Concerning Some Statistical Problems on Graphs

By

DMITRY I. SHEMETOV
DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Applied Mathematics

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

James Sharpnack, Chair

Naoki Saito

Miles Lopes

Committee in Charge

2021

This dissertation is dedicated to my friends.

Contents

Concerning Some Statistical Problems on Graphs

Abstract

In this dissertation, we consider three statistical problems unified by an underlying graph structure.

The first problem concerns graph scan statistics. Scan statistics can be used for early detection of disease outbreaks by testing large regions for faint but detectable abnormal measurements. Traditional scan statistic implementations, such as the SaTScan software, fuse neighboring measurements to obtain more powerful global null tests. In this project, we provide an algorithm for detecting and localizing elevated levels in graph data. We present methods and software that can utilize both spatial proximity and network information. We apply this methodology to an outbreak of porcine reproductive and respiratory syndrome (PRRS) in pig farms and find hotspots of the disease. Our method can provably control the family-wise error rate and control the false discovery rate while detecting and localizing outbreaks.

The second problem concerns distributed communication with memory constraints. Memory constraints are common in modern statistical systems like those occurring in wireless sensor networks. We consider a model system of sensors communicating in a tree structure under bit-rate constraints to solve a hypothesis testing problem. We obtain bounds on the hypothesis testing error rate for a central class of decision rules as functions of the bitrate, tree width, and depth. These bounds give fundamental tradeoffs on memory and communication in wireless sensor networks.

The final problem concerns random graph model fitting. Exponential random graph models (ERGMs) are a family of random graphs often used in social science due to their elegance of definition and flexibility of expression. Fitting the parameters of such models to data is quite challenging however, due to the complexity of the partition function and the need to count subgraphs. Traditional Markov chain Monte Carlo methods used for fitting suffer from poor initialization errors and the need to produce many graph samples. We study a pseudolikelihood based on the large deviations theory for ERGMs developed by Chatterjee and Diaconis and investigate its performance under noisy subgraph count estimates. The work in this project shows preliminary results that our proposed method may be viable in low edge density graph regimes.

Acknowledgments

This dissertation covers three major projects I worked on during my time at UC Davis. I am glad to have had the opportunity to sharpen my mind and character closer to the qualities of a researcher. I have also been fortunate to meet a number of excellent people over these years. I owe a lot to these folks, many of whom I am lucky to call friends. I am grateful to Dr. James Sharpnack for helping me get back on track academically and professionally; his encouragement was crucial to my success. A sincere thanks to Dr. Naoki Saito and Dr. Miles Lopes for reviewing my dissertation and pushing me to work harder. I am thankful for my long-time housemates at the "Anderson House": Jordan Snyder, Robert Bassett, David Weber, Will Wright, and Kiril Paramonov (Hon.); I learned so much from each of you. A special thanks to my dear friend Jenette Sellin for the music, the egg burritos, and the endless kindness. Eugene Shvarts, Sam Fleischer, and Alaa Mousawi (and the LANL Machine Learning Summer School) – thanks for the fond memories. A big thank you to my mother Victoria for always encouraging me in that tough Russian way. And of course, I could not have done without my second family at the Bird Path Zen sangha; thank you all for being a bastion of sanity. Apologies to anyone I missed here.

If I may indulge for a paragraph of reflections on my experience in graduate school. My research trajectory was, perhaps, unusual in the length of time it took me to settle on a topic. I began by studying Hidden Markov models and information theory with a group of physicists. I remember fondly the company of many students and postdocs there, staying late into the night philosophizing about the nature of computation and information. While I spent years on that work, it does not appear here; the philosophy never really touched firm mathematical ground, so to speak. By far one of the biggest challenges I found in research was evaluating what was personally interesting and what was publishable. It turns out that both are important: without being truly invested you will burn out and without publishing you have no currency in the trade. In a perfect world these two would be identical, but I often found this was not so. I have also found it easy to be confused about what will be rewarding and what simply sounds good on arXiv; the most valuable skills I have learned in graduate school help reduce that confusion. While the academic life promises intellectual freedom, it is full of soft constraints, ever-changing, and dependent on funding fortunes. It takes both self-knowledge and knowledge of the waters one swims in to navigate skillfully. For the first, I

have personally found meditation and the arts to be helpful. For the second, see [**41**] for intuition pumps on the subtleties of the bureaucracy. Hopefully this will help someone be less clueless than I was when entering graduate school. I wish anyone following in this path luck and grit. See you on the other side.

The work in Chapter 3 was possible thanks to the support of the NSF BIGDATA:AI project #1838207.

CHAPTER 1

Introduction

Graphs are becoming one of the most widely-studied data structures. It is oft-remarked that we are now in the Information Age, with the term referring to the computerization of the lives of people living in the colloquial West, in which vast stores of data are now being harvested by governments, companies, and individuals in an ever-increasing push to be "data-driven" [**25**]. The empirical mindset, while always limited by the nobility of its motivations, the quality of its data, and the thoroughness of its analysis, nonetheless presents an uncontroversial advantage in many concrete real-world tasks such as building fault-tolerant communication systems [**97**], playing Go [**109**], and, arguably not long from now, driving cars autonomously [**7**]. Leaving aside the natural and non-trivial question of whether these are desirable goals, let us focus on the shapes of data in question. With substantial study dedicated to regular structures such as lattices, the rise in data acquisition through the breakthroughs of computers have opened the path for widely heterogenous structures, encompassed elegantly by the graph (or network). Graphs naturally arise in applied and theoretical fields, ranging from chemistry, physics, and biology to public health and social science. Whether the scientists are interested in studying chemical reactions, molecular structure, solid-state matter structure or trying to understand social networks for public health, security, or anthropology, graphs are central structures of study. The flexibility of the graph allows scientists to represent phenomena under study more accurately than they would otherwise. As scientific work often relies on encoding relational information between entities and said relationships between entities naturally produce graphs, it is not surprising that graphs play a central structural role; in a way, they are simply our label for relational information encoding. Data on graphs is now ubiquitous: we have Internet search engines ranking websites by analyzing data on the websites while keeping the sites' linking structure in view, while Netflix recommendation system challenge asking for recommendations based on a user's viewing history and their similarity to other viewers [**12**]. As such data becomes relevant in more industries, graph databases now join relational databases as an indispensable tool in data

management [**100**]. With all the data available on people's behavior through the internet, with A/B testing social science can now be conducted on a previously untenable scale [**25**][1].

The graph concept was not formally introduced until the 18th century mathematician Leonard Euler's work on the famous Königsberg bridge problem. Unlike the mathematical branches addressing motion, calculation, and measurement, which were motivated by scientific and technical pursuits, graph theory's origins can often be traced to questions that were little more than curious puzzles. For example, the problem of a Knight's Tour (also solved by Euler in 1759) asks a player to visit every square of a standard chess board exactly once while limited to making only a knight's moves. Another was 'The Icosian Game' invented by Sir William Rowan Hamilton, which required players to take paths on the graph of a dodecahedron. 'The Icosian Game' was naturally tied to Sir Hamilton's better discovery, the quarternions, by being an instance of a non-commutative algebraic system of roots of unity, which Sir Hamilton labeled 'The Icosian Calculus' (modern algebraic terminology would refer to this system as a group presentation of the icosahedral rotation group). Despite these humble origins, connections to deeper mathematics emerged. Euler's Polyhedral Formula had already anticipated the discovery of future topological invariants. As chemical theory was in rapid development in the 19th century, Alexander Crum Brown's graph notation (essentially the one used today) was introduced in 1864. The first textbook on graph theory was published in 1936 by König.[2] With the development of computer science, graphs were integrated into core data structures and algorithms, and were part of key problems in complexity theory such as the traveling salesman problem. Even more uses for graphs can be found in the relatively recent field of network science, which has seen many applications in fields such as social science, economics, and physics [**82**].

With this context in mind, we turn to the work in the present dissertation. This dissertation focuses on three separate research projects with graphs serving as an underlying theme. Despite this unity, it should be noted that the three projects work are quite different, with each project embedded in a substantially different research context. For this reason, each ensuing project chapter presents its own literature and develops its own notation (with minor overlap). Unless otherwise

[1]This coming at a time of concerning rates of scientific reproducibility [**9**, **33**].
[2]Those interested in the historical progression of ideas in graph theory can refer to [**16**] for a translated, corrected, and annotated collection of historically important papers.

specified, the work of each chapter is in preparation for submission to a journal. Let us take a high-level view of the work.

1.1. Signal detection in communication networks

The first project concerns *signal detection in communication networks*. Graphs provide a natural way to describe communication networks, where vertices are computing nodes and edges are communication channels. As modern day communication networks contain many devices exchanging messages in a wide variety of topologies, so has research turned to understanding these structures. One of the most common topologies is called the *fusion center*, which is effectively a star graph with a single root node that receives information from the leaf nodes. It is the root node's job to combine the messages sent to it in order to solve a statistical problem, such as signal detection. Bandwidth limitations in the communication channel lead to a loss of information, prompting much academic work to study the fundamental limitations of signal detection in such a constrained fusion center setting [**39**, **84**, **116**, **118**, **119**, **128**]. Other communication schemes assume more general graph structures, in which every node exchanges messages until a group consensus emerges [**8**, **20**, **68**, **121**]. *Hierarchical structures* are of particular interest because they relieve the designer from the restrictions of the fusion center setting, allow for smaller computations at the root node, and are simple enough to study with analytic methods [**30**].

We study the fundamental limits of simple hypothesis testing with a tree communication structure. Moreso, we consider bitrate constraints on the messages passed between the underlying nodes, and for ease of exposition, we consider channels that only pass a single bit. The trees are assumed to be perfect to simplify the analysis, though our results can be used as approximation for less regular trees.[3] We characterize the information loss due to the hierarchical communication scheme as a function of the degree of the tree. On one extreme, the binary tree leads to an almost complete loss of information, with no gain due to increasing the number of sensors. As the degree, m, of the m-ary tree increases, more of the information at the sensor nodes is preserved as the data is passed up the communication tree. We obtain bounds on the error rates for all m-ary trees for a central class of communication functions (likelihood ratio quantizers), which we conjecture are

[3]A perfect m-ary tree is one in which every node has m children and all leaves have the same depth.

asymptotically optimal. In our bounds, the bias-sample rates are subparametric for m-ary trees, which recovers the parametric rate as m goes to infinity.

1.2. Scan statistics in spatio-temporal graphs

The next project considers *scan statistics in spatio-temporal graphs* in the context of disease spread. Scan statistics, broadly speaking, attempt to solve the problem of finding clusters of abnormal activity, given data distributed in space and/or time. Traditionally, scan statistics use a small library of regularly shaped regions to scan on regular domains such as lattices and $\mathbb{R}^3$ have received plenty of study in the 20th century, with many data distributions (e.g., Gaussian, Poisson, Binomial) considered [**48**, **64**, **67**, **73**, **137**]. This work produced the well-used software SATScan, among others [**70**].

In this project, we use the structural information in a graph to provide information on other variables of interest taking place on the graph. A number of graph statistics methods have been proposed, some with the goal of attaining computational feasibility [**91**, **112**]. Theoretical results in this area have only recently received attention [**5**, **104**]. Our graphs encode spatio-temporal information between disease-susceptible sites along with relational information that poses risk of disease transmission. We develop and study a graph scan statistic based on shortest path length neighborhoods that is able to detect and localize disease hotspots on a network structure, while controlling standard measures of error. We apply this method to a proprietary industry dataset detailing the prevalence and spread of Porcine Reproductive and Respiratory Syndrome (PRRS) on farms in the US. Using the pig shipment information between these farms, our statistic was naturally able to mark high-degree hotspots that acted shipment hubs, while simultaneously detecting low-degree nodes that had elevated counts for their region, suggesting the presence of transmission vectors external to the information captured in our network. Parts of this work were presented at the GeoVet 2019 Conference [**105**].

1.3. Inferring exponential random graph models

Our final project considers a new approach to *scalable inference of exponential random graph models* (ERGMs) using a graph limit pseudolikelihood with approximate subgraph counts. ERGMs are a widely used random graph model to model network phenomena in social science [**99**]. One of

their advantages is that they can flexibly promote or reduce the presence of structural features in the sampled graphs by tuning their structure function and the associated parameters. The observation that transitivity measures such as the clustering coefficient (which measure the prevalence of triangle graphlets) are elevated in real-world networks in comparison to random graph models [**130**] has brought attention to the study of more general local features of the graph. For these reason, the structural features in ERGMs are often taken to be subgraphs such as wedges or triangles.

Fitting random graph models serves a number of uses: for graph similarity computation, generation of similar graphs, proving the statistical likelihood of structural properties; an entrance to the literature of modeling network data can be found in [**42**, **98**]. Fitting ERGM parameters, while typically done with Monte Carlo approaches [**110**], still suffers from three main difficulties, two computational and one statistical: evaluating the partition function is intractable; counting subgraphs is slow for large graphs;[4] and different model parameters can lead to essentially identical graph distributions. We study a method that relies on two key pieces to overcome these challenges. First, we approximate the partition function by relying on a pseudolikelihood based on large deviations theory in the work of Chatterjee and Diaconis [**26**]. Second, we study the performance of this fitting method under approximate subgraph counts obtained by a Monte Carlo counting scheme. This counting scheme is based on the literature of efficient graphlet (induced subgraphs) counting methods [**2**, **23**, **86**].[5] The present author's work in [**86**] developed a Monte Carlo approach to estimating the graphlet counts which can be leveraged to trade compute time for accuracy in counting general subgraphs. We simulate noised graphlet counts and compare the pseudolikelihood inference quality under the noise, finding certain regions of robustness. The results give us preliminary evidence that a Monte Carlo subgraph estimation method combined with the pseudolikelihood method could be feasible.

[4]Subgraph counting, while polynomial in the size of the graph, is exponential in the size of the subgraph and hits practical limitations for large graphs (millions of nodes) of interest, such as in social network graphs.
[5]Graphlets received plenty of attention for their applications in biological network comparison [**92**, **93**].

CHAPTER 2

Compressed Statistics in Hierarchies

2.1. Introduction

In recent years, there has been a considerable amount of work extending statistical estimation to the setting of distributed computation [**21**, **36**, **101**]. This interest is fueled in part by the increased need to solve distributed statistical problems in fields such as wireless sensor networks (WSNs) and machine learning (ML). In WSNs, for example, one of the challenges is to use many redundant communication-constrained sensors to collect, communicate, and process data, in an efficient manner, for the purpose of solving a wide variety of problems, ranging from environmental to industrial monitoring (i.e. forest fire detection or pipeline health monitoring) [**31**]. In the field of machine learning, growing industry attention has led to increased demand for parallel computer architecture and algorithms [**76**,**131**,**138**] to maintain computational cost scaling. Parallel architectures, however, come with communication constraints which can reduce the computational gains expected of the algorithms running on them.

