$$\operatorname{diag}(\tilde{\beta}_{dr} U^T) X_d \operatorname{diag}(V\tilde{\beta}_{dc}^T) = \operatorname{diag}(\tilde{\beta}_{dr} U^T) XX^T \operatorname{diag}(\tilde{\beta}_{dc} V^T).$$

The full model then becomes:

$$\begin{aligned} Z = \beta_0 11^T + \textstyle\sum_{l=1}^p (\operatorname{diag}(\tilde{\beta}_{rl} U^T) X_{rl} + X_{cl} \operatorname{diag}(\tilde{\beta}_{cl} V^T) + \\ \operatorname{diag}(U\tilde{\beta}_{dr,l}^T) X_d \operatorname{diag}(V\tilde{\beta}_{dc,l}^T)) + U\Lambda V^T + a1^T + b1^T + E. \end{aligned}$$

Given the special structure of the dyadic covariate, the following holds:

$$\text{diag}(\tilde{\beta}_{dr}U^T)X = \text{diag}(X)U\tilde{\beta}_{dr}^T.$$

Thus, by substitution:

$$\begin{aligned}\text{diag}(\tilde{\beta}_{dr}U^T)XX^T\,\text{diag}(\tilde{\beta}_{dc}V^T) &= \text{diag}(X)U\tilde{\beta}_{dr}^T X^T\,\text{diag}(\tilde{\beta}_{dc}V^T)\\ &= \text{diag}(\tilde{\beta}_{dr}U^T)X[\tilde{\beta}_{dc}V^T\,\text{diag}(X)].\end{aligned}$$

To derive full conditionals, vectorize the dyadic term to isolate $\tilde{\beta}_{dr}, \tilde{\beta}_{dc}$:

$$vec(\text{diag}(X)U\tilde{\beta}_{dr}^T X^T\,\text{diag}(\tilde{\beta}_{dc}V^T) = \{[X^T\,\text{diag}(\tilde{\beta}_{dc}V^T)]^T \otimes \text{diag}(X)U\}vec(\tilde{\beta}_{dr})$$

$$vec(\text{diag}(\tilde{\beta}_{dr}U^T)X[\tilde{\beta}_{dc}V^T\,\text{diag}(X)]) = \{\text{diag}(X)V \otimes \text{diag}(\tilde{\beta}_{dr}U^T)X\}vec(\tilde{\beta}_{dc}).$$

Define:

$$\tilde{Z} = Z - \beta_0 11^T - (\text{diag}(\tilde{\beta}_r U^T)X_r + X_c\,\text{diag}(\tilde{\beta}_c V^T)) - U\Lambda V^T - a1^T - b1^T,$$

which is decorrelated:

$$\tilde{Z}_* = s \times \tilde{Z} + t \times \tilde{Z}^T$$

where

$$H_{dc} = s \times \{\text{diag}(X)V \otimes \text{diag}(\tilde{\beta}_{dr}U^T)X\} + t \times \{[\text{diag}(\tilde{\beta}_{dr}U^T)X] \otimes \text{diag}(X)V\},$$

$$H_{dr} = s \times \{\text{diag}(\tilde{\beta}_{dc}V^T)X \otimes \text{diag}(X)U\} + t \times \{\text{diag}(X)U \otimes \text{diag}(\tilde{\beta}_{dc}V^T)X\}.$$

The full conditionals for $\tilde{\beta}_{dc}$ and $\tilde{\beta}_{dr}$ are:

$$\tilde{\beta}_{dr}|X_c, X_r, U, \Lambda, Z, \tilde{\beta}_{dc}, a, b \sim N(V_{dr}m_{dr}, V_{dr})$$

$$\tilde{\beta}_{dc}|X_c, X_r, U, \Lambda, Z, \tilde{\beta}_{dr}, a, b \sim N(V_{dc}m_{dc}, V_{dc})$$

where

$$\begin{array}{ll} V_{dc} = (H_{dc}^T H_{dc} + \Sigma_{\tilde{\beta}_{dc}}^{-1})^{-1} & V_{dr} = (H_{dr}^T H_{dr} + \Sigma_{\tilde{\beta}_{dr}}^{-1})^{-1} \\ m_{dc} = vec(\tilde{Z}_*)^T H_{dc} + \mu_{\tilde{\beta}_{dc}} \Sigma_{\tilde{\beta}_{dc}}^{-1} & m_{dr} = vec(\tilde{Z}_*)^T H_{dr} + \mu_{\tilde{\beta}_{dr}} \Sigma_{\tilde{\beta}_{dr}}^{-1} \end{array}.$$

B.3 General Dyadic Covariates

Consider a generalized dyadic covariate, X_d, that does not take on an outer product form. Rewrite $X_d = Q\Sigma W^T$ (singular value decomposition of X_d where Σ is a diagonal matrix of singular values) and thus $X_d = \sum_{i=1}^n \sigma_i q_i w_i^T$ where σ_i are the diagonal elements of Σ. As such,

$$\text{diag}(\tilde{\beta}_{dr}U^T)X_d \,\text{diag}(\tilde{\beta}_{dc}V^T) = \text{diag}(\tilde{\beta}_{dr}U^T)\sum_{i=1}^N \sigma_i q_i w_i^T \,\text{diag}(\tilde{\beta}_{dc}V^T).$$

The above can be expanded to:

$$\text{diag}(\tilde{\beta}_{dr}U^T)\sigma_1 q_1 w_1^T \,\text{diag}(\tilde{\beta}_{dc}V^T) + ... + \text{diag}(\tilde{\beta}_{dr}U^T)\sigma_N q_N w_N^T \,\text{diag}(\tilde{\beta}_{dc}V^T).$$

Isolate $\tilde{\beta}_{dc}$:

$$\text{diag}(\tilde{\beta}_{dr}U^T)\sigma_1 q_1 w_1^T \,\text{diag}(\tilde{\beta}_{dc}V^T) = \text{diag}(\tilde{\beta}_{dr}U^T)\sigma_1 q_1[\tilde{\beta}_{dc}V^T \,\text{diag}(w_1)]$$

and vectorize to get:

$$vec(\text{diag}(\tilde{\beta}_{dr}U^T)X_d[\tilde{\beta}_{dc}V^T\,\text{diag}(X)]) = \{\text{diag}(w_1)V \otimes \text{diag}(\tilde{\beta}_{dr}U^T)\sigma_1 q_1\}vec(\tilde{\beta}_{dc}).$$

For $\tilde{\beta}_{dr}$,

$$\text{diag}(\tilde{\beta}_{dr}U^T)\sigma_1 q_1 w_1^T\,\text{diag}(\tilde{\beta}_{dc}V^T) = \text{diag}(\sigma_1 q_1)U\tilde{\beta}_{dr}^T w_1^T\,\text{diag}(V\tilde{\beta}_{dc}^T),$$

which can be vectorized as

$$vec(\text{diag}(\tilde{\beta}_{dr}U^T)\sigma_1 q_1 w_1^T\,\text{diag}(\tilde{\beta}_{dc}V^T)) = \{[w_1^T\,\text{diag}(V\tilde{\beta}_{dc}^T)]^T \otimes \text{diag}(\sigma_1 q_1)U\}vec(\tilde{\beta}_{dr}).$$

Now, to derive the full conditionals, first define analogues to equations 3.9 and 3.10 from Chapter 3.3.2:

$$H_{dc} = s\times\sum_{i=1}^{n}\{\text{diag}(w_i)V\otimes\text{diag}(\tilde{\beta}_{dr}U^T)\sigma_i q_i\}+t\times\sum_{i=1}^{n}\{[w_i^T\,\text{diag}(\tilde{\beta}_{dr}U^T)]^T\otimes\text{diag}(\sigma_i q_i)V\},$$

and

$$H_{dr} = s\times\sum_{i=1}^{n}\{[w_i^T\,\text{diag}(\tilde{\beta}_{dc}V^T)]^T\otimes\text{diag}(\sigma_i q_i)U\}+t\times\sum_{i=1}^{n}\{\text{diag}(w_i)U\otimes\text{diag}(\tilde{\beta}_{dc}V^T)\sigma_i q_i\}.$$

The full conditionals for $\tilde{\beta}_{dr}$ and $\tilde{\beta}_{dc}$ are given by:

$$\tilde{\beta}_{dr}|X_c, X_r, U, \Lambda, Z, \tilde{\beta}_{dc}, a, b \sim N(V_{dr}m_{dr}, V_{dr})$$

$$\tilde{\beta}_{dc}|X_c, X_r, U, \Lambda, Z, \tilde{\beta}_{dr}, a, b \sim N(V_{dc}m_{dc}, V_{dc})$$

where

$$V_{dc} = (H_{dc}^T H_{dc} + \Sigma_{\tilde{\beta}_{dc}}^{-1})^{-1} \qquad V_{dr} = (H_{dr}^T H_{dr} + \Sigma_{\tilde{\beta}_{dr}}^{-1})^{-1}$$

$$m_{dc} = vec(\tilde{Z}_*)^T H_{dc} + \mu_{\tilde{\beta}_{dc}} \Sigma_{\tilde{\beta}_{dc}}^{-1} \qquad m_{dr} = vec(\tilde{Z}_*)^T H_{dr} + \mu_{\tilde{\beta}_{dr}} \Sigma_{\tilde{\beta}_{dr}}^{-1}.$$

B.4 Bivariate Probit Likelihood

Let $M_{ij} = \beta_0 + \sum_{l=1}^{p}(x_{irl}\tilde{\beta}_{rl}U_i^T + x_{jcl}\tilde{\beta}_{cl}V_j^T) + a_i + b_j + U_i^T \Lambda V_j$. The likelihood is then:

$$\begin{aligned}
&\log(L(\beta, U, V, \Lambda, a, b|X, A)) = \textstyle\sum_{i<j}(Y_{ij}Y_{ji}\log(\Phi(M_{ij}, M_{ji}, \rho))+ \\
&(1 - Y_{ij})Y_{ji}\ log(\Phi(-(M_{ij}), M_{ji}, -\rho))+ \\
&Y_{ij}(1 - Y_{ji}\log(\Phi(M_{ij}, -(M_{ji}), -\rho))+ \\
&(1 - Y_{ij})(1 - Y_{ji})\log(\Phi(-(M_{ij}), -(M_{ji}), \rho)))+ \\
&\textstyle\sum_{i=1}^{n}\log(1 - \Phi(m_{ii}, \sqrt{1+\rho}))
\end{aligned} \tag{B.1}$$

where $\Phi(x, y, \rho)$ indicates evaluating the cumulative distribution function of the bivariate normal distribution.