Both these areas share the need to communicate and process data for the purpose of solving a statistical problem, while being constrained by communication topologies and communication bandwidth. In WSNs, in order to communicate data through the network, some methods use a spanning tree architecture [**135**], structuring their nodes in hierarchy. The problem then is to communicate the collected sensor data up the tree and maintain statistical accuracy, while compressing to meet the bandwidth constraints. In ML, the communication constraints commonly arise from two factors: data privacy concerns and memory constraints on the worker nodes. The need to compress or sub-select the data must then be balanced with the need to maintain statistical accuracy. Therefore, to aid the development of these algorithms, it is important to study the trade-offs in a systematic way.

While a large set of work has addressed the effects of communication constraints the fusion center setting, the hierarchical setting has not been fully developed yet in the literature. Hence, in this chapter, we provide an analysis of hypothesis testing in the tree setting.

In this work, we consider a decentralized detection problem involving a simple-simple hypothesis test while under bandwidth constraints and in a hierarchical communication setting. We characterize the asymptotics of the error probability of hypothesis testing in this setting for a particularly important class of decision strategies. We derive bounds on the error probability as functions of the number of samples, width and depth of the tree, and bitrate. But before we get into the details, let us sketch the theoretical context of our work.

2.2. Background

2.2.1. Related Work. There is substantial prior work on related problems. One related field is that of decentralized detection, which received a lot of attention a few decades ago. One of the central concerns in this field is the problem of using multiple sensors to solve a statistical problem by combining their data through some form of message passing; generally the sensor topology is assumed to be a fusion center setup. A review of some relevant results from work in this field can be found in [**119**]; they use a minimum cost functional framework and describe a number of conditions on the optimality of decision strategies for hypothesis testing in the fusion center and the hierarchy setting (while their analysis does touch on hierarchical structures, they focus on the fusion center scaling with the number of sensors and do not provide results dependent on depth). For example, using identical sensors performing a likelihood ratio test for binary hypothesis testing in the fusion center is asymptotically optimal [**118**]. More recent work on the problem considered the detection of a known vector from compressed measurements under Gaussian noise [**39**]; other work has focused on the effects of quantization [**128**] and on the effects of memory on sequential detection [**79**]. Another major avenue of research in distributed information processing has focused on the diffusion of consensus [**8**, **20**, **121**]; a review of other routing schemes for WSNs can be found here [**30**]. The study of the computational complexity of select decentralized decision problems can be found in [**117**].

Fundamental lower founds for problems other than parameter estimation and hypothesis testing have been considered in the fusion center as well, such as the hide-and-seek problem, which requires one to detect a single elevated mean coordinate from a vector of measurements at each sensor [**101**]. Lower bounds in this area often rely on information theoretic or metric entropy arguments;[1] recent work in similar vein can be found here [**37**, **139**]. While there is a rich literature surrounding two related fields: strong data processing inequalities [**95**] and the contraction properties of stochastic operators [**32**], the theories contained therein focus on maximization of contraction coefficients over input/output distributions; our work, on the other hand, requires the maximization of contraction coefficients over both channels and input distributions.

2.2.2. The Centralized, Fusion-center, and Hierarchical Settings. In this section, we will sketch some context for decentralized detection in various settings. To this end, let us define the problem and our error metric. The basic problem of decentralized detection is to do a hypothesis test between

$$\begin{aligned} H_0 &\sim \text{Bern}\left(p_0 = \frac{1}{2} - \delta\right) \\ H_1 &\sim \text{Bern}\left(p_1 = \frac{1}{2} + \delta\right), \end{aligned} \tag{2.1}$$

where $\text{Bern}(p)$ is a Bernoulli random variable with probability p, with n samples from one of the two hypotheses. Let p_e denote the probability of error of a test T defined as

$$p_e \coloneqq \frac{1}{2}(P_0(T=1) + P_1(T=0)),$$

where $P_i(x) = \Pr(x|H_i)$ and we have made the Bayesian assumption $\Pr(H_0) = \Pr(H_1) = 1/2$.

In the centralized setting, given n samples from one of two hypotheses we have the well-known error bound[2]

$$p_e \sim \Theta(\exp(-n\delta^2)),$$

[1]Fundamental tools here are Fano's inequality, the data processing inequality [**34**], and metric entropy [**132**].
[2]Here we use Big Theta notation $f(n) \sim \Theta(g(n))$ as shorthand for

$$\exists k_1, \exists k_2, \exists n_0, \forall n > n_0 : \; k_1 \cdot g(n) \leq f(n) \leq k_2 \cdot g(n).$$

which is attained by the maximum likelihood threshold test and can be derived through standard methods like Hoeffding's inequality and standard nonparametric lower bounds [**120**]. The $\delta \sim n^{-1/2}$ relation is what is known as the *parametric sample-bias tradeoff*.

In the fusion center setting, a similar bound can be derived as a function of the number of sensors. Recall that the fusion center setting involves a root node receiving information from m sensors in what is effectively a star topology. To make the problem interesting, each sensor should receive more data than they can send through their limited bitrate channel, so let us suppose we have $n = m \cdot k$ samples from the distribution, with k samples per node, and m sensors with a 1-bit channel. It is asymptotically optimal (in the number of sensors), for the sensors to use a likelihood ratio test with the same threshold [**118**]. The exact sample-bias tradeoff is more difficult to capture in this case, but can be approximated to first-order

$$p_e \leq O(\exp(-\beta \frac{n}{k} \delta^k)),$$

where β depends on k, δ, m (this can be seen by reducing the problem to the centralized case with with $\frac{n}{k}$ samples and updating the Bernoulli parameter to the probability of 1 in a likelihood ratio test on k samples).

Now let's turn our attention to the same problem but in the hierarchical setting. The fully general hierarchical setting assumes the nodes are arranged in an arbitrary tree structure. Analysis of this case is very challenging. Let us make the regularity assumption that the tree structure is a perfect m-ary tree.[3] Binary data from the Equation (2.2) comes in at the leaf nodes and is passed up to the root node which makes a determination on distribution of the data based on the messages it receives. Each sensor node in the tree uses a single-bit decision rule. Let us take the simplest case of this setting possible: the binary tree, i.e., $m = 2$. It can be seen that it is impossible to achieve any asymptotic scaling in this setting at all. To do this, let us follow the data as it moves up the layers. The first set of nodes receive their single bit of data and are able to pass it through their single bitrate channel without a problem. The next set of nodes however have the task of summarizing two bits of data with a single bit. A natural choice of decision rule is the following "voting" strategy: if both

[3]As previously mentioned, a perfect m-ary tree is one where each node has either m children or no children and all the leaf nodes are of the same depth.

inputs are 0 output a 0, if both inputs are 1 output a 1, and break ties with a fair coin flip. Suppose all the sensor nodes use this rule. We claim that regardless of the depth of the tree, the root node will effectively have access to just two data samples from which it must make its hypothesis determination. To convince yourself of this, let us note the output probability of a 1 for a sensor node that uses the strategy laid out above. Denoting by p the Bernoulli parameter at the input nodes, this probability is $\Pr(\text{Output 1}) = \frac{1}{2}(\Pr(\text{Input 01}) + \Pr(\text{Input 10})) + \Pr(\text{Input 11}) = \frac{1}{2}((1-p)p + p(1-p)) + p^2 = p$. The probability being p tells us that the Bernoulli variable from the children nodes to the parent nodes has not changed. Because of this, the parent at the next level will have the same output probability and so will all the nodes all the way up to the two nodes sending information to the root node. Thus, the root node effectively has access to just two samples from a Bernoulli distribution with parameter p. (For another angle, consider that the "always send left" channel (i.e., $(x, y) \mapsto x$) has the same output probability as the voting strategy above.) An exhaustive check of the rest of the decision rules (of which there are just 16, with significant symmetry), shows that this cannot be improved.

Fortunately, for $m > 2$ the story changes. It is one of our main results to quantify the dependence of the probability of error on the width of the tree m, the depth of the tree D, and the separation between the hypotheses δ. We will see that the constraints imposed by the compression and the hierarchical communication lead to error bounds with a qualitatively different character from the centralized case and from the fusion center case.

2.2.3. Binary distributed detection. In this section, we will consider the practical step of quantization that occurs prior to the framing used in our problem. Namely, since the sensor nodes we use receive binary data, we consider the information loss that would occur if binary quantization was applied to real-valued data prior to hypothesis testing. This loss of information can be captured in the decrease in Chernoff information, as shown below.

Consider a simple vs. simple hypothesis testing framework where there are n independent identically distributed (iid) samples, $X_1, \ldots, X_n$, from either a common null distribution P_0 or common alternative distribution P_1 (with P_1 absolutely continuous with respect to P_0). For a proposed rejection region R_n, the size is $\alpha_n := P_0^n(R_n)$ and the power is $\beta_n := P_1^n(R_n)$. When there are no communication constraints, then the Neyman-Pearson lemma states that the likelihood

ratio test (with possible randomization) achieves the optimal power for a given size. The Chernoff information [**29**] is defined as

$$C(P_0, P_1) \coloneqq -\min_{0 \le \lambda \le 1} \log \left(\int \left(\frac{dP_1}{dP_0} \right)^{\lambda} dP_0 \right).$$

Then in the centralized case, in which one computer has access to all of the data, it is well known that there is a test such that

$$\alpha_n + \beta_n \le 2 \exp\left(-nC(P_0, P_1)\right),$$

and the error exponent is optimal (see Theorem 11.9.1 in [**34**]).

In the distributed detection setting [**123**], there are n sensors, each observing an X_i and they pass a message $\gamma_i(X_i)$ from a limited alphabet to a fusion center. We will consider binary messages where the message can be a single bit, $0, 1$, and the fusion center will decode the correct hypothesis from the messages $\{\gamma_i(X_i)\}_{i=1}^n$. It was shown that identical $\gamma_i = \gamma$ are asymptotically optimal [**118**], so the resulting error exponent at the fusion center is the Chernoff information $C(\gamma(P_0), \gamma(P_1))$ (where $\gamma(P_j)$ is the distribution of $\gamma(X_1)$ under $H_j, j = 0, 1$). The Chernoff information between the induced Bernoulli distributions will not be larger than the original Chernoff information, and so the quantization may lead to a loss of information. Furthermore, the optimal γ, which maximizes the Chernoff information, is a likelihood ratio quantizer (a fact that dates back to [**102**]). Finding the threshold for the likelihood ratio quantizer that maximizes the Chernoff information can be solved with a convex optimization.

Given a prior distribution on the hypotheses, the maximum a posteriori (MAP) estimator of the hypothesis from an observation X_i is also a likelihood ratio quantizer with a simple threshold. Define the resulting Bernoulli successes after quantization,

$$p_j = \mathbb{P}\{\gamma(X_1) = 1 | H_j\}, \quad j = 0, 1.$$

Let $p^* = (p_0 + p_1)/2$ and $\delta = p_1 - p^* = p^* - p_0$. For the MAP estimator with a uniform prior ($\mathbb{P}(H_0) = \mathbb{P}(H_1) = 1/2$), the resulting Bernoulli distributions are related to the total variation

divergence

$$V(P_0, P_1) := \sum_{x \in \Omega} |P_0(x) - P_1(x)|$$

in the following way. We have that $p_0 \leq 1/2 \leq p_1$ and

$$1 - 2\delta = p_0 + 1 - p_1 = 1 - V(P_0, P_1).$$

Not only is $2\delta = V(P_0, P_1)$ for the MAP estimator, but also if we look at the resulting Bernoulli distributions then $V(\gamma(P_0), \gamma(P_1)) = 2\delta$.

In general, the Chernoff information between the two Bernoulli distributions is a complicated closed-form expression of their success probabilities. This expression greatly simplifies in the symmetric case where $p^* = 1/2$, in which case

$$C(\gamma(P_0), \gamma(P_1)) = -\log\left(2\sqrt{\left(\frac{1}{2} - \delta\right)\left(\frac{1}{2} + \delta\right)}\right)$$

(see Appendix A.2) For two hypotheses P_0, P_1, any quantizer will have Chernoff information less than the above expression, meaning that the $p^* = 1/2$ case maximizes the information retained for a given total variation. To ease our exposition, we will often focus on this case. However, when possible, we will state our results for any binary quantizer, $\gamma(X_i)$, and their resulting Bernoulli distributions parametrized by p^* and δ.

In some cases, the MAP estimator with uniform prior is also optimal. As an example in which $p^* = 1/2$ is optimal, consider two univariate normal distributions with shifted means (we will denote a normal distribution with mean μ and variance σ by $N(\mu, \sigma)$). For $P_0 \sim N(-\mu, 1)$ and $P_1 \sim N(\mu, 1)$, then the Chernoff information $C(P_0, P_1) = \mu^2/2$ [**83**]. By symmetry, the quantizer with the largest resulting Chernoff information is also the MAP estimator for uniform prior. The total variation is $V(P_0, P_1) = 1 - 2\Phi(-\mu)$ and $p^* = 1/2$ (here Φ denotes the cumulative distribution function of the normal distribution). When $\mu = 1$ the TV divergence is approximately 0.683 and the Chernoff information is 0.5. The Chernoff information between the Bernoulli's induced by the MAP estimate is then 0.314, which indicates that there is a substantial loss of information due to the quantization at the sensors. This relative loss is exacerbated if we allow $\mu \to 0$, because $\delta = \mu/\sqrt{2\pi}$

and $C(\gamma(P_0), \gamma(P_1)) = 2\delta^2 + o(\delta^2) = \mu^2/\pi + o(\mu^2)$. Hence, asymptotically in the low SNR regime for the shifted normal means, the loss of information due to quantization is by a factor of $2/\pi$.

We will consider a hierarchical detection setting, in which the sensors are leaf nodes in a perfect m-ary tree, and binary messages must be passed up the hierarchy to a root node. The question that we will address is, what is the additional loss of information due to the hierarchical communication scheme?

2.2.4. Our Problem. The general problem we will consider is the following. Perform the hypothesis test

$$\begin{aligned} H_0 &\sim \text{Bern}\left(p_0 = p^* - \delta\right) \\ H_1 &\sim \text{Bern}\left(p_1 = p^* + \delta\right), \end{aligned} \tag{2.2}$$

given $n = m^D$ samples of the data fed in at the bottom sensors of a complete, depth D, m-ary tree and the sensor nodes use a b-bit decision rule. We will study the performance of various decision rules under the probability of error metric. We will particularly be interested in questions of decision rule optimality and the underlying sample-bias tradeoffs. Even though analytic results were only attainable in very restricted cases, we provide analyses of suggestive examples for the more general cases.