B.5 Comparison of Censored Binary Formulations

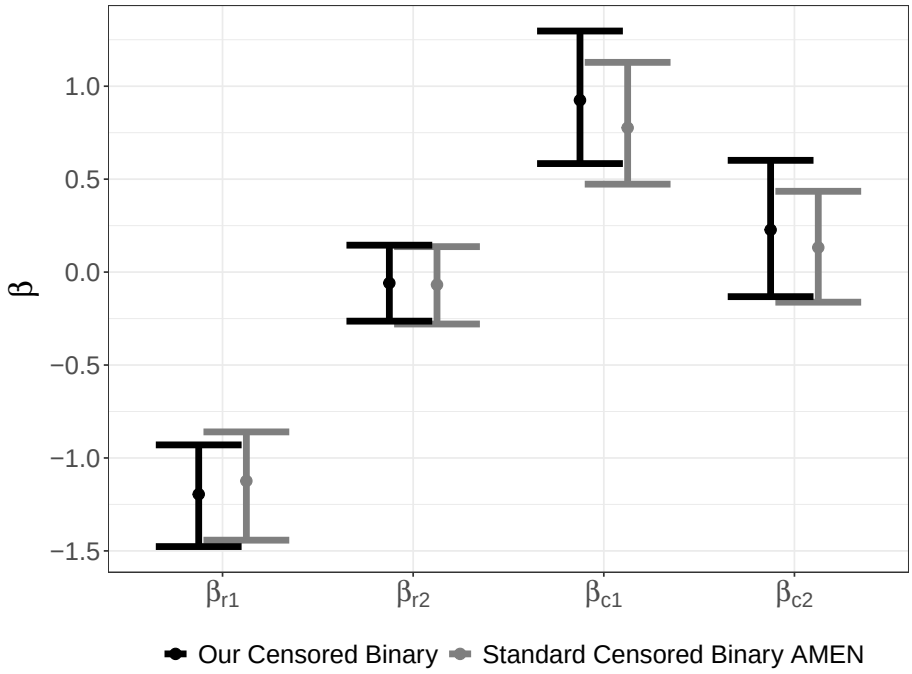

Figure B.1: 95% posterior CI for row and column coefficients under our version of the censored binary versus that of the standard censored binary AMEN model as in the 'amen' package. Data is generated as in Section 3.4.3.

A censored binary likelihood for networks was introduced in Hoff et al. (2013), leveraging the set-based likelihood machinery of that paper. The formulation in that work does not naturally lend itself to our proposed updates for community dependent coefficients. In this section we demonstrate that the two approaches lead to the same parameter estimates (in a non-community dependent setting). The standard version of the model matches the data generating mechanism for the censored network structure as in Chapter 3.4.3. That is, once Z values are simulated, $Y_{ij} = 1$ if Z_{ij} is in the top m $Z_{i.}$ values. Both models yield near equivalent posterior estimates for all parameters of interest as shown in Figure B.1. As an additional note, recall that in Chapter 3.4.3, there are community dependent covariate effects that are not accounted

for in these non-community dependent models. Specifically, β_{r2} and β_{c2} vary between communities. The posterior estimates for these coefficients are approximately 0 for both estimation approaches (which is the average of the community dependent $\tilde{\beta}$ values).

B.6 Covariate Information for AddHealth

In the AddHealth study, high school students (grades 9-12) ranked their top 5 female and top 5 male friends. We analyze the same-sex male-to-male friendships in two schools. Covariates of interest include:

- Race — an indicator of whether the individual is white
- Drinking — an indicator of whether a an individual drinks
- Smoking — an indicator of whether an individual smokes
- GPA — numeric grade point average
- Living situation
  - Living with father
  - Living with mother
  - Number of people in household
- Club involvement — number of clubs a student is involved in
- Sport involvement — number of sports a student is involved in

We further include dyadic covariates for whether or not two students share the same grade, share the same binary drinking behavior, share the same binary smoking behavior, and whether they share the same race.

B.7 Trace plots for AddHealth Networks

Geweke z-statistics assessing the convergence of the Markov Chain Monte Carlo algorithm are reported in the main text. Figure B.2 presents traceplots for several parameters in the model for visual inspection of the stationarity of the chain.

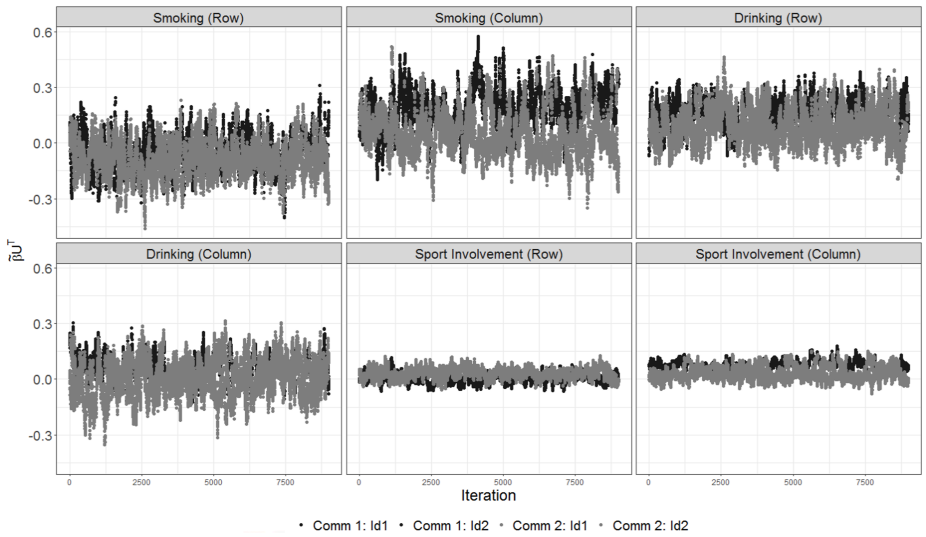

Figure B.2: Trace plot for 4 individuals' $U_i\tilde{\beta}$ for multiple covariates from school B. The nodes belonging to the same community overlap so two distinct colors are used to distinguish between communities.

B.8 Dyadic Posterior Credible Intervals for AddHealth Networks

Each of Figures B.3-B.10 present the post-processed posterior credible intervals for the community dependent dyadic covariates. In each figure, the top left panel (labeled $\tilde{\beta}_{dr,1}\tilde{\beta}_{dc,1}$) represents the coefficient for a pair of individuals where both sender and

receiver are in community 1. The top right panel (labeled $\tilde{\beta}_{dr,1}\tilde{\beta}_{dc,2}$) represents the coefficient for a pair of individuals where the sender is in community 1 and the receiver is in community 2. The bottom left figure (labeled $\tilde{\beta}_{dr,2}\tilde{\beta}_{dc,1}$) represents the coefficient for a pair of individuals where the sender is in community 2 and the receiver is in community 1. The bottom right panel (labeled $\tilde{\beta}_{dr,2}\tilde{\beta}_{dc,2}$) represents the coefficient for a pair of individuals where both sender and receiver are in community 2.

B.8.1 School A

The following figures show the 95% credible intervals for school A for dyadic covariates. As with the row and column effects for this school, the dyadic covariates for this school exhibit little to no community structure.