2.3. Our Contributions

Our analysis relies on a few key regularity assumptions. First, we assume that the nodes are formed in a perfect tree of a given depth and width. Second, we assume that all the sensor nodes use an identical decision rule. This assumption is realistic in practice because sensor algorithm design assumes identical sensors. Finally, we restrict our attention to the class of monotone threshold rules. Further justifications of these assumptions are provided below.

Consider the symmetric hypothesis case. Define the parameter map to be a binomial tail sum of the form

$$f(p; m, k) = \sum_{i=k}^{m} \binom{m}{i} p^i (1-p)^{m-i},$$

(the origin of this function will be explained in the proofs section). The *majority threshold rule* is defined to be $f(p; m, \frac{m+1}{2})$ if m is odd and $f(p; m, \frac{m}{2}) - \frac{1}{2}\binom{m}{m/2}p^{\frac{m}{2}}(1-p)^{\frac{m}{2}}$ if m is even.[4] If the tree's sensor nodes all use the identical, majority threshold rule, we prove the following lower and upper bounds for the probability of error.

THEOREM 2.3.1. *For a perfect m-ary tree of depth $D+1$ for the hypothesis testing problem Equation (2.2) with $p^* = 1/2$ and using the $b = 1$ majority threshold rule $f(p) = f(p; m, \lfloor \frac{m+1}{2} \rfloor)$, we have*

$$p_e \geq \frac{1}{2}\left(1 - 2\left(f'(1/2)\right)^D \delta\right),$$

To complement this result and understand optimality of this bound, we also derive the following upper bound on the majority voting channel in the same setting.

THEOREM 2.3.2. *With the same assumptions on the hierarchy structure as in Theorem 2.3.1 and using a decision rule with the parameter map $f(p) = f(p; m, k)$, we have that as $\delta \to 0$ then $p_e \leq \epsilon$ if*

$$D > \frac{-\log \delta}{\log\left(f'(1/2)\right)} + \log_k \left(\frac{\log\left(\binom{m}{k}\epsilon^{k-1}\right)}{\log\left(\binom{m}{k}\left(\frac{1}{4}\right)^{k-1}\right)} \right).$$

If m is fixed, then this quantity takes its maximum value for $k = \lfloor \frac{m+1}{2} \rfloor$.

To give some intuition for these results, let us consider some corollaries.

2.3.1. Corollaries. First, note that with δ small enough these bounds are tight up to the constant term introduced by ϵ. What we mean by this is that the relationship between D, δ, and $f'(1/2)$ is identical between the lower bound and the upper bound up to a constant depending on ϵ and m. The lower bound is focused on capturing the effects of escaping the unstable fixed point region (e.g., when $f'(1/2)^D\delta < 1/2$), where improvement is slow. The upper bound matches in this region, modulo a constant dependent on on ϵ. Outside that region, the lower bound loses its effectiveness, however, and the upper bound parts ways with the lower bound as $\epsilon \to 0$.

A closer inspection of parameter relationship in the bounds present interesting features. Namely, we can derive a sample-bias tradeoff from the quantity $f'(1/2)^D\delta$. To do this, first note that the

[4]The reason for the extra term in the case of even m is that we set the majority threshold rule to break ties with a random choice.

tree depth is related to the number of data samples available at the bottom of the tree $n = m^D$. This allows us to calculate the sample-bias tradeoff and compare to the standard tradeoffs seen in the centralized case and in the fusion center case. A little algebra yields that the tradeoff is $\delta \sim n^{\log_m f'(1/2)}$. Consider this for the first nontrivial case $m = 3$, which gives $\delta \sim n^{-\log_3 \frac{3}{2}} \approx n^{-0.37}$. This is notably less than the standard parametric tradeoff $\delta \sim n^{-1/2}$ seen in the centralized and fusion center cases.[5] Thus, we have shown that the sample rate for the best candidate decision rules is still fundamentally less than the sample rate in the centralized case. This points out that the constraints presented in hierarchical compressed communication introduce a qualitative change one's ability to solve statistical problems.

If we generalize the hypotheses to have an asymmetric bias (that is $p^* \neq 1/2$ in Equation (2.2)), the situation becomes much trickier. First, most hypotheses pairs will not be distinguishable by the finitely many deterministic threshold rules available to us, so stochastic rules will have to be employed. This requires one to find the correct mixture of deterministic channels so that the parameter map has the appropriate fixed point at p^* (see Figure 2.1 for an illustrative plot). While solutions to this problem can be found numerically, it leaves open a large number of degrees of freedom in the solution as m becomes large. It is unclear what mixture of monotone threshold rules yields the optimal probability of error rate for a given fixed point p^*. Second, the lower bound and the upper bound no longer match. This is because the location of the inflection point in the parameter map no longer matches with the fixed point. Nonetheless, the methods from the symmetric case can still be used to get a sense of the performance of those methods. Let's denote by p_i the inflection point of $f(p)$. Then Theorem 2.3.1 should be updated to include p_i instead of $1/2$, leaving us with $\delta \sim n^{-\log_m f'(p_i)}$ for the lower bound (Theorem 2.3.2 cannot be so easily adapted). While numerical results can also be derived as desired, analytic results become hard to describe in this case, since no closed form for p_i is available.

2.4. Proofs of Results

2.4.1. From Decision Rules to Parameter Maps. In this section, we provide proofs of our results. We first lay out some definitions and relevant conceptual handholds. The main object of

[5] As a sanity check, we can prove that $\log_m f'(1/2) \to \frac{1}{2}$ as m gets large, which recovers that centralized case. See Appendix A.4.

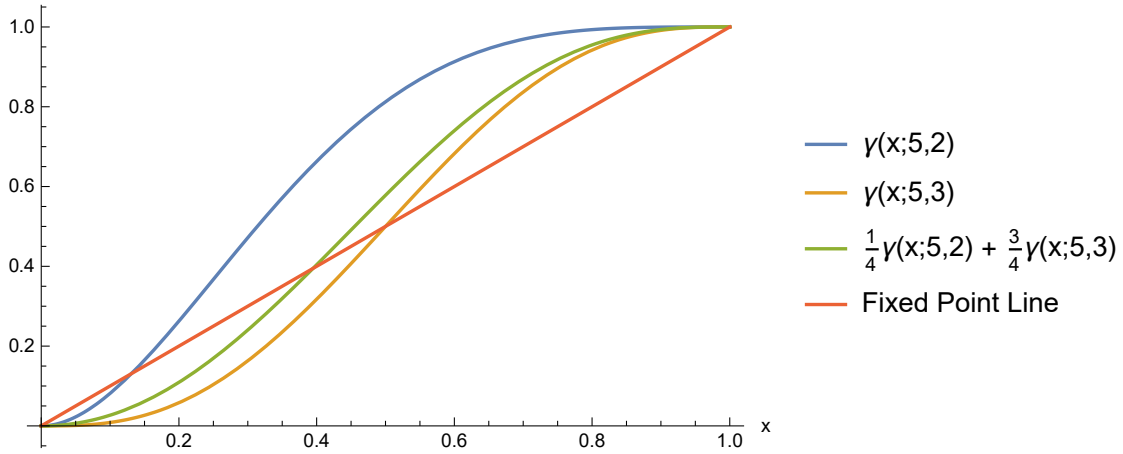

FIGURE 2.1. Here we see a few parameter maps (i.e., $\gamma(x;5,2)$). Intersections with the diagonal line correspond to fixed points of the parameter maps.

study in this section will be what we call parameter maps, which are maps associated with decision rules that govern the change in the data distribution parameter as it moves up the layers in the tree. Then we proceed to prove a series of lemmas needed for our results.

Let us first fix a perfect m-ary tree of depth D. A *decision rule* used by a sensor node in the tree is a (stochastic) function $\gamma : [2^b]^m \to [2^b]$, with bitrate b, input size m, and $[2^b] = \{0, 1, 2, \ldots, 2^b - 1\} \subset \mathbb{N}$. A *decision strategy* is the composite tree function made up of the decision rules of the constituent sensor nodes. An *identical decision strategy* is one where all the sensors use the same decision rule. We focus on the case $b = 1$ and reduce to the study of identical monotone threshold strategies. A *monotone threshold strategy* is a strategy where each sensor uses a monotone threshold rule, that is, a function γ of the form

$$\gamma(\boldsymbol{x}; m, k) \coloneqq \begin{cases} 1 & \sum_{i=1}^{m} x_i \geq k \\ 0 & \sum_{i=1}^{m} x_i < k \end{cases}$$

or a stochastic mixture of such thresholds. We will use the following weighted summation notation to denote stochastic mixtures of threshold rules

$$\sum_{i=0}^{m} \alpha_i \gamma(\boldsymbol{x}; m, i) \coloneqq \begin{cases} \gamma(\boldsymbol{x}; m, 0) & \text{with prob. } \alpha_0 \\ \gamma(\boldsymbol{x}; m, 1) & \text{with prob. } \alpha_1 \\ \vdots & \\ \gamma(\boldsymbol{x}; m, m) & \text{with prob. } \alpha_m \end{cases}.$$

The *majority threshold rule* is the decision rule $\gamma(\boldsymbol{x}; m, \lfloor \frac{m+1}{2} \rfloor)$. (For a more detailed look at these function classes see Appendix A.1.)

Now comes our key observation: the m children of any parent sensor present a set of iid Bernoulli random variables with some parameter p (since the inputs and decision rules below are all identical). The transformation of these variables into an output of 0 or 1 introduces a transformation of the Bernoulli parameter of the children into the Bernoulli parameter of the parent given by

$$f(p) \coloneqq \Pr(\gamma(\boldsymbol{x}) = 1) = \sum_{\boldsymbol{x} \in \gamma^{-1}(1)} p^{|\boldsymbol{x}|}(1-p)^{m-|\boldsymbol{x}|} \tag{2.3}$$

($\boldsymbol{x} \in \{0,1\}^m$, $|\boldsymbol{x}| = \sum_{i=1}^m x_i$ and γ is the given parent's decision rule). Such a transformation between Bernoulli parameters, we call a *parameter map*. Each monotone threshold rule $\gamma(\boldsymbol{x}; m, k)$ yields the parameter map of the form

$$f(p; m, k) = \sum_{i=k}^{m} \binom{m}{i} p^i (1-p)^{m-i}, \tag{2.4}$$

while a stochastic threshold rule $\sum_{i=0}^m \alpha_i \gamma(\boldsymbol{x}; m, i)$ has a parameter map of the form

$$f(p; m, \boldsymbol{\alpha}) = \sum_{k=0}^{m} \alpha_k f(p; m, k).$$

We will suppress dependence on m and k, unless necessary. Naturally, we can calculate the induced parameter in any layer by composing f repeatedly. Eventually, this will give us the induced distribution on the input to the root node, from which we can calculate the probability of error of the strategy (see Lemma 2.4.1). Thus, we will focus on studying the iterated behavior of parameter maps f associated with each decision rule γ.

A few remarks about some obvious choices for the leaf sensor nodes and the root sensor node. First, it is clear that leaf nodes receiving a single bit of data should simply send their bit up to their parent. Second, while it is optimal by the Neyman-Pearson lemma for the root node to use a maximum likelihood ratio threshold, we insist that the root node use the same threshold function as the rest of the nodes. This not only helps simplify our presentation, but is asymptotically equivalent (in the depth of the tree) to the root node using a maximum likelihood ratio threshold (see Appendix A.5 for an elaboration).

Naturally, when presented with the problem of detecting the two hypotheses in Equation (2.2), the first step is to design a decision rule with a parameter map such that the hypotheses are distinguishable. This requirement translates into the requirement that the sole unstable fixed point of the parameter map lies between the biases of the Bernoulli variables of the hypotheses. As the iterations accumulate, both hypotheses will converge to the two opposing fixed points and thus separate asymptotically. Clearly no channel can asymptotically distinguish every pair of hypotheses.

We focus on the majority threshold rule as it asymptotically distinguishes any symmetric hypotheses. While some stochastic rules can do this too, we conjecture that their per-sample performance will be worse than the majority threshold rule (see Appendix A.3).

2.4.2. Proofs and Lemmas. We begin with a statement of Theorem 2.3.1 in a more general form.

THEOREM 2.4.1 (Lower bound). *The probability of error of a depth $D+1$ tree strategy with parameter map $f(p)$ can be bounded from below as follows*

$$2p_e \geq 1 - 2f'(p_i)^D\delta,$$

where $f'(p_i)$ is the maximum derivative of the parameter map on the interval $(0,1)$.

This lower bound relies on two key lemmas that we prove below. The first lemma allows us to express the probability of error in terms of the parameter map iterates. The second lemma provides bounds on iterates of the map in terms of the exponent through a Taylor expansion.

LEMMA 2.4.1 (Probability of error). *Suppose that we perform the hypothesis test Equation* (2.2) *in a depth $D+1$ tree, with the parameter map $f(p)$. Then the probability of error is given by*

$$2p_e = f^D\left(p^* - \delta\right)) + \left(1 - f^D\left(p^* + \delta\right)\right),$$

where $f^D(p)$ denotes the D-fold composition of f with itself.

PROOF. This follows since

$$\begin{aligned} 2p_e &= \Pr(\text{root} = 1|H_0) + \Pr(\text{root} = 0|H_1) \\ &= f_r(f^{D-1}\left(p^* - \delta\right)) + \left(1 - f_r(f^{D-1}\left(p^* + \delta\right))\right), \end{aligned}$$

where f_r, the parameter map of the root node's decision rule, equals f. □

LEMMA 2.4.2 (Derivative lower bound on iterates). *Suppose $f : [0,1] \to [0,1]$ is a monotonically increasing, differentiable function with attracting fixed points at $0, 1$ (i.e., $f'(0) = f'(1) < 1$) and a unique inflection point at $p_i \in (0,1)$. Then for $\delta > 0$ we have that*

$$\begin{aligned} f^D(p^* + \delta) &\leq p^* + f'(p_i)^D\delta \\ f^D(p^* - \delta) &\geq p^* - f'(p_i)^D\delta. \end{aligned}$$

PROOF. The initial bound follows from Taylor expansion around the fixed point and overestimating the derivative. Iterated composition follows by monotonicity. □

PROOF OF THEOREM 2.3.1. Combining the previous lemmas and applying them to the parameter map of the majority threshold rule yields the result. It is also easy to see that the parameter maps of monotone threshold rules in general satisfy the requirements: parameter maps are Binomial survival functions, which are monotonic in p and whose derivative is unimodal in p, yielding a unique inflection point. Furthermore their fixed points at 0 and 1 are attractive (it can easily be seen that $f'(0; m, k) = f'(1; m, k) = 0$), which implies the existence of a repelling fixed point in $(0,1)$ by continuity. □

Next we provide a proof the upper bound.

LEMMA 2.4.3 (Upper bound: escape from fixed point). *Consider an iterable map, $f(p)$ that satisfies the same conditions as in Lemma 2.4.2. Let $D = D(\delta, x_0)$ be defined as*

$$D(\delta, x_0) \coloneqq \arg\min\{k \geq 0 : f^k(1/2 + \delta) < x_0\},$$

where $x_0 \in (0, 1)$. We then have that

$$\limsup_{\delta \to 0} \frac{D}{-\log_2 \delta} \leq \frac{1}{\log_2 f'(1/2)}.$$

PROOF. We focus on the convergence to the fixed point at 1, with the understanding that the other fixed point can be studied in a similar way through a simple conjugate operation by the function $T(p) = 1 - p$. We proceed by upper bounding the given parameter map with splines. For convenience of notation, we recenter and rescale $f : [0, 1] \to [0, 1]$ to the function $\hat{f} : [-1, 1] \to [-1, 1]$, by means of the conjugation operation $\hat{f}(p) = T \circ f \circ T^{-1}(p)$ where $T(p) = 2x - 1$. This brings the unstable fixed point at $1/2$ to 0.

Let us build a grid of points $x_i = 2^{-2^i}$ for $i = 0, 1, 2, \dots$ (e.g., $\frac{1}{2}, \frac{1}{4}, \frac{1}{16}, \frac{1}{256}, \dots$), which will serve as basepoints for our splines. To create a spline, we draw a line from the unstable fixed point at 0 to one of the grid points x_i. Thus, we define the spline function $s_i(p) = \frac{\hat{f}(x_i)}{x_i} p$. The sawtoothed upper bounding spline function is then a combination of these splines, defined as $g(p) = s_i(p)$ if $p \in (x_{i+1}, x_i]$.

Now that we have the upper bounding function, we want to get an estimate of the number of iterations required to get a value over the threshold x_0. It is easy to see that the number of iterations D_i required for $s_i(x)$ to escape the region $(x_{i+1}, x_i]$ is

$$D_i = \left\lceil \frac{\log \frac{x_i}{x_{i+1}}}{\log \frac{\hat{f}(x_i)}{x_i}} \right\rceil.$$

Adding up the iteration counts for each spline region, we get the total number of iterations for the upper bounding function

$$D \coloneqq \sum_{i=0}^{I} D_i,$$

where $I \coloneqq \arg\min_{i \geq 1} \left\{ 2^{-2^i} < \delta \right\} = \lceil \log_2(-\log_2 \delta) \rceil$. We can upper bound this quantity as follows

$$\begin{aligned} D &\leq \sum_{i=0}^{I} \frac{2^i}{\log_2 \frac{\hat{f}(x_{i+1})}{x_{i+1}}} + I \\ &\leq \sum_{i=0}^{I_\rho} \frac{2^i}{\log_2 \frac{\hat{f}(x_0)}{x_0}} + \sum_{i=I_\rho+1}^{I} \frac{2^i}{\log_2 \left(\hat{f}'(0) - \rho \right)} + I \\ &\leq \frac{2^I}{\log_2 \left(\hat{f}'(0) - \rho \right)} + O(I + 2^{I_\rho}), \end{aligned}$$

where $\hat{f}'(0) > \rho > 0$ is a quantity that tends to zero as δ tends to zero. The first inequality is simply an estimate upper bounding the ceiling function. The second inequality partitions the denominator terms into two classes: those where $\frac{\hat{f}(x_0)}{x_0} \leq \frac{\hat{f}(x_i)}{x_i} \leq f'(0) - \rho$, which span the indices $\{0, 1, \ldots, I_\rho\}$, and those where $f'(0) - \rho \leq \frac{\hat{f}(x_i)}{x_i}$, which span the indices $\{I_\rho + 1, \ldots, I\}$. The number of terms in each partition clearly depends on ρ, though as δ tends to zero we have $I \gg I_\rho$. Within each partition, we upper bound the terms by using the minimal derivative in each partition. Dividing through by 2^I, we thus obtain our result

$$\limsup_{\delta \to 0} \frac{D}{-\log_2 \delta} \leq \frac{1}{\log_2 \left(\hat{f}'(0) \right)}.$$

Therefore, we have that $D(\delta, x_0) \approx \frac{-\log_2 \delta}{\log_2 \hat{f}'(0)}$ (modulo a small term dependent on x_0). □

The previous lemma provided an upper bound on the number of iterations needed to escape a small neighborhood of the unstable fixed point. The following lemma quantifies the additional iterations needed to attain a desired accuracy.

LEMMA 2.4.4 (Upper bound: accuracy term). *For a parameter map $f(p; m, k)$ with $k \geq \lfloor \frac{m+1}{2} \rfloor$, we have $f^D(p_0) < \epsilon$ if $p_0 \in (0, \frac{1}{4})$ and*

$$D(\epsilon, p_0) > \log_k \left(\frac{\log \left(\binom{m}{k} \epsilon^{k-1} \right)}{\log \left(\binom{m}{k} p^{k-1} \right)} \right).$$

PROOF. Because $f(p)$ is concave around 0, we have that $f(p; m, k) \leq \binom{m}{k} p^k$ in a neighborhood of 0, if m is odd (if m is even, there is just a factor of $1/2$ difference in the binomial coefficient; since this difference is not significant, we focus on just the odd case). Let us call $g(p) = \binom{m}{k} p^k$.

By the monotonicity of both functions, the bound holds through function iteration, and hence $f^D(p) \leq g^D(p)$. We have that $g^D(p) = \binom{m}{k}^{\frac{k^D-1}{k-1}} p^{k^D}$, from which we can deduce our result through algebra. □

Combing the two lemmas above results in Theorem 2.3.2.

2.5. Discussion

2.5.1. Generalizing the Framework. While the framework presented here is stated in a fairly limited context, it can be used to approximate results in a much broader class of settings. Two major directions for generalization are increasing the bitrate and generalizing to a broader class of trees.

Let us consider how one might approach settings with a greater bitrate. Suppose that we have an m-ary tree where the sensors can send b bits of information and the leaf nodes begin with b bits. One way to reduce this case to one with a bitrate of 1 is to split the sensors, starting at the bottom of the tree, into b many sensors reporting only 1 bit and taking m bits as input. This procedure is equivalent to making the reduction that the original sensor groups its inputs into b many chunks of m bits and operates on them individually. The result of this operation is to change the in-degree of the layer above from m to $b \cdot m$ and replace the in-degree of the nodes on the given level to m. This procedure can be iterated up the tree, which results in a root node with in-degree b and the subtrees immediately following being perfect m-ary subtrees with bitrate 1. Each of the subtrees can be analyzed using our iterative mapping framework and then combined in a fusion center at the root. Naturally, this is an underestimate of the true performance capabilities of the perfect tree, since we reduced the capabilities of each node, however the quality of the approximation should improve as m increases due to sample pooling. This would make an interesting point for further investigation.

Now let us turn our attention to a broader class of trees. Given a general non-perfect, we can either add sensors or remove sensors to turn the tree into a perfect m-ary tree for some m. While the exact procedure will depend on the tree, only adding sensors will produce an overestimate of the performance (since completing the tree will require adding leaf nodes, which in turn add data). Analogously, only substracting nodes will produce an underestimate. This can give one a range of estimates for the accuracy of the actual tree of sensors. Other operations that may be useful for

completing trees are splitting nodes, deleting intermediate nodes, and adding intermediate nodes. Splitting nodes refers to a similar procedure described in the paragraph above where a node with too many inputs is split into multiple nodes with fewer inputs. Deleting intermediate nodes is the operation of reconnecting a node's children directly to their grandparents and removing their parent; adding intermediate nodes is the inverse of this operation. Using these operations, one can massage a tree that is approximately perfect into one that can be handled by our framework. More detailed investigations of these approximations could provide interesting results.

In the practice of sensor networks, the trees will have additional structure in being the spanning trees of a sensor network. Since sensor networks are commonly laid out in a lattice grid or another planar graph, we can expect the spanning trees to have more regularity than general trees. Additionally, the spanning tree algorithm can be modified to select more regular trees. With these considerations, our framework is likely more applicable to the trees that come up in practice.

2.5.2. Future Directions. There are many questions left to answer. The characterization of the performance of general tree strategies remains out of reach. It is likely that identical monotone threshold strategies will in fact be asymptotically optimal among all identical strategies, though this is a hard fact to prove. The exact characterization of the effects of the bitrate change are also unknown and likely will not be derivable from the methods presented here. The fact that the problem requires tracking the parameters of the distributions as they move up the layers, means that with a higher alphabet size, the distributions being transformed only become more complicated. Without a significant reduction to the family of parameter maps, progress here will be hard to come by.

CHAPTER 3

Graph Scan Statistics with False Discovery Rate Control

3.1. Introduction

This work studies the detection and localization of Porcine reproductive and respiratory syndrome (PRRS) outbreaks in pig farms by monitoring the transportation of pigs between farms as well as PRRS positive tests. More generally, we are interested in detecting and locating anomalously large values of a count variable in different locations and times by considering a dynamic network which represents connections between the sites for that time period. To tackle this problem we build a spacetime graph in which each vertex corresponds to a time period (such as year) and location (such as farm). We can increase the power of our tests by scanning the spacetime graph for neighborhoods where there are abnormally large counts. By "fusing" the measurements of sites within spacetime neighborhoods, we can increase the signal-to-noise ratio (SNR) and detect outbreaks more quickly and at fainter increases in intensity.

We will tackle two broad problems: detection and localization. Detection of an outbreak consists of testing whether or not an outbreak has occurred anywhere within the spacetime network. Our approach will follow the scan statistic framework in which we test for elevated intensity for a Poisson count distribution within neighborhoods of vertices in the spacetime graph. Each neighborhood produces a test with a corresponding p-value, and so multiple testing corrections must be applied to ensure that detection is not overwhelmed by false alarms. The error metric that we consider for the detection problem will be the family-wise error rate (in the weak sense), which is the probability that there is any false alarm when all of the observed counts follow the null hypothesis. Thereby if the entire network over the study time period sees a "normal" amount of infection then controlling the family-wise error rate means that we ensure no false alarms.

Localization refers to determining the locations of the outbreak within the spacetime network. In the testing framework, these locations are the neighborhoods and we consider there to be a

detected outbreak if we reject the corresponding null hypothesis for that neighborhood. The error metric that we use for localization is the false discovery rate, which is the expected ratio of the false alarms to the total alarms raised. Controlling the false discovery rate is more forgiving than controlling the probability of any false alarm or missed outbreak.

In what follows, we will begin by reviewing the standard literature on scan statistics and work our way to recent developments in graph scan statistics. We will also develop the necessary background to discuss false discovery rate and family-wise error rate.

3.2. Literature Background

3.2.1. Scan Statistics. The field of scan statistics is a mature field, with the study of scan statistics going back to Sir Ronald Fisher in the 50s [**43**] (and arguably even further back with Rev. Micheller calculation of the probability of celestial body clustering [**80**]). The prototypical question in scan statistics asks for the probability of a cluster of events in a given region in time or space. For example, decision-makers in public health may be interested in clusters of cancer cases or birth defects. Systems infrastructure professionals look to design systems with capacity to accommodate clusters of system requests. Quality control specialists may investigate clusters of defects in a product. Regardless of domain, scan statistics help distinguish a plausible coincidence from an improbable grouping indicative of a deeper issue.

The now standard reference on scan statistics [**48**] defines the 1-dimensional scan statistic S_w as the largest number of events in a window of length w. This statistic forms the central object of study, as it is necessary to a researcher interested in answering the question "what is the likelihood of a cluster of a given size forming in my data?" The space one scans over can be single-dimensional, as in a time-series, or multidimensional, as in a spatial geographic scan statistic. The space can also be either continuous or discrete. In the multidimensional case (such as latitude and longitude), the complexity is increased as one can scan over a variety of regular shapes (such as circles, ellipses, and rectangles), ideally choosing one most closely resembling the true regions of interest. The features of the statistic distribution are naturally different if the underlying space is discrete or continuous (e.g., scanning in time vs scanning RNA strings). The perspective taken in the book by Glaz et al. is directly probabilistic and focuses on deriving approximations to the distribution of the

scan statistic S_w in order to aid the computation of critical values. Naturally, scan statistics differ depending on the family of distributions assumed for the observed data, with typical examples being Gaussian [**64**], Poisson [**73**, **137**], and Binomial distributions [**67**]. The difficulty in calculating the probabilities naturally increases with the complexity of the domain and the exotic nature of the count distribution.

This extensive work has led to a number of software packages. The widely used spatial scan statistic SATScan [**70**] exists as free software, for example. SATScan works by scanning either a continuous or discrete multidimensional domain with regular shapes (such as circles, ellipses, and rectangles). For each shape and location it then computes a p-value for a hypothesis test for that statistic, then uses the smallest such p-value as the resulting statistic. To determine a threshold, the method will simulate under the null hypothesis (such as Gaussian white noise) and then calculate the scan statistic. By repeating this procedure it can determine a threshold that gives the desired confidence level. Limitations of SATScan are provided in [**87**]. Mainly, the shapes that it uses can be too restrictive for localizing an anomaly and may suffer from a loss of power.

3.2.2. Graph Scan Statistics. In recent years, graph scanning has become more prevalent as graph data has become more prevalent [**1**, **82**, **100**]. Since graphs may not necessarily be spatially embeddable, regularly-spaced, or uniform in dimension, a general graph does not lend itself to scanning with regular shapes. Therefore the theory and methods of regular scan statistics is less relevant than one may hope.

One of the works that began to address this recently is [**5**], which states the following:

> The task of detection in networks is critical for an increasing number of applications, for example, in surveillance and environment monitoring. Although the detection problem formulated above seems of great practical relevance, the statistics literature is almost silent on the subject, with the notable exception of the closely related topics of change-point analysis [...] and sequential analysis [...]. What is further puzzling is that a number of publications addressing the task of detection in sensor networks all assume overly simplistic models. For example, [...] the values at the sensors are assumed to all have the same distribution under the null and the alternative.