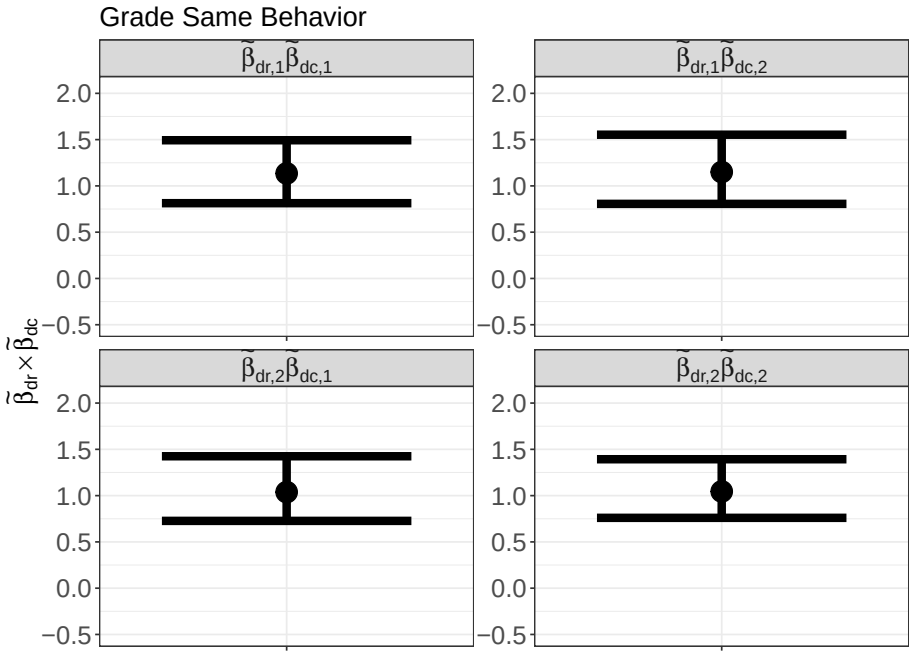

Figure B.3: School A: 95% CI for dyadic grade covariates with community dependent model, $K = 2$

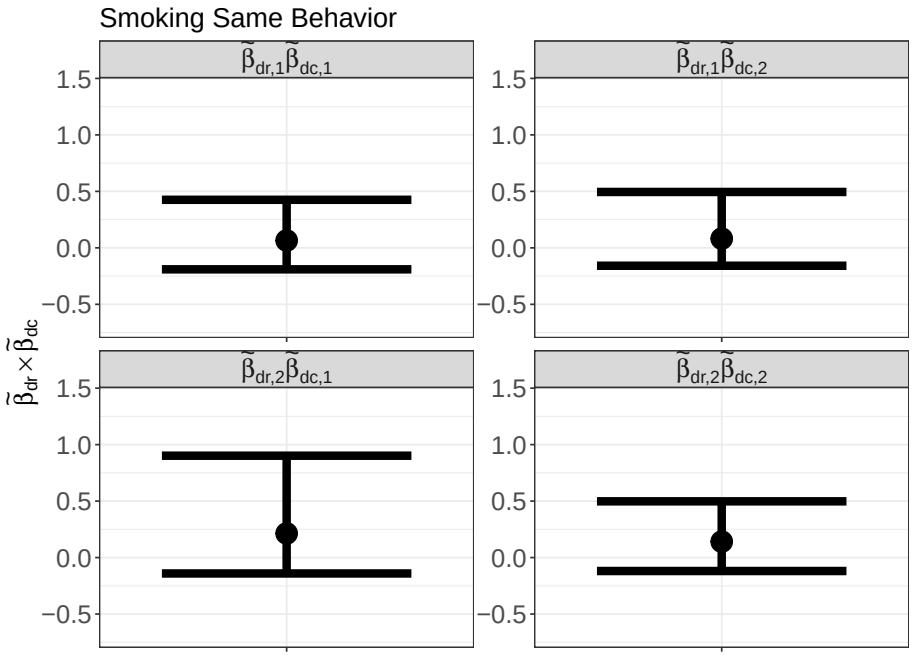

Figure B.4: School A: 95% CI for dyadic smoking covariates with community dependent model, $K = 2$

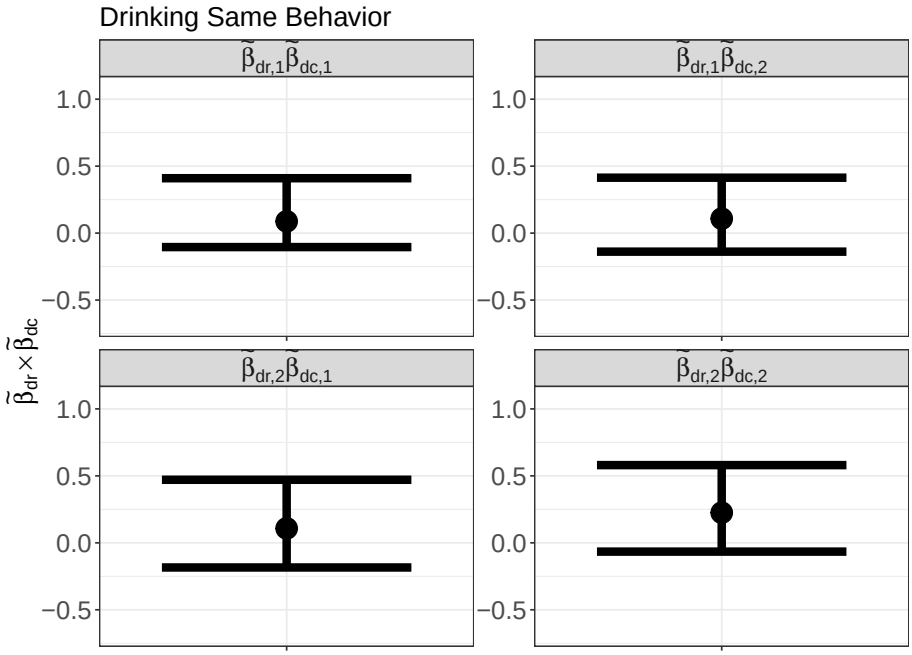

Figure B.5: School A: 95% CI for dyadic drinking covariates with community dependent model, $K = 2$

Figure B.6: School A: 95% CI for dyadic race covariates with community dependent model, $K = 2$

B.8.2 School B

The following figures show the 95% credible intervals for school B for dyadic covariates. We note that community dependent behavior is exhibited for the smoking and drinking behavior: pairs where both sender and receiver are in community 1 have positive effects on friendship formation when both are smokers or drinkers. This behavior is completely absent for smoking when both sender and receiver are in community 2 and is largely absent for drinking when at least the sender of ties is in community 2.

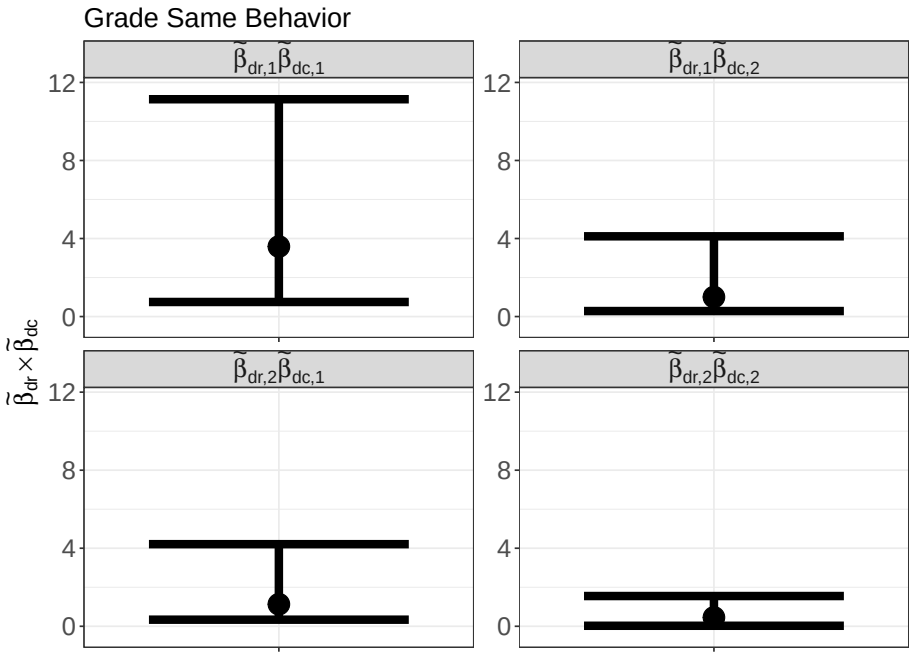

Figure B.7: School B: 95% CI for dyadic grade covariates with community dependent model, $K = 2$

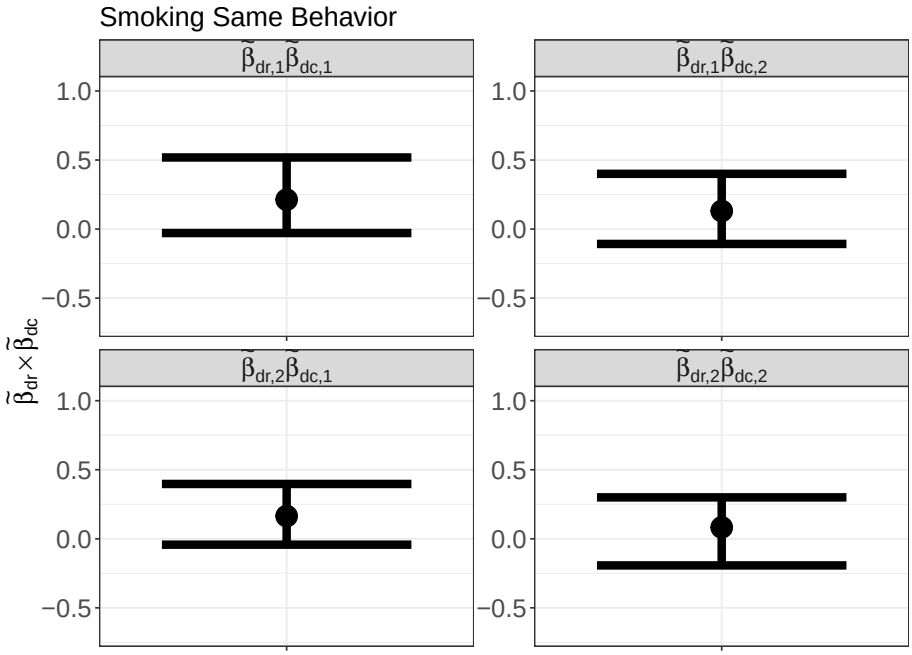

Figure B.8: School B: 95% CI for dyadic smoking covariates with community dependent model, $K = 2$