Let us present the hypothesis testing framework from [**5**] to give us some language (our notation will differ in order to stay consistent with the rest of this dissertation). Let a network of n vertices be denoted by G_n (which may have either a Euclidean or a general graph structure). To each vertex $v \in G_n$, we attach a random variable X_v, which carries the data the vertex may collect (this may multi-dimensional). We will suppose that these variables are independent at each site. The data will be assumed to follow a normal location model as it is popular in the signal processing literature. Thus a situation where there is no signal (the null hypothesis) is modeled as

$$H_0^n : X_v \sim N(0,1), \quad \forall v \in G_n,$$

where $N(\mu, \sigma)$ denotes the normal distribution with mean μ and variance σ. Let $K \subset G_n$ be a cluster, or a subset of vertices, generally some connected component of G_n. The case where the vertices in K present an anomaly is modeled as

$$H_{1,K}^n : X_v \sim N(\mu_K, 1), \quad \forall v \in K,$$

where $\mu_K > 0$. Decomposing μ_K as $\mu_K = |K|^{-1/2}\Lambda_K$, where $|K|$ is the number of vertices in K and Λ_k is the signal strength, we have that for any cluster K, the minimal error rate is

$$\min_T \Pr(T = 1|H_0^n) + \Pr(T = 0|H_1^n) = 2\Pr(N(0,1) > \Lambda_K/2),$$

where the minimum is taken over all tests T for H_0^n versus $H_{1,K}^n$ and the minimum is attained by the likelihood ratio test (by Neyman-Pearson's lemma). This is only a simple hypothesis testing problem, however. Letting $\mathcal{K}_n$ be a collection of clusters in G_n and defining

$$H_1^n = \cup_{K \in \mathcal{K}_n} H_{1,K}^n,$$

a test between H_0^n and H_1^n is then a true scanning problem, while the testing problem becomes composite. The worstcase risk for the problem is defined as error of the worst-performing region

$$\gamma_{\mathcal{K}_n}(T) = \Pr(T = 1|H_0^n) + \Pr(T = 0|H_{1,K}^n)$$

and the minimax risk is then the value of the test with the best worst-case performance

$$\gamma_{\mathcal{K}_n} = \inf_T \gamma_{\mathcal{K}_n}(T).$$

Two hypotheses H_0^n and H_1^n are said to be *asymptotically inseparable* if

$$\liminf_{n\to\infty} \gamma_{\mathcal{K}_n} = 1,$$

meaning that no test can do better than guessing, even with access to data. Naturally, the hypotheses are asymptotically separable if that same limit tends to zero.

A standard scan statistic in this space is

$$\max_{K\in\mathcal{K}_n} \frac{1}{\sqrt{|K|}} \sum_{v\in K} X_v.$$

The normalization allows for different-sized clusters K to have the same standard normal distribution under the null, allowing for comparison of different-sized clusters on the same scale. This scan statistic has been used in many areas, including in point cloud detection [**48**] and in outbreak detection [**38**, **69**].

A large majority of the literature assumes G_n to have a Euclidean structure (as described in the scan statistics section above). The cluster class $\mathcal{K}_n$ in this case often derives from a class of domains $\mathcal{A}$ in $\mathbb{R}^d$ as

$$\mathcal{K}_n = \{K = A \cap G_n : A \in \mathcal{A}\},$$

where A are often scalable templates. When the template class is parametric, the scan statistic was proved to be asymptotically minimax [**6**]. More general results with more technical conditions are available in [**5**, **104**] (the conditions are placed on the cluster shapes and the signal strength). Other graph scan statistics can be found in [**91**, **106**, **112**].

In the present work, our statistic will be similar to [**91**], using shortest path-length balls as the scan regions. Secondly, instead of focusing on asymptotic minimax bounds, our central concern is the control of false discovery rate. In a recent work [**66**] provided asymptotic distribution derivations for the multiscale scan statistic over discrete space and under the assumption of data distributed under an exponential family. In some cases, these statistics are shown to be optimal. The scan statistic is

designed to control the family-wise error rate (FWER; defined in the section after) asymptotically and across many scales and region shapes simultaneously. The scan statistic we use is based on this work.

3.2.3. Multiple Hypothesis Testing. Multiple hypothesis testing arises often in scientific endeavors when analysts seek answers to multiple questions in a single analysis. When a family of hypotheses are tested, a higher probability of false positives among the family is incurred. A concrete instance arises in drug testing, when a single drug is tested for multiple benefits (endpoints). A suite of corrections have been developed to control these errors [**11**, **56**, **108**]. Let us define the common measure of error called the *the family-wise error rate* (FWER). Consider performing a multiple test of m hypotheses. Define the random variable V to be the number of false rejections among these hypotheses and S to be the number of correct rejections. The FWER is defined to be the probability of having at least one false rejection in all the hypotheses tests

$$\text{FWER} = \Pr(V \geq 1).$$

This turns out to be a strict measure of error. A less stringent measure of error is the *false discovery rate* (FDR). The FDR is defined as

$$\text{FDR} \coloneqq \mathbb{E}[Q]$$

$$Q \coloneqq \begin{cases} \frac{V}{V+S}, & V+S \neq 0 \\ 0, & V+S = 0 \end{cases},$$

where the expectation is understood to be taken over a distribution over the hypotheses. In words, this is the expected ratio of false rejections to the total number of rejections. In general, the FDR is a lower bound on the FWER[1] and allows for greater power (true positive rate) at the cost of tolerating more false rejections.

When testing for multiple hypotheses, one can control family-wise error rate by the Bonferroni correction, in which each p-value $p_i, i = 1, \ldots, m$ is adjusted by a factor of $1/m$. This correction can be overly conservative however because it is designed to ensure that it is unlikely to have a single

[1]When all of the hypotheses are true nulls, then the FDR reduces to the FWER because all rejections are false.

false positive, thereby controlling the family-wise error rate (the probability of a false positive). In the Benjamini-Hochberg (BH) procedure [**11**] is a simple method that is designed to control the false discovery rate (FDR). The BH procedure sorts the p-values from largest to smallest and rejects all of the p-values after a stopping criteria (when a p-value falls below a line of slope α with 0 intercept). [**11**] showed that even under some dependency structures (positive regression dependence) the BH procedure still controls the FDR by α, and also provides some corrections under general dependence. Hierarchical FDR control refers to a simultaneous FDR control in which the FDR is not only controlled at the base level, but also at every level of a hierarchy [**134**]. In our setting, we want to fuse measurements to improve the power of our test, not to make a more stringent FDR control procedure.

Some works achieve this, although not in exactly the same setting as our own. In [**49**], one sided scanning windows are scanned p-values are aggregated to give a window based p-value, but the validity of their FDR controlling procedure is highly specific to their setting. In [**136**], regions in images are aggregated using the median p-value, and then FDR is controlled by estimating it directly through the Storey procedure [**113**]. The methodology in [**136**] is most like our own, but we directly use the BH procedure to our aggregated p-values.

3.2.4. Applications. For many examples of detection applications, see [**5**, **103**]. One area of application is anomaly detection in digital images. An image can naturally be considered as a rectangular grid of pixel color values, at which point the detection of anomalous regions can be tackled as the discovery of regions of pixel average deviation. Another area is epidemics detection. Early warnings for epidemics is a key concern in the emerging field of syndromic surveillance [**55**,**124**]. Given the improvements in data collection and aggregation in health systems, substantial research effort is now devoted to detecting and localizing outbreaks of health problems. This in turn can be used to detect the onset of influenza season and other suspected viruses.

Two other domains that provides a natural source of detection problems are that of surveillance systems and wireless sensor networks (WSNs). A quote from [**35**] gives us a sense of the applications of WSNs:

> "The first category includes environmental and habitat monitoring, precision agriculture, indoor climate control, surveillance, treaty verification, and intelligent alarms.

> The second includes structural monitoring, ecophysiology, condition-based equipment maintenance, medical diagnostics, and urban terrain mapping. The most dramatic applications involve monitoring complex interactions, including wildlife habitats, disaster management, emergency response, ubiquitous computing environments, asset tracking, healthcare, and manufacturing process flow."

For a concrete example, consider the sensor network proposed in [**22**] to detect clandestine radioactive weapon travel on road sides. Traditionally, portal monitors are placed on "choke point" entrances to potential targets such as cities, which have the drawbacks of conspicuity and increasing vehicular traffic. In contrast, the sensor network solution proposed is smaller and discrete, has lower power costs, and maintain accuracy compared to portal monitors, while allowing detection at higher traffic speeds. Such WSN detection problems will also have relevance to the chapter of this dissertation that deals with hierarchical compressed statistics.

3.3. Data and Methodology

3.3.1. Monitoring PRRS Outbreaks in Pig Farms. PRRS is a viral disease characterized by two main symptoms, reproductive failure in pregnant sows and respiratory issues in pigs of any age. PRRS is the most economically damaging disease affecting United States swine production after the now-eradicated classical swine fever [**77**]. While it was only reported in a few countries in the late 1980s (such as North America and China), it can now be found in major swine-raising countries worldwide. Its annual costs of damage have been estimated to be around $664 million in the US [**58**]. This had led to entire production systems being designed around strategies for controlling or eliminating this disease [**85**].

An important feature of PRRS epidemiology is that it is able to persist in a carrier pig for up to 200 days (although, commonly a pig becomes immune within 2 months). The primary route of transmission is close contact with a carrier pig; infection likely takes place in nose-to-nose contact or by contact with urine or faeces. The virus spreads readily through bodily fluids and contact, along with aerosol transmission in a range of 3 km. Overall, the transmission between groups of pigs is an ongoing area of research [**3**].

Accurate assessment of a farm's exposure to the virus is difficult and expensive. Farms must perform triage and target high risk areas with vaccines or other sanitation measures [**78**]. Pig transmission between farms, from birthing farms to nurseries to finishers or during sale, introduces the potential for cross-farm infection [**89**]. While infection can be mitigated through quarantining procedures and through the purchase of only PRRS-free stock, inconsistency in control standards allows the disease to continue to spread.

3.3.2. Bioportal Data. Underlying our analysis is a proprietary data set from the pig industry. The data concerns outbreaks of PRRS at approximately 200 farms in North America.

The data contains the following features: spatial location of the farms, farm type (sow farms, nurseries, finishers, wean-to-finishers, and boar farms), timestamped information about pig shipments between the farms, and case data from pig virus tests. Each farm, or premise, has a geolocation coordinate described by its longitude and latitude position. The type of the farm determines the specialization of the pigs at the farm. Accordingly, different farm types have different infection profiles. For example, because the virus can spread from a sow to its piglets in gestation, PRRS can persist in sow farms. The farm type also affects the pig trade characteristics (pig shipments tend to roughly flow from sow farms to nurseries to finishers, though exceptions exist). All the information described below is considered after dropping missing values and selecting only the trade shipping types (i.e., filtering out deaths and other types of movement).

The virus case data provides a timestamped table of PRRS incidents, along with the farm where the case was reported. Each case corresponds to a test confirming the presence of the virus at a given farm. While genetic data for each case is available, we did not use it in this analysis. Overall, the number of cases reported is 381 in the years from 2005 to 2013. The average number of cases reported by each farm is 1.12, while the median was 0. The number of farms with more than 3 cases was 21. The maximum number of cases was 25. A histogram of the number of case counts can be seen in Figure 3.2. While most farms are unaffected or have had a single incident, a small number of farms have case numbers in the 4-6 range.

The pig shipment information contains 38,792 entries. Each timestamped entry reports the headcount, or the number of pigs, in the shipment. The average headcount is 55, including zeros, approximately 2500, excluding zeros, the total number of pigs moved over the 8 years is 11 million,

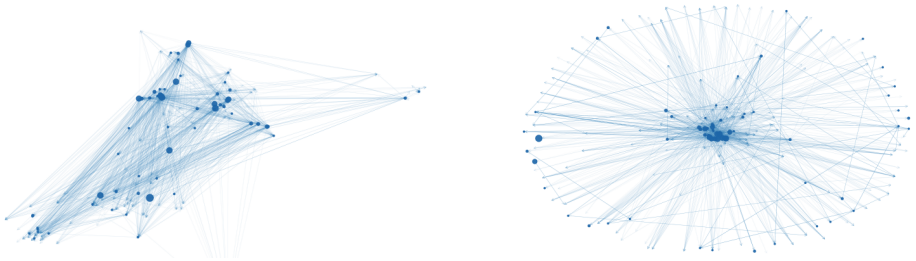

FIGURE 3.1. The two networks here present the case data and the movement data for the years 2005-2013 in two different network embeddings: spatial (left) and force-directed (right). The spatial embedding uses the GPS coordinates to embed the graph in a plane. The force-directed layout uses the "spring layout" from NetworkX [**50**], which places springs on the graph edges with strength proportional to the frequency of transportation between nodes and runs a simulation of the physical spring system to determine an equilibrium state. The size of each node is proportional to the number of cases at the node. The opacity of each edge is in proportion to the normalized total pig headcount traveling along that edge.

and the maximum number of pigs moved between two farms in a year is approximately 15000. The spatial adjacency matrices, pre-normalization, have peak values at approximately 15000 and the rest (excluding zeros) average out to 2442, median 1995 (including zeros, the average is 55). This data allows us to build an interaction graph for the farms, which can be seen in Figure 3.1. While the graph is quite dense, an infectious cluster can be seen in the center of the force embedding (right), which propagates out towards lower case-count nodes.

Of the pairs made from the 204 premises, just 43 pairs were within the airborne transmission radius of one mile; 34 of these pairs had no shipment connection.

Due to the highly contagious nature of PRRS, an outbreak in a pig farm can spread to other farms by the transportation of pigs. We can visually observe the impact of the transportation of pigs on the spread of PRRS between the farms (see Figure 3.1, left, for a spatial embedding). The edges in the graph are weighted proportional to the number of pig shipments between farms. We see that the cases are dispersed across spatially distant farms, however if we plot the farms according to a force-directed graph layout then we see that the high case counts concentrate within a highly connected core of farms (see Figure 3.1, right). This suggests that more than spatial proximity, the network connectivity is driving the spread of PRRS in the farms.

3.3.3. Spacetime transportation network structure. In this section, we will describe the network structure in our dataset. Roughly speaking, the network is constructed out of the pig shipment and spatial proximity between the farms. The shipment data is grouped into year-bins, which produces one network for each year. These networks are joined into a single spacetime network by connecting identical farms across time. Before we go into detail, let us set some notation.

Let $G = (V, E, A)$ be a weighted directed graph, where $V = \{v_1, \ldots, v_n\}$ is the graph vertex set, $E = \{e_1, \ldots, e_m\}$ is the graph edge set, and A is the adjacency matrix with entry A_{ij} containing the strength of the connection between nodes i and j. Let us denote the shortest path length (SPL) distance (i.e., shortest number of hops ignoring the edge weight) by $d(\cdot, \cdot) : V \times V \to \mathbb{R}$ and the corresponding ball of radius r by $B(v, r) = \{w \in V \mid d(v, w) \leq r\}$. The number of nodes in said ball will be denoted by $|B(v, r)|$. Nodes and edges can carry values such as case counts via node and edge functions $f : V \to \mathbb{R}$ and $g : E \to \mathbb{R}$.

As previously mentioned, each node in the network represents a farm at a point in time. Each farm has eight copies, representing the farm in each of the eight years in the data range. An edge between nodes exists if any of the following hold: the two nodes represent the same farm separated in time by a single time unit (a year), the two nodes represent farms that have a pig transportation relationship, or the two nodes are within each other's airborne transmission radius. If the edge connects the same farm in time, we call it a temporal edge, otherwise it is called a spatial edge. Due to the presence of spatial and temporal connections between the nodes, we call this type of network a *spacetime network*. A visual representation can be found in Figure 3.6.

The time dimension is binned into 8 year-long intervals, spanning the years 2005-2012 (both ends inclusive). This time-length is a scale choice made based on PRRS gestation and infection length: we need a scale choice that is short enough to have a fast resolution, but long enough to capture the transmission dynamics.

The weights of the edges correspond to transmission parameters in our susceptible-infected-susceptible (SIS) model (the details of this model are described in Section 3.3.6). That is, the higher the parameter, the larger the probability that an infection is transmitted between two nodes along that edge. The temporal edges are set with a dummy weight; this is because the SIS model deterministically transmits the infection state from one farm to its copy in the next year, more on

that later. Temporal edges are nonetheless important for detection and localization neighborhoods. The weight on spatial edges is a function of the size of the pig shipments between the farms and an adjustable transmission parameter in our model.

It is important to note that the edges are directed because 1) pig shipments represent one-way transmission vectors and 2) temporal connections transmit the infection from a past version of a farm to the future. Many farms have a bidirectional shipment relationship, which gives rise to two separate edges between the corresponding nodes. Due to the directedness of the graph, the neighborhoods formed by the SPL distance are irregularly shaped and asymmetric. Given that we are interested in the localization of outbreaks and detecting hotspots, the neighborhoods are aimed at capturing the regions of consequence for an infected farm. In that sense, each neighborhood corresponds to an area of influence for a farm, in terms of proximity and pig shipment flow; in graph theoretic terms this is called an out-neighborhood. To that end, the average out-degree of each node in our formed network was 8.6, while the range of the out-degrees was $[1, 47]$.

If two farms are less than 1 mile apart, then an airborne transmission factor is introduced into both of the directed edges. In our dataset, there were 43 pairs of farms with this relationship. We detail the setting of spatial proximity parameter in the parameter tuning section.

3.3.4. Detecting with Graph Scan Statistics. The first problem we tackle is the problem of detecting infectious spread on a network, to help us establish the presence of network effects. We frame the problem as a multiple hypothesis testing problem and then present a p-value based scan statistic with error rate control.

Let us first formulate the method as a hypothesis test. Fixing a neighborhood size r, let us denote the collection of all SPL neighborhoods by $\mathcal{R} = \{B(v, r) \mid v \in G\}$. Suppose we have an n node network G with integer values on its vertices, fix a neighborhood size r, and consider the hypotheses

$$H_0 = \cap_{v \in G} H_{v,0} = \cap_{v \in G} \{N(v, r) \sim \text{Pois}(\lambda_0 | B(v, r)|)\}$$

$$H_1 = \cup_{v \in G} H_{v,1} = \cup_{v \in G} \{N(v, r) \not\sim \text{Pois}(\lambda_0 | B(v, r)|)\},$$

where $N(v,r)$ is the total case count in $B(v,r)$, the SPL neighborhood or radius r, and λ_0 is a fixed background infection rate. The null hypothesis corresponds to the assumption that each node has infections according to an independent background process. The alternative corresponds to finding at least one neighborhood where the counts are elevated beyond this assumption. The likelihood ratio test (LRT) for the above hypotheses is based on the test statistic

$$T_{B(v,r)}(Y,\theta_0) \coloneqq \sqrt{\frac{\sup_{\theta\in\Theta}\prod_{i\in B(v,r)} f_\theta(Y_i)}{\prod_{i\in B(v,r)} f_{\theta_0}(Y_i)}}$$

where f_θ denotes the density of the Poisson distribution and the null $H_{v,0}$ is rejected when $T_{B(v,r)}(Y,\theta_0)$ is large. It is known to be powerful and have optimality properties (with assumptions on the structure of $\mathcal{R}$, [**71**]). Note that this test is a local test, in the sense that it provides a test per neighborhood (i.e., a test between $H_{v,0}$ and $H_{v,1}$, for a given v). The generalized likelihood ratio test (GLRT) T_n provides the global test over the entire graph

$$T_n \coloneqq \max_{R\in\mathcal{R}} T_R(Y,\theta_0).$$

This statistic controls the family-wise error rate (FWER) (see [**66**]).

In the Poisson case, following [**66**], we can write $T_{B(v,r)}(Y,\theta_0)$ as

$$\begin{aligned} T_{B(v,r)}(Y,\theta_0) &\coloneqq \sqrt{2|B(v,r)|\left[\bar{Y}(v,r)\log\left(\frac{\bar{Y}(v,r)}{\exp(\theta_0)}\right) - (\bar{Y}(v,r)-\exp(\theta_0))\right]} \\ &= \sqrt{2|B(v,r)|\left(\log\left(\frac{N(v,r)}{\mathbb{E}[N(v,r)|H_0]}\right)-1\right)+\mathbb{E}[N(v,r)|H_0]}, \end{aligned}$$

where $\bar{Y}(v,r)$ is the average of the counts in $B(v,r)$, $N(v,r)=\bar{Y}(v,r)|B(v,r)|$, $\exp(\theta_0)=\lambda_0$ is the expected number of counts per vertex, and $\mathbb{E}[N(v,r)|H_0]=|B(v,r)|\lambda_0$. From the latter form, it is clear that the LRT statistic is a monotonically increasing function of the number of counts in a region. This dependence relates it to the p-value statistic, which is a monotonically decreasing function of the same quantity, defined as

$$p_v \coloneqq S(N(v,r);\lambda_0|B(v,r)|) = \sum_{k=N(v,r)}^{\infty} e^{-\lambda_0|B(v,r)|}\frac{(\lambda_0|B(v,r)|)^k}{k!}, \tag{3.1}$$

where $S(x;\lambda)$ is the survival function of the Poisson distribution with parameter λ. Given this equivalence relationship, we elect to use the p-value statistic, due to its simplicity. The global test on the whole graph is analogously $p = \max_{v \in G} p_v$.

For individual tests, $H_{v,0}$ versus $H_{v,1}$, thresholding the p-value at a desired significance level will give the correct type 1 error rate. However, for the global null test of H_0 versus H_1, a multiple testing correction needs to be applied in order to obtain a desired type 1 error. Type 1 error control for the global null test is equivalent to family-wise error rate (FWER) control, namely the FWER is controlled at level $\alpha \in (0,1)$ for a test Φ if

$$\sup_{R \in \mathcal{R}} \mathbb{P}_{H_{R,0}}[\Phi \text{ rejects any } H_{R',0} \text{ with } R' \subset R] \leq \alpha.$$

A crude method to achieve this is the Bonferroni correction [**56**], whereby the significance level is divided by the number of tests (neighborhoods) and then if any p-value is below this adjusted significance level then we reject H_0—a procedure that guarantees family-wise error rate control. However, when the tests are correlated, this adjustment may be too aggressive and a less severe adjustment may be in order. Popular implementations of the scan statistic of regular shapes, such as the SatScan [**70**] software, will directly calculate a p-value for the global null test by repeatedly simulating under the null hypothesis, each time calculating the GLRT statistic, and approximating this global p-value with the proportion of times that this exceeds the actual GLRT statistic (calculated from the real data).

To recap the hyperparameters in the scan statistic: the background rate λ_0, the neighborhood size r, and the detection threshold q. The background rate can be set through a number of methods, including moment matching in an isolated region of the data or with population-level infection data. The neighborhood r is a hyperparameter that depends on the expected size of the region of influence for infectious farms; we found $r = 1, 2$ to give reasonable results. The optimal detection threshold can be found empirically, controlling for the false positives while maximizing the test power. Applying these scan statistic to the spacetime graph, we can detect the presence of elevated regions and establish with high likelihood the presence of regions with high infections due to network effects (i.e., pig shipment or airborne transmission).

3.3.5. Localizing Infections with Graph Scan Statistics. The next problem is that of localization. Once we have established the presence of network effects, we aim to localize regions of high activity, producing hotspots of high infection. This is a multiple testing problem with dependence between the hypotheses, so care must be taken to correct the increase in the number of false positives. In this section, we will outline the multiple testing problem, elaborate on the dependence structure of the hypotheses, introduce the false discovery rate, and show how the well-known Benjamini-Hochberg correction provides control over said rate [**10**].

We use the same p-value-based statistic as defined in the detection section in Equation (3.1), but we associate the p-value with each neighborhood, denoted as $p_v, v \in G$. We will aim to control the FDR. The Benjamini-Yakutelli (BY) correction [**11**] is a method for determining p-value thresholds with a guaranteed FDR control. The method proceeds by ordering the neighborhood-associated p-values in ascending order $p_{(i)}$ and then rejects all neighborhoods with index less than k defined by

$$k = \arg\max_i \left\{ \bar{p}_{(i)} \leq \frac{\alpha i}{m \cdot c(m)} \right\}, \tag{3.2}$$

where $m = |G|$, α is the desired control level for the FDR, and $c(m)$ is a factor based on the dependence between the hypotheses. See Figure 3.5 for a visual depiction. With no dependence the factor is $c(m) = 1$ (and is referred as the Benjamini-Hochberg (BH) procedure, due to its precedence); for arbitrary dependence, it is $c(m) = \sum_{i=1}^{m} \frac{1}{i}$ (the m-th harmonic number). While it may seem like we need a correction due to the dependence in our hypotheses, it turns out this factor is not needed. A result in the original BY correction paper states that if the joint distribution of the test statistics satisfies a condition called *positive regression dependence on a subset* (PRDS), then the BH correction controls the FDR at a level less than or equal to $\frac{m_0}{m}\alpha$, where m_0 is the number of true null hypotheses.

We will now provide a proof that our statistics joint distribution satisfies PRDS. This proof is based on the proof of Lemma A.2 in [**17**]. A set $D \subset \mathbb{R}^n$ is said to be *non-decreasing (non-increasing)* if for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$ with $\boldsymbol{x} \leq \boldsymbol{y}$, elementwise comparison, we have that $\boldsymbol{x} \in D \implies \boldsymbol{y} \in D$ ($\boldsymbol{y} \in D \implies \boldsymbol{x} \in D$).

DEFINITION 3.3.1. *A p-value collection* $\mathbf{p} = (p_i)$ *satisfies the strong positive regression dependence on any subset (PRDS) if for any non-decreasing set* $D \subset [0,1]^n$ *then*

$$\mathbb{P}\{\mathbf{p} \in D | p_i = u\}$$

is non-decreasing in u *for any* i*.*

PROPOSITION 3.3.1. *The p-value collection* $\{p_v, v \in G\}$*, where* p_v *are graph neighborhood p-values, satisfies the strong PRDS condition.*

PROOF. Let B_v be the ball centered around v and let $X[B_v] = \sum_{i \in B_v} X_i$ for random variables $X_i \sim$Poisson(λ_i). Notice that p_v is a non-increasing function in $X[B_v]$, so the PRDS condition is equivalent to

$$\mathbb{P}\{(X[B_v])_{v \in G} \in D | X[B_i] = m\}$$

is non-increasing in m for any non-increasing set D. Let $A = B_i$ and let $\tilde{X} \sim$Multinomial(α, m) for $\alpha_v = \lambda_v / \sum_{w \in A} \lambda_w$ and $\alpha = (\alpha_v)_{v \in A}$. Notice that $X_v | X[A] = m$ has the same distribution as $\tilde{X}_v$ for $v \in A$. Then we have that

$$\begin{aligned} \mathbb{P}\{(X[B_v])_{v \in G} \in D | X[B_i] = m\} &= \mathbb{P}\{(X[B_v \backslash A] + X[B_v \cap A])_{v \in G} \in D | X[A] = m\} \\ &= \mathbb{P}\{(X[B_v \backslash A] + \tilde{X}[B_v \cap A])_{v \in G} \in D\} \end{aligned} \tag{3.3}$$

Letting $\tilde{X}_1 \sim$Multinomial$(\alpha, m-1)$ and $\tilde{X}_1 \sim$Multinomial$(\alpha, 1)$ independently, we note that $\tilde{X}$ is identically distributed with $\tilde{X}_1 + \tilde{X}_2$. Then

$$\begin{aligned} \mathbb{P}\{(X[B_v])_{v \in G} \in D | X[B_i] = m\} &= \mathbb{P}\{(X[B_v \backslash A] + \tilde{X}_1[B_v \cap A] + \tilde{X}_2[B_v \cap A])_{v \in G} \in D\} \\ &= \mathbb{P}\{(X[B_v \backslash A] + \tilde{X}_1[B_v \cap A])_{v \in G} \in D - (\tilde{X}_2[B_v \cap A])_{v \in G}\} \\ &\leq \mathbb{P}\{(X[B_v \backslash A] + \tilde{X}_1[B_v \cap A])_{v \in G} \in D\} \\ &= \mathbb{P}\{(X[B_v])_{v \in G} \in D | X[A] = m - 1\}. \end{aligned}$$

where the inequality from the fact that $D - (\tilde{X}_2[B_v \cap A])_{v \in G} \subset D$ (which in turn follows from D being non-increasing) and the final equality follows from Equation (3.3). This proves the strong PRDS condition. □

3.3.6. Simulating Outbreaks with an SIS Model. To validate our model, we rely on the common Susceptible-Infected-Susceptible (SIS) compartment model of infectious spread. This model allows us to determine how our scan statistic performs on simulation data of the spread of an infection over a graph and compare with a null model, where infections occur. All simulations were done with the Epidemics on Networks (EoN) Python package [**81**].

To begin, let us describe the basics of a network SIS model. At any given point in time, a vertex is considered to be in one of two states: susceptible or infected. A background rate of infection is assumed, which is Poisson in its event distribution and governed by a spontaneous infection rate parameter λ. The corresponding interevent distribution is exponentially distributed with the same parameter. Once a node is infected, it can transmit the infection to its susceptible neighbors. The infection transmission parameter τ controls the global likelihood of transmission. The strength of the edge weight $a \in (0, 1)$ between nodes determines the actual likelihood of transmission $a \cdot \tau$. Once infected, the node's time to recovery is governed by an exponential distribution set with a recovery parameter γ. After recovery, the node can be infected again by its neighbors or spontaneously.

The null simulation is the SIS model with spontaneous infections but with the neighbor transmission parameter set to zero (i.e., there are no neighbor-based infections). The alternative simulation is the full SIS model as described above, with non-zero neighbor transmission parameters.

Some adjustments must be made to perform the simulation on a spacetime graph. Since the spacetime graph consists of temporal snapshots of the network, transmission needs to stay localized within each temporal slice. We simulate sequential runs of an SIS model in each snapshot, proceeding in ascending year order. When the simulation in one temporal slice is finished, we transfer the susceptible/infected state of each node to the next slice and begin the simulation with that initial condition. This lets us produce a case count for each vertex, which is a count of the number of times each node entered the infected state.