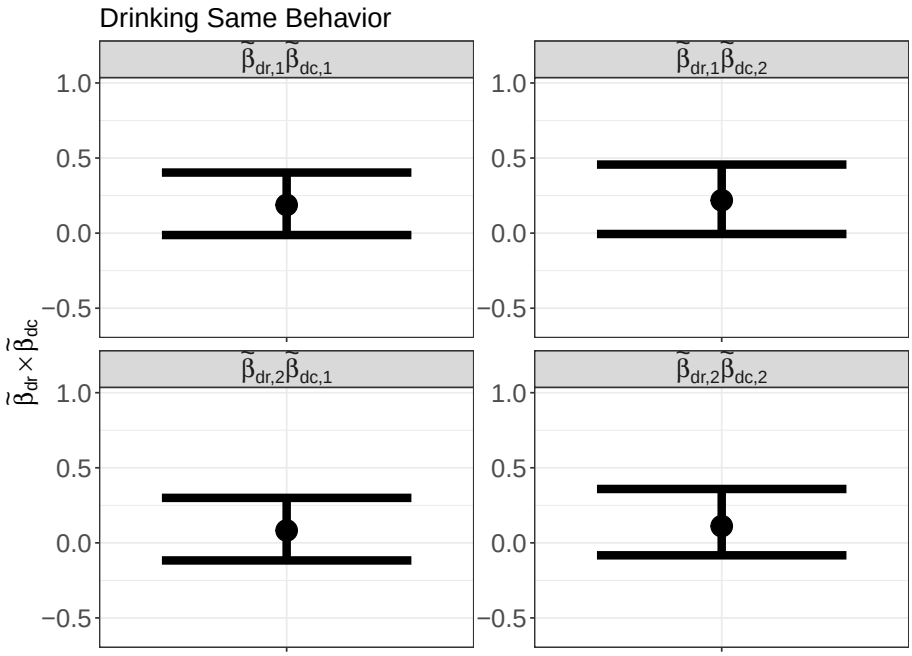

Figure B.9: School B: 95% CI for dyadic drinking covariates with community dependent model, $K = 2$

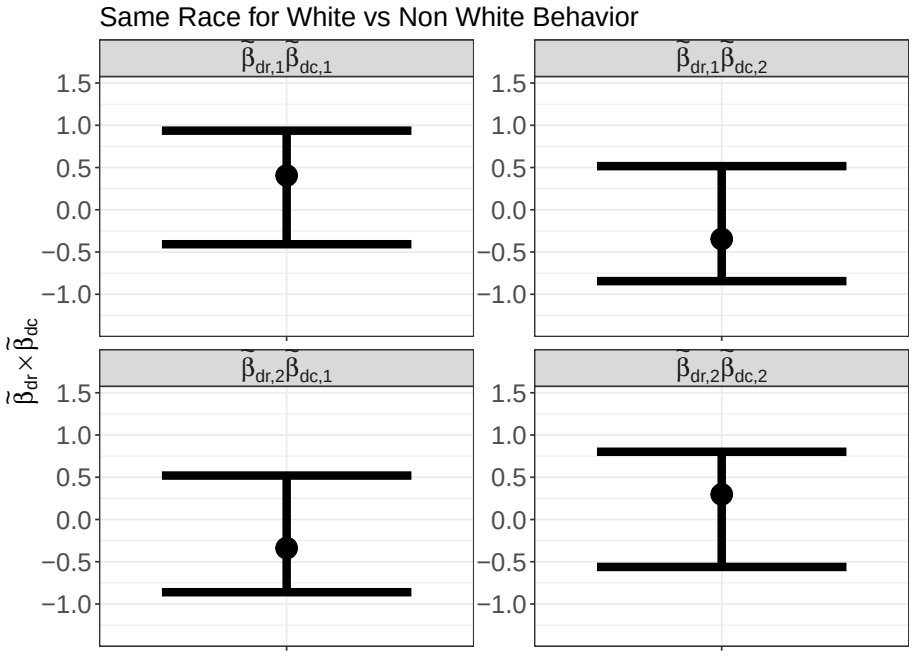

Figure B.10: School B: 95% CI for dyadic race covariates with community dependent model, $K = 2$

B.9 Covariate Assisted Clustering

When covariate information is available, it can potentially provide additional information about community structure. In Binkiewicz et al. (2017), they consider what happens when we observe a network and covariate information about the nodes. In order to add in covariate information for clustering, Binkiewicz et al. (2017) alter the Laplacian such that:

$$L(\alpha) = L_\tau + \alpha X X^T$$

where X is a $N \times p$ matrix of covariate information and $L_\tau = D_\tau^{-1/2} A D_\tau^{-1/2}$ with $D_\tau = D + \tau I_{N \times N}$. Let $\tau = \sum_i D_{ii}/N$. The goal is that this additional information will be representative of the true underlying block structure and thus will provide boosted signal, making the recovery task simpler.

Appendix C

Experimentation

C.1 Expected Cost of Misclustering

Let T_{ij} denote the event that $U_i = U_j$ and $\hat{U}_i \neq \hat{U}_j$. For node i, the expected loss due to the event T_{ij} is:

$$\begin{aligned} E\left[\sum_{j=1}^{N} A_{ij} Z_j \gamma U_i \mathbb{1}_{T_{ij}}\right] &= E\left[\sum_{j=1}^{N} A_{ij} Z_j \gamma U_i | T_{ij}\right] P(T_{ij}) \\ &= \sum_{j=1}^{N} E[A_{ij} Z_j \gamma U_i | T_{ij}] P(T_{ij}) \\ &= N \times a \times q \times E[\gamma U_i | T_{ij}] P(T_{ij}) \\ &= P(T_{ij}) \times N \times a \times q \times \sum_{k=1}^{K} \gamma_k P_k \end{aligned}$$

Note that when $\hat{U}_i \neq \hat{U}_j$, Z_i and Z_j are independent; thus $E[Z_j|T_{ij}] = q$. Also, $E[A_{ij}|T_{ij}] = a$ since $U_i = U_j$. The probability of T_{ij} depends heavily on the network structure and algorithm $f(\cdot, \cdot)$ (for label recovery). If we define D_k to be the number of misclustered nodes in community k and N_k to be the number of nodes in true community k, for $K = 2$:

$$P(T_{ij}) = \frac{E[\sum_{k=1}^{K} (N_k - D_k) \times 2D_k]}{N^2}$$

For general K, we need to consider which community misclustered nodes belong to. Let M be a $K \times K$ matrix with $k, l \in \{1, ..., K\}$ such that:

$$M_{kl} = \sum_{j=1}^{N} \mathbb{1}_{\hat{U}_j \neq U_j} \times \mathbb{1}_{\hat{U}_{jl}=1} \times \mathbb{1}_{U_{jk}=1}. \tag{C.1}$$

This yields a matrix that determines how many misclustered nodes in community k were placed into each other community. Note that this yields 0 for diagonal elements. For a node, i, belonging to community k that is correctly clustered, loss is associated with D_k since they will not share a cluster with the misclustered nodes. For a node, i, belonging to community k that is incorrectly clustered, loss is associated with the number of people in their true community but not in their estimated one, $N_k - M_{kw}$ where w is the estimated community that i belongs to. Assume that N_k is known and we have K equally sized communities. If we considered the worst case scenario for clustering (random guess), then for the overall number of misclustered nodes, D, $E[D] = N\Big(1 - \frac{1}{K}\Big)$. For example, if we have 1500 nodes and 3 equally sized communities, under random guess, 1000 nodes will be misclustered. We can now consider the number of misclustered nodes per community. Given N_k,

$$\begin{aligned} E[D_k] &= E\Big[\sum_{i:U_{ik}=1} \mathbb{1}_{\hat{U}_i \neq U_i}\Big] \\ &= N_k\Big(1 - \frac{1}{K}\Big), \end{aligned}$$

and therefore

$$E[D_k^2] = N_k\Big(1 - \frac{1}{K}\Big)\frac{1}{K} + E[D_k]^2.$$

Similarly,

$$E[M_{kw}] = E\Big[\sum_{i:U_{ik}=1} \mathbb{1}_{\hat{U}_i \neq U_i, \hat{U}_{iw}=1}\Big],$$
$$= N_k\left(\frac{1-1/K}{K-1}\right)$$

and

$$E[M_{kw}^2] = N_k\Big(\frac{1-1/K}{K-1}\Big)\Big(1-\frac{1-1/K}{K-1}\Big) + E[M_{kw}]^2.$$

Combining the above yields:

$$P(T_{ij}) = \frac{\sum_k \Big(N_k E[D_k] - E[D_k^2] + \sum_{w\neq k} N_k E[M_{kw}] - E[M_{kw}^2]\Big)}{N^2}. \tag{C.2}$$

In practice, the overall $E[D]$ would depend on the algorithm of choice. For Figure C.1, we control the proportion of misclustered nodes. In this case, $1-1/K$ would be replaced with this proportion. Under random assignment, for a misclustered node, we assume that a node has equal probability of being classified into the $K-1$ other communities. Figure C.1 validates Eq (4.9) as it shows expected versus observed cost due to misclustering nodes for a $N = 1000$ node network generated by a 2 equally sized community SBM. Across the x-axis, the proportion of permuted nodes is increasing. If 30% of nodes are permuted, 300 nodes are randomly selected and their true labels are swapped (since there are only 2 communities in this example). This vector of permuted labels acts as an estimated label thus showing what can happen as number of mislabelled nodes increases.