3.4. Experimental Results

3.4.1. Parameter Tuning. So far, we have described our general framework for scanning and for simulating outbreaks, but have not given procedures for setting parameters for the scan statistic and the model. In this section we give some guidance on how to do that. Our goal is a not precise

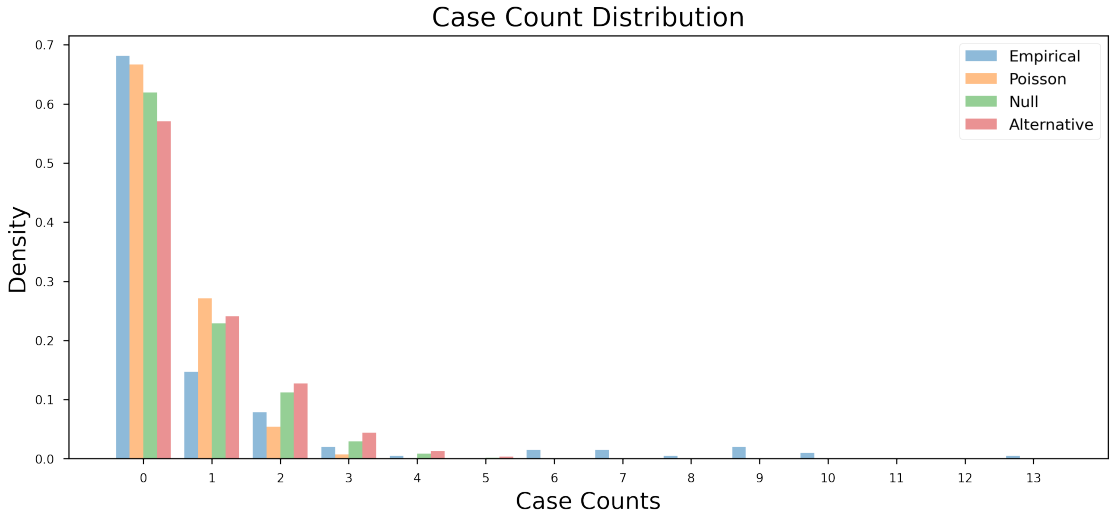

FIGURE 3.2. The empirical distribution of the case counts versus a typical simulation from each of the three models using our parameter settings. Note the match in the zero counts in the null model and the tail match in the alternative model.

model fit; rather we look for a qualitative match of the data distribution and the SIS models. Hence, we employ a blend of statistical and heuristic methods for the parameter choice.

To set the background rate ν and the recovery rate γ, we used a combination of plausible biological reasoning and moment matching with the data. Specifically, for ν, we chose parameters to match the number of farms with zero counts in the null distribution. Assuming a Poisson model, this leads to the estimator $-\log\left(\sum_{i=1}^{n} \mathbf{1}_{X_i=0}\right)$; in practice, we found that this estimator must be scaled by a factor of $1/2$ since both the empirical and the SIS model deviate from the Poisson model. With this approach, we set $\nu = 1/20$. Simultaneously, we chose $\gamma = 3$ to correspond to an average farm recovery time of 4 months; this is a plausible time scale, though in actuality this quantity is difficult to estimate. This completes the null SIS model (see Figure 3.2).

The alternative model requires the transmission parameter τ and the edge weights $a_{v,w}, v, w \in G$ to be carefully set. Since the effective transmission rate between nodes $v, w \in G$ is $\tau \cdot a_{v,w}$, we use $a_{v,w}$ to set the relative scale of the edges and τ is then used to scale the overall transmission strength. To set $a_{v,w}$, we use the number of pigs moved from farm v to farm w normalized by the maximum pig count over all the edges. This linear relationship effectively implies that each pig transferred carries the same added amount of risk. To set τ then, we match the tail of the case count

empirical distribution, keeping ν and γ fixed as before. Since closed form expressions for this figure are intractable, we proceeded by simulation and settled on a parameter range of $\tau \in [1, 2]$, giving an average transmission parameter range of $[1/10, 2/10]$. Finally, the spatial proximity parameter is set to 100, which is in units of headcounts. Recall that this parameter controls the added transmission strength of the edges between farms that are within a mile of each other (i.e., plausible airborne transmission distance). This parameter setting implies that for any farm pair within mile of each other, we effectively added an extra 100 pig movements between their farms in both directions. This quantity was chosen as it is on the order of the average pig shipment size (see the head count movement statistics in the data section). Of the 9 farm pairs that had a pig shipment relationship prior to the addition of these headcounts, the geometric mean of the percentage increase in the edge weights as a result of the addition was 1.46. To effectively bring the simulation to a regime with phenomena of interest, namely an epidemic, we started each simulation run by seeding the top 10 highest case count nodes as infected.

3.4.2. Detection. To measure the effectiveness of the scan statistic on the detection problem, we produced receiver-operating characteristic (ROC) curves [**40**] (see Figure 3.3). These curves show the distinguishability between the null and the alternative SIS simulations for a few versions of our scan statistic. Noting the overall trend of the scan statistics, the size of the neighborhood does not appear to change the performance in this case.

The distribution of the König $r = 2$ values for a batch of simulations can be seen in Figure 3.4. The two distributions are highly separated for our parameter setting, explaining why the scan statistic has such an optimistic ROC result. Naturally, this plots suggest that the alternative distribution shifts rightward and away from the null distribution as the global transmission parameter τ increases; thus τ is an effective difficulty parameter capturing the signal strength in the data. In turn, this confirms our theoretical reasoning that the scan statistic picks up the correct signal in the data.

3.4.3. Localization. We turn our attention to the localization of the epidemic using our scan statistics. The plots in Figure 3.5 provide a demonstration of the Benjamini-Hochberg procedure as it is applied to the spacetime simulations. Here we see that a certain few regions are rejected in the alternative, while zero are rejected in the null, aligning with our previous König statistic distribution

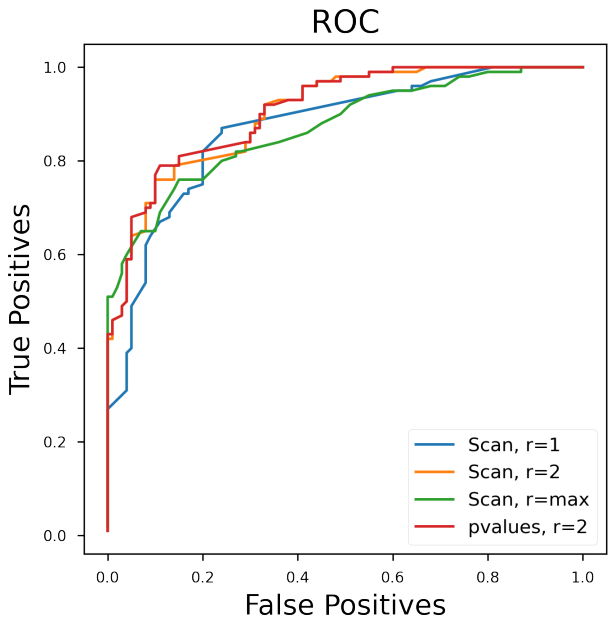

FIGURE 3.3. These curves represent the performance of the test statistics described in Section 3.3.4 on the detection task of distinguishing an alternative SIS simulation from a null SIS simulation. For each model, 100 simulations were performed on the spacetime graph built from the 2005-2013 data. Variation with the neighborhood size is captured, where the neighborhoods $r = 1, 2$, and 'max' (whole graph) perform similarly. Note that the p-value statistic performs almost identically well, as predicted in Section 3.3.4.

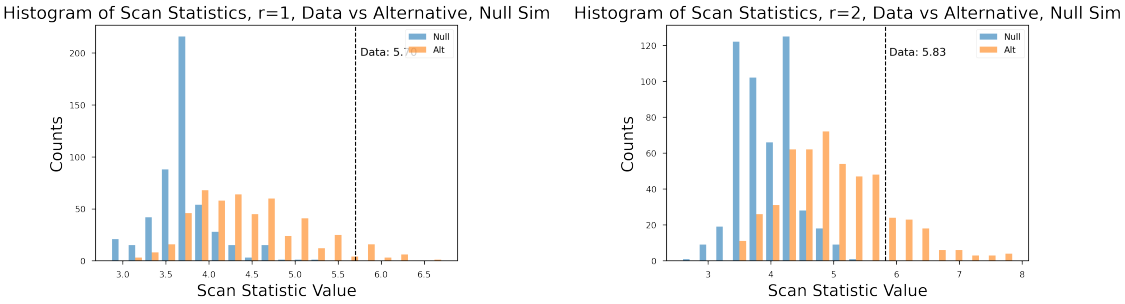

FIGURE 3.4. The distribution of the König statistic for $r = 2$ for a batch of 500 null and alternative simulations compared against the König statistic on the empirical data. The alternative distribution captures the behavior of the empirical data in its tail, while the null distribution never reaches those values. The overlapping region presents opportunity for false positives and false negatives seen in the ROC curves.

visualization. The rejected neighborhoods are visualized in a spacetime network in Figure 3.6. The years containing the largest rejected regions correspond almost entirely to the years with the largest

case counts. The rejected node centers split into two classes: high out-degree hotspots and low degree hotspots. The high out-degree hotspots (average out degree of 31 in the class, compared to 5.3 over the whole graph) are natural hotspots as they accumulate high counts through their many connection. The low degree hotspots were a bit more puzzling, as both their in and out degrees were small. The case counts in these neighborhoods were significantly smaller than the other rejected regions, however the case counts were still large enough to register as significant, given the smallness of interaction neighborhood of these nodes. Thus it appears that the scan statistic picked up on particularly surprising instances of virus transmission, suggesting the presence of transmission vectors other than that captured in pig shipments. Additionally, these two classes did not appear to have a strict relationship to farm type, as we found finisher-type farms in the high out-degree class and we also found nurseries in the high in-degree class. In fact, one of the highest case count nodes was a finisher farm with virtually zero out-degree and a small in-degree, contrary to expectation. Furthermore, the case count distribution of the rejected neighborhood center nodes show greatly elevated levels compared to the accepted. In Figure 3.7, the case counts of the accepted neighborhoods follow a Poisson-like distribution, while the counts of the rejected neighborhoods capture the flat, large-tail distribution seen in the empirical data in Figure 3.2.

3.5. Conclusion

In this work, we propose graph scanning methods for detecting and localizing disease outbreaks in space-time network. The localization method crucially relies on a multiple testing correction, which we proved controls the false discovery rate, even in the setting of hypothesis dependence. We validated the method with a series of SIS simulations tuned to a real dataset of pig farm outbreaks, confirming that the method can detect epidemics and localize hotspots. This methodology gives one the ability to detect incipient PRRS outbreaks and localize the outbreaks based on a transportation network. Naturally, it is would be interesting to see if these methods can lead to methods for robust early warnings in epidemic outbreaks in human populations and for other diseases. Further testing would be necessary, including more realistic simulations and updated parameter settings that more closely match the real-world environment. Extensions to distributions other than Poisson and perhaps to the continuous time setting [**88**] are also possible. Removing r as a hyperparameter and

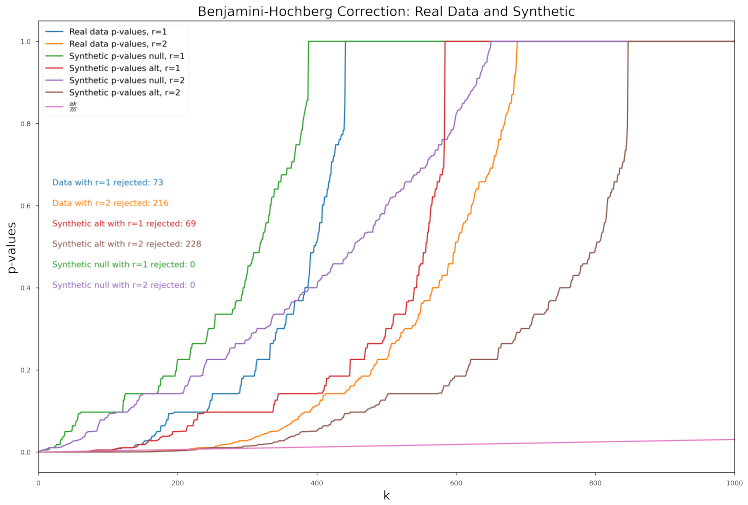

FIGURE 3.5. Benjamini-Hochberg correction curves for the data and the simulations using both neighborhood sizes $r = 1$ and $r = 2$. (The alternative spacetime simulations here were set $\tau = 2.0$ to increase the number of rejected regions, while keeping the rest of the parameters the same.) Here, the bottom-most pink line provides the rejection threshold with $\alpha = 0.5$. The number of rejected regions is reported for each line (for reference, the spacetime network used in the simulations contains 9×204 nodes, so for $r = 1$ the data had less than $1/30$ regions rejected). The flat values at the top correspond to large regions in the network that had no cases at all. Contrary to what would happen under independent hypotheses, these curves are non-linear due to the overlap between the neighborhoods.

performing multiscale scanning could also be a promising extension as in [**106**]. While our theoretical guarantees have concerned localization false discovery rate, a testing framework incorporating the power of the test could go directly towards addressing the needs in industrial applications. The software used for this work, without the proprietary data, may be released in the future on GitHub; meanwhile, it may be released to select inquiries.[2]

[2]See `www.github.com/dshemetov`.

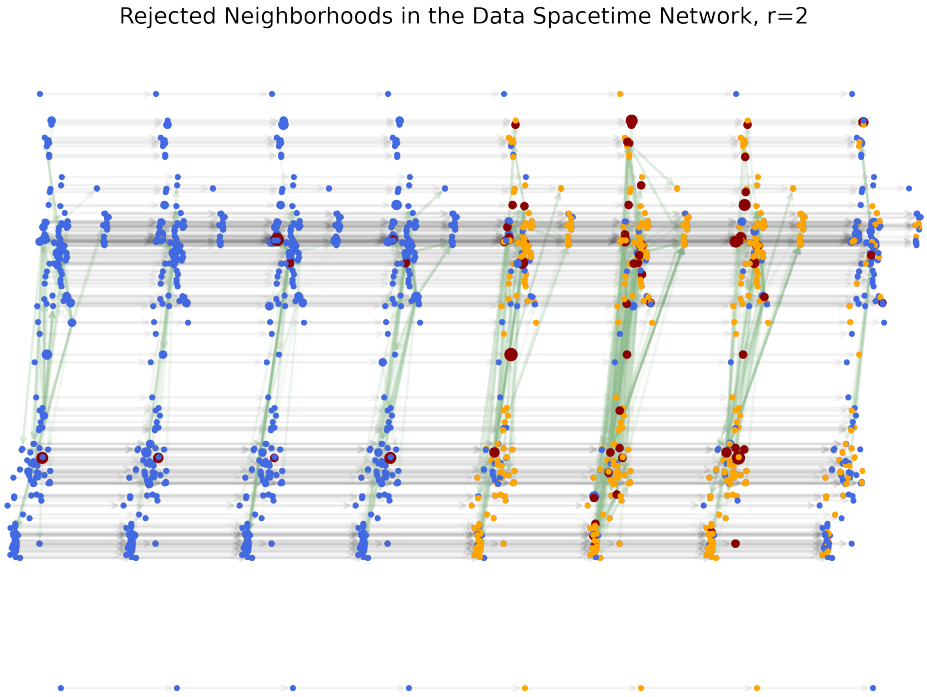

FIGURE 3.6. Here we see the spacetime graph laid out in slices from the year 2005 on the left to the year 2012 on the right, with the 10 smallest p-value neighborhoods (with BH rejection parameter $\alpha = 0.5$) colored. Only the out-edges for nodes with a positive number of case counts are drawn, for legibility (this results in a few visually disconnected components which are not actually present in the data). Nodes in accepted regions are colored blue, the rejected neighborhood centers are dark red, and members of the rejected neighborhoods with 0 case counts are colored orange. The orange nodes can therefore be seen as vulnerable nodes picked up by the statistic.