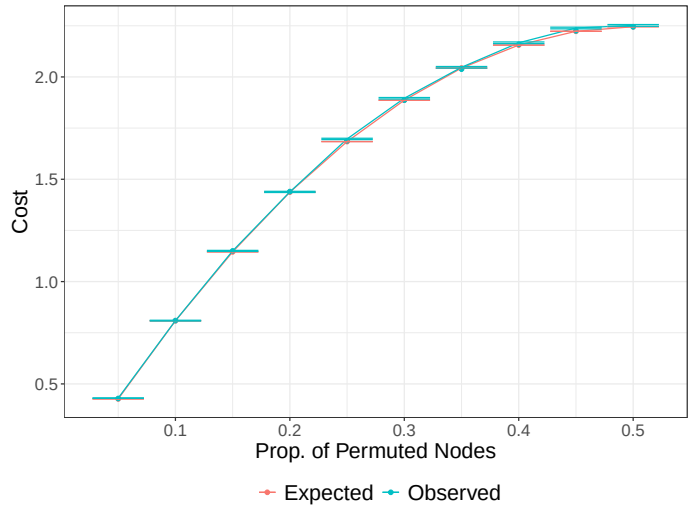

Figure C.1: Comparing observed and expected cost from mispaired nodes. Y-axis is cost which is the part of the bias that is only associated with community effects. Standard error bars are over 450 simulations.

C.2 Exposure Probabilities with Community Interference Under 3-Net

We are able to estimate exposure probabilities, $P(E_i^0)$ and $P(E_i^1)$ under community informed design. Using Monte Carlo methods, we generate M clusterings as in Ugander and Yin (2020). Similar to their work, define a random generator for clusterings as $\mathcal{P}$. To tie with the results from Chapter 4.3, note that $P(E_i^0) = P(Z_i = 0, G_i = 0)$ and $P(E_i^1) = P(Z_i = 1, G_i = d_i^c)$. The exposure probabilities can then be estimated:

$$P[E_i^Z|\mathcal{P}] = \frac{1}{M}\sum_{m=1}^{M} P[E_i^Z|c^{(m)}]$$

where c is a realized clustering. Following from their Theorem 4.2, this is easily extended for our model when Γ is diagonal. When this is true, the proper network exposure is within community network interference. Using the 3-net algorithm on community induced sub-graphs, the within community full-neighborhood exposure probabilities for any node i satisfy (extended from Ugander and Yin (2020)):

$$P[E_i^1|\mathcal{P}] \geq \frac{q}{|B_{2|G_k(i)}(i)|} \geq \frac{q}{|B_2(i)|} \geq \frac{q}{(1+d_{max})\kappa}$$

where κ is a restricted graph growth coefficient, d_{max} is the maximum degree of all nodes, and define $B_r(i)$ be the r hop neighborhood for i and $B_{r|G_k}(i)$ be the r hop neighborhood for i in the sub-graph for community k, G_k. Similarly,

$$P[E_i^0|\mathcal{P}] \geq \frac{1-q}{|B_{2|G_k}(i)|} \geq \frac{1-q}{|B_2(i)|} \geq \frac{1-q}{(1+d_{max})\kappa}.$$

The above is a straightforward result as $|B_2(i)| \geq |B_{2|G_k}(i)|$. Also, Theorem 4.5 from Ugander and Yin (2020) can be extended for bounding the relative error of the Monte Carlo estimator for the exposure probabilities:

$$E\left[\frac{\hat{P}[E_i^1|\mathcal{P}]-P[E_i^1|\mathcal{P}]}{P[E_i^1|\mathcal{P}]}\middle|\mathcal{P}\right]^2 \leq \frac{|B_{2|G_k}(i)|}{Mq} \leq \frac{|B_2(i)|}{q}.$$

C.3 Variance of Difference in Means Estimator

Consider the variance of the difference in means estimator with community interference. Again, let N_T be the number of treated individuals and N_C be the number of not treated (control) individuals where these quantities are assumed to be fixed. For deriving the variance, the notion of network exposure is required as this relates to the probability of treatment. For notational simplicity in this section, we use $P(E_i^1)$ to define the probability that node i is network exposed to treatment. Under our interference assumption, this is interpreted as the probability that i and all of is within community neighbors are treated. Details on these probabilities can be found

in the previous section. The variance can then be expressed as:

$$Var\Big[\frac{1}{N_T}\sum_{i=1}^{N} Y_i(1)Z_i - \frac{1}{N_C}\sum_{i=1}^{N} Y_i(0)(1-Z_i)\Big] =$$
$$\frac{1}{N_T^2}\left[\sum_{i=1}^{N}(P(E_i^1)(1-P(E_i^1))Y_i(1)^2 + \sum_{i=1}^{N}\sum_{\substack{j:U_i=U_j\\ j\neq i}}^{N}(P(E_i^1\cap E_j^1) - P(E_i^1)P(E_j^1))Y_i(1)Y_j(1)\right] +$$
$$\frac{1}{N_C^2}\left[\sum_{i=1}^{N}(P(E_i^0)(1-P(E_i^0))Y_i(0)^2 + \sum_{\substack{j:U_i=U_j\\ j\neq i}}^{N}(P(E_i^0\cap E_j^0) - P(E_i^0)P(E_j^0))Y_i(0)Y_j(0)\right] -$$
$$2\times\frac{1}{N_T N_C}\left[\sum_{i=1}^{N}\sum_{\substack{j:U_i=U_j\\ j\neq i}}^{N}(P(E_i^1\cap E_j^0) - P(E_i^1)P(E_j^0))Y_i(1)Y_j(0) - \sum_{i=1}^{N} Y_i(1)Y_i(0)\right] \tag{C.3}$$

which follows similarly to arguments presented in Aronow and Samii (2017) and Eckles et al. (2016). Notice that for the covariance terms, we only include pairs of individuals who share the same community. This is because individuals in different communities do not belong to the same cluster under community informed design. As such, $P(E_i^z \cap E_j^z) = P(E_i^z)P(E_j^z)$ when i and j are in different communities (which still holds when using estimated labels).

Following a similar argument to that of Lemma B.6 and Theorem 4.6 in Ugander and Yin (2020), consider independent randomization at the cluster level using the one hop max algorithm. While this is not the main algorithm used in this work for cluster randomization, results using one hop max are shown in Figure C.2, exhibiting similar behavior to epsilon net. Under one hop max,

$$\text{if } dist(i,j) > 4, \text{ then } P(E_i^z \cap E_j^z) = P(E_i^z)P(E_j^z).$$

If $dist(i,j) > 4$, this means that nodes i, j will not be in the same cluster and thus their exposure probabilities are independent. This holds trivially under community informed design.

Let all responses for $i \in \{1, ..., N\}$ be within $[0, \bar{Y}]$ then:

$$Var(\hat{\mu}(1)) \leq \frac{\bar{Y}^2}{N_T}\left[(1 + d_{max})\kappa^3 - q\kappa^2\right] \tag{C.4}$$

where κ is a restricted growth coefficient. Restricted growth assumptions are required to ensure that the number of neighbors a node has does not grow too quickly. If a node is connected to too many different clusters, this causes an increase in variance. As such, this condition allows the variance to be bounded tighter than by the maximum degree. The full derivation is as follows:

$$\begin{aligned}
&\frac{1}{N_T^2}\left[\sum_{i=1}^{N}(P(E_i^1)(1 - P(E_i^1))Y_i(1)^2 + \right. \\
&\left. \sum_{i=1}^{N}\sum_{j:\{U_i=U_j\},j\neq i}^{N}(P(E_i^1 \cap E_j^1) - P(E_i^1)P(E_j^1))Y_i(1)Y_j(1)\right] \leq \\
&\frac{\bar{Y}^2}{N_T^2}\left[\sum_{i=1}^{N}(P(E_i^1)(1 - P(E_i^1)) + \sum_{j:\{U_i=U_j\},j\in B_4(i),j\neq i}^{N}(P(E_i^1 \cap E_j^1) - P(E_i^1)P(E_j^1))\right] \leq \\
&\frac{\bar{Y}^2}{N_T^2}\left[\sum_{i=1}^{N}(P(E_i^1)(1 - P(E_i^1)) + \sum_{j:\{U_i=U_j\},j\in B_4(i),j\neq i}^{N}(P(E_j^1) - P(E_i^1)P(E_j^1))\right] \leq \\
&\frac{\bar{Y}^2}{N_T^2}\left[\sum_{i=1}^{N}(P(E_i^1)(1 - P(E_i^1))) + \sum_{j:\{U_i=U_j\},j\in B_4(i),j\neq i}^{N}(1 - P(E_i^1))\right] \leq \\
&\frac{\bar{Y}^2}{N_T^2}\left[\sum_{i}|B_4(i)|(1 - P(E_i^1))\right] \leq \\
&\frac{\bar{Y}^2}{N_T}\left[(1 + d_{max})\kappa^3 - q\kappa^2\right].
\end{aligned}$$

From the exposure probabilities, $(1-P(E_i^1)) \leq 1-\frac{q}{(1+d_{max})\kappa}$ and $|B_4(i)| \leq (1+d_{max})\kappa^3$.

Similarly, we derive $Var(\hat{\mu}(0)) \leq \frac{\bar{Y}^2}{N_C}\left[(1 + d_{max})\kappa^3 - (1 - q)\kappa^2\right]$. Combining the

above yields results following closely with Theorem 4.8 in Ugander and Yin (2020),

$$Var(\hat{\tau}) \leq 2\bar{Y}^2\kappa^2\left[(1+d_{max})\kappa\Big(\frac{1}{N_T}+\frac{1}{N_C}\Big)-\Big(\frac{q}{N_T}+\frac{1-q}{N_C}\Big)\right]. \tag{C.5}$$

However, note that under community informed design, $|B_4(i)|$ becomes $|B_{4|G_k}(i)|$ and thus $d_{max} \geq \max_{k\in\{1,\ldots,K\}} d_{max|G_k}$ due to the partitioning of the network into sub-graphs. Additionally, exposure probabilities under community informed design are larger than those of standard graph cluster randomization under full neighborhood exposure. However, if standard graph cluster randomization assumed community neighborhood exposure, the latter would be a reduction in both designs while the former is exclusive to community informed design. As such, we provide a tighter variance bound under our design:

$$2\bar{Y}^2\kappa^2\left[(1+d_{max})\kappa\Big(\frac{1}{N_T}+\frac{1}{N_C}\Big)-\Big(\frac{q}{N_T}+\frac{1-q}{N_C}\Big)\right] \geq \tag{C.6}$$

$$2\bar{Y}^2\kappa^2\left[(1+\max_{k\in\{1,\ldots,K\}} d_{max|G_k})\kappa\Big(\frac{1}{N_T}+\frac{1}{N_C}\Big)-\Big(\frac{q}{N_T}+\frac{1-q}{N_C}\Big)\right]. \tag{C.7}$$

C.4 Algorithms

Algorithm 2 Community Epsilon Net

```
Initialize C to be N length vector of zeros
Q is a N length vector of community labels
for k ∈ 1 : K do
  J = {i ∈ {1 : N} : Q_i = k}
  G_k = G[J, J]
  S ← ∅, unmarked nodes
  π ← uniformly draw a random ordering of nodes
  for i ∈ π do
    if i is unmarked then
      S ← S ∪ i
      for j ∈ B_{2|G_k}(i) where B_{2|G_k}(i) is the 2 hop neighborhood for i in G_k do
        mark node j if unmarked
      end for
    end if
  end for
  for i ∈ J do
    c_i ← argmin{j ∈ S, j → dist(i, j)}
  end for
  C[which(i ∈ k)] ← c_i + max(C) to get unique cluster levels onces k is greater
  than 1
end for
Output: C
```

C.5 Generative Model from Ugander et al

Below is the generative model for outcomes that is used in the small world example from Ugander and Yin (2020):

$$Y_i(z) = Y_i(0) \times \left(1 + \delta Z_i + \gamma \frac{\sum_{j \in N(i)} Z_j}{d_i}\right) \tag{C.8}$$

where $N(i)$ is the set of direct neighbors of i, d_i is the degree of node i and:

$$Y_i(0) = (a + b \times h_i + \sigma \times \epsilon_i) \times \frac{d_i}{\bar{d}}. \tag{C.9}$$

The parameters b, δ, γ, σ are scaling parameters and a is a baseline parameter. h yields a notion of homophily that is described in detail below. The true treatment effect is then $(\delta + \gamma) \times Y_i(0)$. This model is thus specified by full network exposure. For the simulations, $a = 1$, $b = .5$, $\sigma = .1$, $\delta = .5$, and γ is the peer influence parameter that is varied. Explicit details for generating the network are given in Ugander and Yin (2020). It is important to note that the network, A, is generated using a small world model. While this model includes homophily, it is not leveraged when performing clustering. Without explicit inclusion of this, we run the risk of creating imbalanced clusters across the true communities that might exist in the network. Additionally, this structure is not driven by the treatment allocated to homophilous pairs of individuals. To construct h, they solve:

$$D^{-1}Lv_2 = \lambda_2 v_2 \tag{C.10}$$

for v_2 which is then normalized to come from a $N(0, 1)$ (and thus becomes h).

C.6 Additional Figures

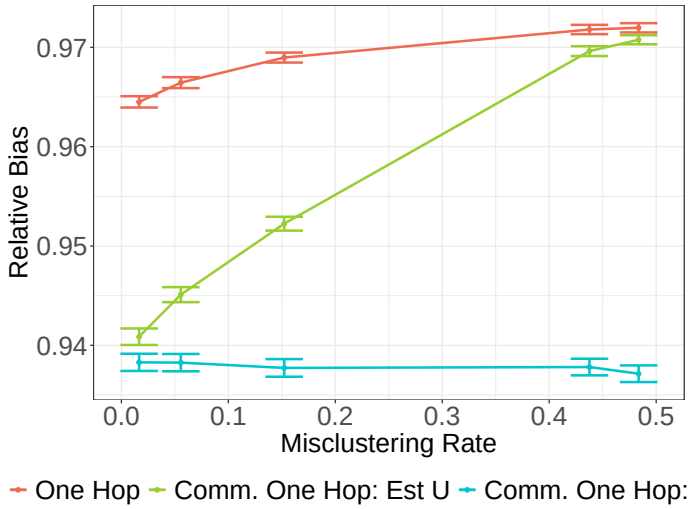

Figure C.2: Relative bias for the experiment presented in the equal sized two community simulation with community interference using one hop max algorithm.

Figure C.2 demonstrates relative bias under various designs, similar to the community interference 2 community SBM example in Chapter 4.5. However, for this simulation, the algorithm for generating clusters is the one hop max algorithm as described in Ugander and Yin (2020). From this figure, we see that the performance under one hop max is similar to that of epsilon net.

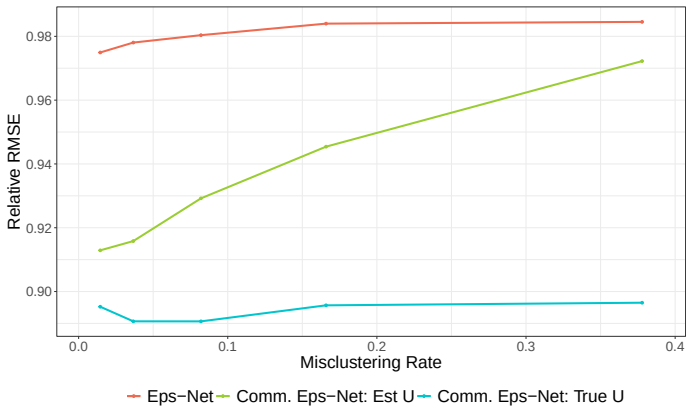

Figure C.3: Relative RMSE for the experiment presented in the equal sized two community simulation with community interference.

While the majority of Chapter 4 focuses on bias reduction, we also consider the root mean squared error (RMSE) of the various methods. Figure C.3 shows the RMSE for the community interference simulation in Chapter 4. This demonstrates how our methods show significant reduction in RMSE as well.

Figure C.4: Example of a middle school network that is colored by estimated community label where $K = 4$.

Figure C.4 shows an example network for a middle school where nodes are colored by their estimated community labels. This figure shows how this middle school does exhibit community structure.

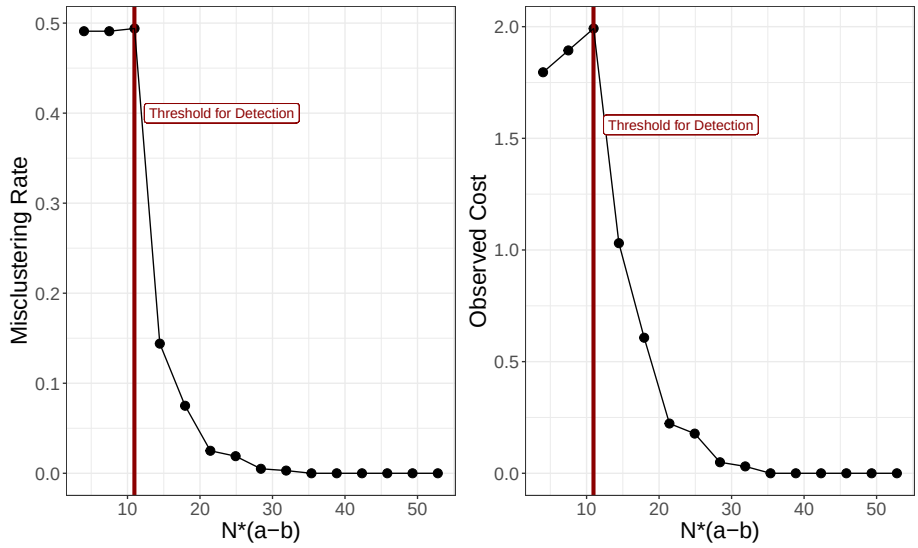

Figure C.5: The plot on the left shows misclustering rate from k-means as in Chapter 4 versus the difference in expected between and within community degree for a 2 equally sized community SBM where the threshold for detection is labelled, similar to that shown in Figure 2.5 in Chapter 2. The plot on the right shares the same x-axis, however the observed cost of misclustered nodes is as discussed in Appendix C.1.

We also briefly discussed the possible ties between the phase transitions seen with community detection (as in Figure 2.5 in Chapter 2) and the observed cost of misclustering nodes on GATE estimation. To make this more explicit, the left plot of Figure C.5 shows the difference in expected within and between community degree versus misclustering rate (using k-means as implemented in Chapter 4) which exhibits a phase transition at the threshold for detection. The right plot shows a similar behavior at the threshold for detection, however this plot investigates the cost to GATE estimation that is attributed to misclustering nodes (as in Appendix C.1). Note that there is some variability due to the different γ values; however this figure suggests that there is possibly a similar phase transition for cost of misclustered nodes as expected.

Bibliography

Abbe, E. (2017). Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531.

Abbe, E., Bandeira, A. S., and Hall, G. (2015). Exact recovery in the stochastic block model. *IEEE Transactions on information theory*, 62(1):471–487.

Abbe, E. and Sandon, C. (2015). Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 670–688. IEEE.

Amati, V., Lomi, A., and Mira, A. (2018). Social network modeling. *Annual Review of Statistics and Its Application*, 5:343–369.

Aronow, P. M. and Samii, C. (2017). Estimating average causal effects under general interference, with application to a social network experiment. *The Annals of Applied Statistics*, 11(4):1912–1947.

Athreya, A., Priebe, C. E., Tang, M., Lyzinski, V., Marchette, D. J., and Sussman, D. L. (2016). A limit theorem for scaled eigenvectors of random dot product graphs. *Sankhya A*, 78(1):1–18.

Aukett, R., Ritchie, J., and Mill, K. (1988). Gender differences in friendship patterns. *Sex roles*, 19(1-2):57–66.

Awan, U., Morucci, M., Orlandi, V., Roy, S., Rudin, C., and Volfovsky, A. (2020). Almost-matching-exactly for treatment effect estimation under network interference. In *International Conference on Artificial Intelligence and Statistics*, pages 3252–3262. PMLR.

Bail, C. A., Argyle, L. P., Brown, T. W., Bumpus, J. P., Chen, H., Hunzaker, M. F., Lee, J., Mann, M., Merhout, F., and Volfovsky, A. (2018). Exposure to opposing views on social media can increase political polarization. *Proceedings of the National Academy of Sciences*, 115(37):9216–9221.

Basse, G. W. and Airoldi, E. M. (2018). Model-assisted design of experiments in the presence of network-correlated outcomes. *Biometrika*, 105(4):849–858.

Benaych-Georges, F. and Nadakuditi, R. (2012). The singular values and vectors of low rank perturbations of large rectangular random matrices. *Journal of Multivariate Analysis*, 111:120–135.

Benaych-Georges, F. and Nadakuditi, R. R. (2011). The eigenvalues and eigenvectors of finite, low rank perturbations of large random matrices. *Advances in Mathematics*,

227(1):494–521.

Bhattacharyya, S. and Bickel, P. J. (2014). Community detection in networks using graph distance. *arXiv preprint arXiv:1401.3915*.

Bickel, P. J. and Chen, A. (2009). A nonparametric view of network models and newman–girvan and other modularities. *Proceedings of the National Academy of Sciences*, pages pnas–0907096106.

Binkiewicz, N., Vogelstein, J., and Rohe, K. (2017). Covariate-assisted spectral clustering. *Biometrika*, 104(2):361–377.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Boguna, M., Pastor-Satorras, R., Diaz-Guilera, A., and Arenas, A. (2003). Emergence of clustering, correlations, and communities in a social network model. *arXiv preprint cond-mat/0309263*.

Bruna, J. and Li, X. (2017). Community detection with graph neural networks. *stat*, 1050:27.

Caltagirone, F., Lelarge, M., and Miolane, L. (2017). Recovering asymmetric communities in the stochastic block model.

Chamberlain, B., Kasair, C., and Rotheram-Fuller, E. (2007). Involvement or isolation? the social networks of children with autism in regular classrooms. *J Autism Dev Disord*, 37(2).

Chan, S. and Airoldi, E. (2014). A consistent histogram estimator for exchangeable graph models. In *International Conference on Machine Learning*, pages 208–216. PMLR.

Decelle, A., Krzakala, F., Moore, C., and Zdeborová, L. (2011a). Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6):066106.

Decelle, A., Krzakala, F., Moore, C., and Zdeborová, L. (2011b). Inference and phase transitions in the detection of modules in sparse networks. *Physical Review Letters*, 107(6):065701.

Dunbar, R. I., Arnaboldi, V., Conti, M., and Passarella, A. (2015). The structure of online social networks mirrors those in the offline world. *Social networks*, 43:39–47.

Dunn, J., Davies, L. C., O'Connor, T. G., and Sturgess, W. (2001). Family lives

and friendships: The perspectives of children in step-, single-parent, and nonstep families. *Journal of Family Psychology*, 15(2):272.

Eckles, D., Karrer, B., and Ugander, J. (2016). Design and analysis of experiments in networks: Reducing bias from interference. *Journal of Causal Inference*, 5(1).

Eisenberg, D. (2003). Evaluating the effectiveness of policies related to drunk driving. *Journal of policy Analysis and management*, 22(2):249–274.

El Alaoui, A., Krzakala, F., and Jordan, M. (2020). Fundamental limits of detection in the spiked wigner model. *The Annals of Statistics*, 48(2):863–885.

Erdös, P. and Rényi, A. (1959). On random graphs publ. *Math. debrecen*, 6:290–297.

Farine, D. R. and Whitehead, H. (2015). Constructing, conducting and interpreting animal social network analysis. *Journal of Animal Ecology*, 84(5):1144–1163.

Fortunato, S. (2010). Community detection in graphs. *Physics reports*, 486(3-5):75–174.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, 97(458):611–631.

Girvan, M. and Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826.

Goldenberg, A., Zheng, A. X., Fienberg, S. E., Airoldi, E. M., et al. (2010). A survey of statistical network models. *Foundations and Trends® in Machine Learning*, 2(2):129–233.

Grinberg, N., Joseph, K., Friedland, L., Swire-Thompson, B., and Lazer, D. (2019). Fake news on twitter during the 2016 us presidential election. *Science*, 363(6425):374–378.

Harris, K., Halpern, E., Whitsel, E., Hussey, J., and Udry, J. (2009). The national longitudinal study of adolescent health: Research design.

Hoff, P. (2007). Extending the rank likelihood for semiparametric copula estimation. 1:265–283.

Hoff, P. (2008). Modeling homophily and stochastic equivalence in symmetric relational data. In *Advances in neural information processing systems*, pages 657–664.

Hoff, P. (2018a). Additive and multiplicative effects network models. *arXiv preprint arXiv:1807.08038*.

Hoff, P. (2018b). *sbgcop: Semiparametric Bayesian Gaussian Copula Estimation and Imputation*. R package version 0.980.

Hoff, P. (2021). Additive and multiplicative effects network models. *Statistical Science*, 36(1):34–50.

Hoff, P., Fosdick, B., Volfovsky, A., and He, Y. (2017). *amen: Additive and Multiplicative Effects Models for Networks and Relational Data*. R package version 1.3.

Hoff, P., Fosdick, B., Volfovsky, A., and Stovel, K. (2013). Likelihoods for fixed rank nomination networks. *Network Science*, 1(03):253–277.

Hoff, P., Raftery, A., and Handcock, M. (2002). Latent space approaches to social network analysis. *Journal of the american Statistical association*, 97(460):1090–1098.

Holland, P., Laskey, K., and Leinhardt, S. (1983). Stochastic blockmodels : First steps. *Social Networks*, 5:109–137.

Holland, P. and Leinhardt, S. (1981). An exponetial family of probability distributions for directed graphs. *Journal of the American Statistical Association*, 76:33–50.

Hollenbach, F. M., Bojinov, I., Minhas, S., Metternich, N. W., Ward, M. D., and Volfovsky, A. (2018). Multiple imputation using gaussian copulas. *Sociological Methods & Research*, page 0049124118799381.

Honaker, J., King, G., Blackwell, M., et al. (2011). Amelia ii: A program for missing data. *Journal of statistical software*, 45(7):1–47.

Hurn, M., Justel, A., and Robert, C. P. (2003). Estimating mixtures of regressions. *Journal of computational and graphical statistics*, 12(1):55–79.

Igarashi, T., Takai, J., and Yoshida, T. (2005). Gender differences in social network development via mbile phone text messages: A longitudinal study. *Journal of Social and Personal Relationships*, 22:691–713.

Jagadeesan, R., Pillai, N. S., and Volfovsky, A. (2020). Designs for estimating the treatment effect in networks with interference. *The Annals of Statistics*, 48(2):679–712.

Karrer, B., Shi, L., Bhole, M., Goldman, M., Palmer, T., Gelman, C., Konutgan, M., and Sun, F. (2021). Network experimentation at scale. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 3106–3116.

Karwa, V. and Airoldi, E. M. (2018). A systematic investigation of classical causal

inference strategies under mis-specification due to network interference. *arXiv preprint arXiv:1810.08259*.

Kiuru, N., Burk, W., Laursen, B., Salmela-Aro, K., and Nurmi, J. (2010). Pressure to drink but not to smoke: Disentangling selection and socialization in adolescent peer networks and peer groups. *Journal of adolescense*, 33(6):801–812.

Kohavi, R., Deng, A., Frasca, B., Walker, T., Xu, Y., and Pohlmann, N. (2013). Online controlled experiments at large scale. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1168–1176.

Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM review*, 51(3):455–500.

Krivelevich, M. and Sudakov, B. (2003). The largest eigenvalue of sparse random graphs. *Combinatorics, Probability and Computing*, 12(1):61–72.

Krivitsky, P. N., Handcock, M. S., Raftery, A. E., and Hoff, P. D. (2009). Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models. *Social networks*, 31(3):204–213.

Krzakala, F., Moore, C., Mossel, E., Neeman, J., Sly, A., Zdeborová, L., and Zhang, P. (2013). Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, 110(52):20935–20940.

Lancichinetti, A. and Fortunato, S. (2009). Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117.

Lelarge, M. and Miolane, L. (2019). Fundamental limits of symmetric low-rank matrix estimation. *Probability Theory and Related Fields*, 173(3):859–929.

Leskovec, J., Kleinberg, J., and Faloutsos, C. (2007). Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data*, 1(1).

Livan, G., Novaes, M., and Vivo, P. (2018). *Introduction to random matrices: theory and practice*, volume 26. Springer.

Manski, C. F. (1995). *Identification problems in the social sciences*. Harvard University Press.

Massoulié, L. (2014). Community detection thresholds and the weak ramanujan property. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 694–703.

Mathews, H., Mayya, V., Volfovsky, A., and Reeves, G. (2019). Gaussian mixture models for stochastic block models with non-vanishing noise. In *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 699–703. IEEE.

Mathews, H. and Volfovsky, A. (2021). Latent community adaptive network regression. *arXiv preprint arXiv:2112.06097*.

Mayer, A. and Puller, S. L. (2008). The old boy (and girl) network: Social network formation on university campuses. *Journal of public economics*, 92(1-2):329–347.

Mayya, V. and Reeves, G. (2019). Mutual information in community detection with covariate information and correlated networks. In *57th Annual Allerton Conference on Communication, Control, and Computing*.

McPherson, M., Smith-Lovin, L., and Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1):415–444.

Mele, A., Hao, L., Cape, J., and Priebe, C. (2019). Spectral inference for large stochastic blockmodel with nodal covariates.

Miolane, L. (2019). Phase transitions in spiked matrix estimation: information-theoretic analysis.

Moody, J., Brynildsen, W., Osgood, D., Feinberg, M., and S., G. (2011). Popularity trajectories and substance use in early adolescence. social networks. *Social Networks*, 33(2).

Mossel, E., Neeman, J., and Sly, A. (2018). A proof of the block model threshold conjecture. *Combinatorica*, 38(3):665–708.

Nadakuditi, R. R. and Newman, M. E. (2012). Graph spectra and the detectability of community structure in networks. *Physical review letters*, 108(18):188701.

Nandy, P., Basu, K., Chatterjee, S., and Tu, Y. (2020). A/b testing in dense large-scale networks: design and inference. *Advances in Neural Information Processing Systems*, 33.

Newman, M. E. (2004). Detecting community structure in networks. *The European physical journal B*, 38(2):321–330.

Newman, M. E. (2006a). Finding community structure in networks using the eigenvectors of matrices. *Physical review E*, 74(3):036104.

Newman, M. E. (2006b). Modularity and community structure in networks. *Proceedings of the national academy of sciences*, 103(23):8577–8582.

Nowicki, K. and Snijders, T. A. B. (2001). Estimation and prediction for stochastic blockstructures. *Journal of the American statistical association*, 96(455):1077–1087.

Ogburn, E. L., Sofrygin, O., Diaz, I., and Van Der Laan, M. J. (2017). Causal inference for social network data. *arXiv preprint arXiv:1705.08527*.

Paluck, E. L., Shepherd, H., and Aronow, P. M. (2016). Changing climates of conflict: A social network experiment in 56 schools. *Proceedings of the National Academy of Sciences*, 113(3):566–571.

Paluck, E. L., Shepherd, H. R., and Aronow, P. (2020). Changing climates of conflict: A social network experiment in 56 schools, new jersey, 2012-2013.

Peixoto, T. P. (2014). Efficient monte carlo and greedy heuristic for the inference of stochastic block models. *Physical Review E*, 89(1):012804.

Qin, T. and Rohe, K. (2013). Regularized spectral clustering under the degree-corrected stochastic blockmodel. *arXiv preprint arXiv:1309.4111*.

Reeves, G., Mayya, V., and Volfovsky, A. (2019). The geometry of community detection via the mmse matrix. *2019 IEEE International Symposium on Information Theory (ISIT)*.

Reeves, G., Pfister, H. D., and Dytso, A. (2018). Mutual information as a function of matrix snr for linear gaussian channels. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 1754–1758. IEEE.

Rienties, B. and Nolan, E.-M. (2014). Understanding friendship and learning networks of international and host students using longitudinal social network analysis. *International Journal of Intercultural Relations*, 41:165–180.

Rohe, K., Chatterjee, S., and Yu, B. (2011). Spectral clustering and the high -dimensional stochastic blockmodel. *The Annals of Statistics*, 39(4):1878–1915.

Rubin, D. B. (1990). Formal mode of statistical inference for causal effects. *Journal of Statistical Planning and Inference*, 25(3):279–292.

Särndal, C.-E., Swensson, B., and Wretman, J. (2003). *Model assisted survey sampling*. Springer Science & Business Media.

Sävje, F. (2021). Causal inference with misspecified exposure mappings. *arXiv preprint arXiv:2103.06471*.

Sävje, F., Aronow, P. M., and Hudgens, M. G. (2017). Average treatment effects in the presence of unknown interference. *arXiv preprint arXiv:1711.06399*.

Sentse, M., Kiuru, N., Veenstra, R., and Salmivalli, C. (2014). A social network

approach to the interplay between adolescents' bullying and likeability over time. *Journal of youth and adolescence*, 43(9):1409–1420.

Shalizi, C. R. and Thomas, A. C. (2011). Homophily and contagion are generically confounded in observational social network studies. *Sociological methods & research*, 40(2):211–239.

Shen, L., Amini, A., Josephs, N., and Lin, L. (2022). Bayesian community detection for networks with covariates. *arXiv preprint arXiv:2203.02090*.

Shrum, W., Cheek Jr., N., and Hunter, S. (1988). Friendship in school: gender and racial homophily. *Sociology of Education*, 61:227–239.

Staber, U. (1993). Friends, acquaintances, strangers: gender differences in the structure of enterpreneurial networks. *Journal of Small Business and Entrepreneurship*, 11:73–82.

Stadtfeld, C., Vörös, A., Elmer, T., Boda, Z., and Raabe, I. J. (2019). Integration in emerging social networks explains academic failure and success. *Proceedings of the National Academy of Sciences*, 116(3):792–797.

Stephens, M. (2000). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(4):795–809.

Sussman, D. L. and Airoldi, E. M. (2017). Elements of estimation theory for causal effects in the presence of network interference. *arXiv preprint arXiv:1702.03578*.

Suwan, S., Lee, D., Tang, R., and et al (2016). Empirical bayes estimation for the stochastic blockmodel. *Electronic Journal of Statistics*, 10:761–782.

Toulis, P. and Kao, E. (2013). Estimation of causal peer influence effects. In *International conference on machine learning*, pages 1489–1497. PMLR.

Ugander, J., Karrer, B., Backstrom, L., and Kleinberg, J. (2013). Graph cluster randomization: Network exposure to multiple universes. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 329–337.

Ugander, J. and Yin, H. (2020). Randomized graph cluster randomization. *arXiv preprint arXiv:2009.02297*.

van Buuren, S. and Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in r. *Journal of Statistical Software*, 45(3):1–67.

Xin, L., Zhu, M., and Chipman, H. (2017). A continuous-time stochastic block model for basketball networks. *The Annals of Applied Statistics*, 11(2):553–597.

Xu, Y., Chen, N., Fernandez, A., Sinno, O., and Bhasin, A. (2015). From infrastructure to culture: A/b testing challenges in large scale social networks. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2227–2236.

Yan, B. and Sarkar, P. (2021). Covariate regularized community detection in sparse graphs. *Journal of the American Statistical Association*, 116(534):734–745.

Yin, H., Benson, A., Leskovec, J., and Gleich, D. (2017). "local higher-order graph clustering. In *23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Zhang, P. (2016). Robust spectral detection of global structures in the data by learning a regularization. In *Advances in Neural Information Processing Systems*, pages 541–549.

Zhang, Y. and Rohe, K. (2018). Understanding regularized spectral clustering via graph conductance. *arXiv preprint arXiv:1806.01468*.

Zhou, Y., Liu, Y., Li, P., and Hu, F. (2020). Cluster-adaptive network a/b testing: From randomization to estimation. *arXiv preprint arXiv:2008.08648*.

Biography

Heather Mathews completed her BS degree in mathematics and statistics with a co-major in predictive analytics from Miami University in May 2017. Heather started the Statistical Science PhD program at Duke University in August of 2017. During her time at Duke, Heather was advised by Alexander Volfovsky and co-advised by Galen Reeves. She spent the summer of 2021 as a data scientist intern at Netflix and will be starting her career as a data scientist at Spotify after graduation.