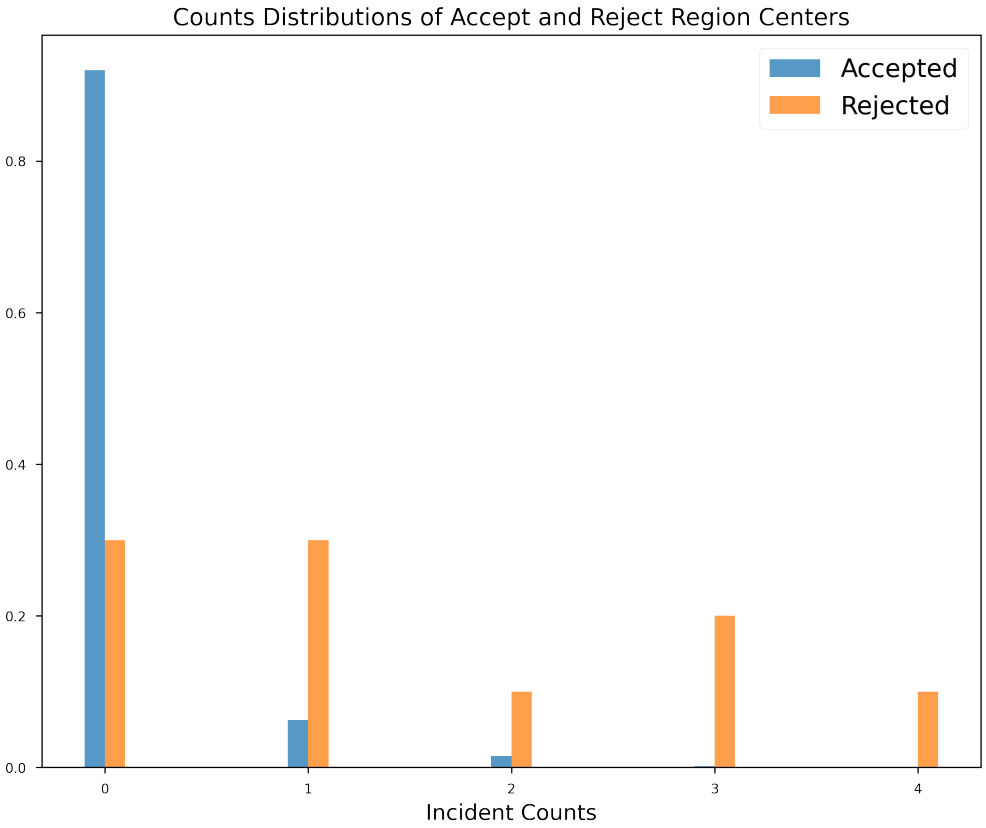

FIGURE 3.7. The case count distribution for accepted regions and rejected regions over 500 simulations.

CHAPTER 4

A Graph Limit-Based Pseudolikelihood for Fitting Exponential Random Graph Models

4.1. Introduction

Graph and network data are increasingly prevalent and accordingly many statistical methods have been developed in recent years. An entrance to the large literature of modeling network data can be found in [**42**]. A prominent model for network data is the *exponential random graph model* (ERGM), which defines the probability distribution over graphs G of a fixed size n

$$p(G;\boldsymbol{\beta}) = \exp\left(\langle \boldsymbol{\beta}, t(G)\rangle - \psi_n(\boldsymbol{\beta})\right),$$

where $\boldsymbol{\beta} \in \mathbb{R}^k$ are the model parameters, $t(G) = (t_1(G), \ldots, t_k(G))$ is a structure function of sufficient statistics of the graph G (e.g., number of triangles), $\langle \cdot, \cdot \rangle$ denoting the Euclidean inner product, and $\psi_n(\boldsymbol{\beta})$ is the log-partition function. Intuitively, a larger β_i increases the presence of the feature $t_i(G)$ in the graph (while interacting in a more complex way with the other structural features). The ERGM model contains many popular random graph models such as Markov random graphs and the well-known Erdös-Rényi (ER) model. The elegance of this model and its ability to select desirable features of a graph have contributed to its popularity in the network modeling community.[1]

Its popularity notwithstanding, estimating the parameters of this model is quite challenging. First, the partition function $Z(\boldsymbol{\beta})$ is generally unknown. This is due to the fact that the partition function is an expression that sums over all possible graphs of n nodes. For large enough n, this becomes completely intractable. Secondly, obtaining the sufficient statistics of the model becomes

[1]Early mathematical developments on ERGMs can be traced from the spatial statistics literature [**13**] to the initial extensions into social science with weak edge dependence [**57**] to generalizations with greater edge dependence, multiple edges between nodes, and edge values [**44**, **129**]. More recent developments focusing on model degeneracy can be found in [**52**, **111**].

prohibitively large as the size of the graph approaches millions of nodes. For instance, if the sufficient statistic is triangle counting, the complexity of the counting task is $O(n^3)$ (and in general, for k-node subgraphs the task is $O(n^k)$). Finally, different parameters $\boldsymbol{\beta}$ can give rise to almost identical distributions on graphs. For a trivial example, choosing edges and triangles as the terms and setting β_1 and β_2 both to any pair of values above a certain threshold, yields a distribution over graphs that is concentrated almost entirely on the complete graph.

Most approaches to the first issue fall in one of two classes: pseudolikelihood-based approaches or Markov chain Monte Carlo (MCMC) approaches. Strauss and Ikeda [**114**] proposed an early alternative to maximum likelihood estimation for networks [**44**] that relied on a simple pseudolikelihood function that ignores dyad dependency.[2] While computationally feasible and widely used for over a decade due to early promising results, it was shown to produce unreliable and biased estimates in various settings, such as the case of strong dependence [**122**]. MCMC approaches have accordingly received increased attention [**46**, **60**, **110**] in tandem with a focus on model degeneracy [**52**, **110**]. A number of software packages have become available for ERGMs including [**61**, **127**, **133**]. Recent theoretical progress by Chatterjee and Diaconis [**26**] has received some attention, leading to the development of a pseudolikelihood method showing promising results in certain parameter regimes [**54**]. The method is based on a large deviations approximation to the normalizing constant using graph limit theory developed in [**27**]. We will return to some of these methods in a later section to provide an exposition on their basics.

The second challenge to estimation presented above requires us to address the difficulty of graphlet counting. Graphlets are a small, induced subgraph of a larger graph, a concept that was first introduced in [**93**]. Independent from ERGMs, they have been used to fit certain network models [**15**], understand biological networks [**94**, **115**], and to create measures of similarity between networks [**107**]. A number of computationally efficient methods have been developed, including exact counting [**2**, **90**, **96**] and approximate counting [**14**, **24**, **51**, **86**, **126**]. The former tend to focus

[2]Dyad independence is a common modeling assumption first used in [**57**] which assumes that the only dependence between the edges of a directed graph is that between the two possible edges between each pair of nodes. This is a fairly restrictive assumption that reduces the random graph models to modeling the attraction and repulsion of the directed edges between each pair of nodes (i.e., friendship relations tend to be attractive, while power relations tend to be repulsive). This assumption was generalized to the much more permissive Markov dependence assumption in [**44**]). These were some of the earliest works the present author is aware of that described log-linear random graph models (i.e., models with the form $\text{logit}(p(\boldsymbol{x};\boldsymbol{\beta})) = \langle \boldsymbol{\beta}, \boldsymbol{x} \rangle$).

on efficient memory management and parallel architectures, while the latter center on developing unbiased and low variance sampling procedures, generally through MCMC procedures (with the exception of the recent [**24**]).

In this project, we study the feasibility of combining the pseudolikelihood method in [**26**] with subgraph estimation methods. To do this, we measure the robustness of the method to noise in the sufficient statistics, where the noise is meant to simulate the error in an MCMC subgraph estimation procedure. The characteristics of the noise will vary depending on the procedure and in general these characteristics are not yet well understood. Bounds on the variance are available, for example in [**86**], but these are dependent on the maximum degree of the graph and are not expected to be very tight. Therefore, for computational efficiency and for generality, we avoid specifying noise too precisely and use binomial noise to fuzz the counts. Using simulations, we find that the in the regime of low edge density the pseudolikelihood is relatively robust to noised counts. But before we dive into the details of the results, let us build some technical notation and context.

4.2. Mathematical Background

Let us build some notation. Let G denote a graph with n nodes and define $\mathcal{G}_n$ to be the space of all simple graphs on n labeled vertices.[3] Let $V(G), E(G)$ denote the sets of vertices and edges in G, respectively. The indices i, j will denote vertices as members of $V(G)$ and accordingly members of $E(G)$ will be pairs of nodes (i, j). A subgraph H in G will refer to an instance of a simple finite graph H in G, in the sense that there exists a function $f : H \to G$ that is edge preserving (i.e., if $(i, j) \in H$ then $(f(i), f(j)) \in G$). A graphlet H in G will refer to an instance of a simple finite induced subgraph H in G, in the sense that there exists a function $f : H \to G$ that is edge preserving in both directions (i.e., $(i, j) \in H$ if and only if $(f(i), f(j)) \in G$). Let $\hom(H, G)$ be the set of homomorphisms from H to G (a homomorphism is an edge preserving map between the set of vertices $V(H)$ to $V(G)$). $|\hom(H, G)|$ can be thought of as a subgraph count of H's in G, including permutations. Denote by G_{ij}^{+} the same graph G with the edge (i, j) present and denote by G_{ij}^{-} the same graph with the edge (i, j) missing (one of these is the same as the original graph). Furthermore, let G_{ij}^{c} denote the edge-information in G without the (i, j)-th edge.

[3] A simple graph is unweighted and undirected with no self-loops or multiple edges.

To avoid unnecessary technical overhead arising from continuous sample spaces, all the distributions in this chapter will be assumed to be over discrete sample spaces (with the exception of a normal distribution, in a single example) denoted by Ω. All the distributions will therefore have an associated probability mass function (pmf) $\Pr(X = k; \beta) = p(k; \beta)$, for a random variable X having the distribution p. Often, when there is no ambiguity, we will suppress the dependence on β and simply write $p(k)$. While the assumption of discreteness will reduce the generality of some of the stated results, many results carry over to continuous measure spaces (albeit with a substantial increase in technical overhead in their proof), so the interested reader should consult the references provided in each section for the statements in fuller generality.

4.2.1. Exponential Random Graph Models. Let us now revisit ERGMs with a precise definition. Define the loglikelihood of a graph

$$l(G; \boldsymbol{\beta}) \coloneqq \log p(G; \boldsymbol{\beta}) = n^2(\langle \boldsymbol{\beta}, \boldsymbol{t}(G) \rangle - \psi_n(\boldsymbol{\beta})). \tag{4.1}$$

The structure function used in the rest of this chapter will be a vector of the sufficient statistics $\boldsymbol{t}(G) = (t(H_1, G), \ldots, t(H_k, G))$ where

$$t(H_i, G) \coloneqq \frac{|\hom(H_i, G)|}{|V(G)|^{|V(H_i)|}}, \tag{4.2}$$

with the normalizing constant given by

$$\psi_n(\boldsymbol{\beta}) \coloneqq \frac{1}{n^2} \log \sum_{G \in \mathcal{G}} e^{n^2 \langle \boldsymbol{\beta}, \boldsymbol{t}(G) \rangle}. \tag{4.3}$$

$t(H_i, G)$ can be interpreted as the density of the subgraph H_i in G. In this chapter, we will usually assume that $t(H_1, G)$ is a constant multiple of the number of edges in the graph. Note that the parameters $\boldsymbol{\beta}$ here have been scaled in order for large n limits to exist. This parametrization will be referred to as *density parameters*.[4]

[4]We can convert to *natural parameters* $\boldsymbol{\eta}$, a parametrization commonly used in the literature and in sampling software, with the transformation

$$\eta_i = n^2 \frac{|\hom(H_i, H_i)|}{|V(G)|^{|V(H_i)|}} \beta_i.$$

Intuitively, the subgraph counts in the natural parametrization are up to isomorphism.

Another important feature of the model is that the conditional probability of the presence of an edge, given the rest of the graph is a very tractable quantity. This will be useful later for MCMC sampling. To see this, let the graph G be given. A simple calculation reveals that the log-odds ratio of edge presence versus absence equals

$$\log\left(\frac{p(G_{ij}^{+};\boldsymbol{\beta})}{p(G_{ij}^{-};\boldsymbol{\beta})}\right) = \langle \boldsymbol{\beta}, \boldsymbol{\Delta}_{ij} \rangle, \tag{4.4}$$

where $\boldsymbol{\Delta}_{ij} \coloneqq t(G_{ij}^{+}) - t(G_{ij}^{-})$ is the difference in the sufficient statistics vectors. A similar calculation reveals that the conditional probability of an edge's presence conditioned on the rest of the graph equals

$$\begin{aligned} p(G_{ij}^{+}|G_{ij}^{c};\boldsymbol{\beta}) &= \frac{p(G_{ij}^{+};\boldsymbol{\beta})}{p(G_{ij}^{+};\boldsymbol{\beta}) + p(G_{ij}^{-};\boldsymbol{\beta})} \\ &= \frac{1}{1 + \frac{p(G_{ij}^{-};\boldsymbol{\beta})}{p(G_{ij}^{+};\boldsymbol{\beta})}} \\ &= \text{logit}^{-1}(\langle \boldsymbol{\beta}, \boldsymbol{\Delta}_{ij} \rangle). \end{aligned}$$

Note that the quantity Δ_{ij} depends only on the differences in the sufficient statistics and not on the statistics themselves. Furthermore, the partition function has been cancelled out. These two facts remove the main computational barriers associated with the full distribution function and thus allow for efficient sampling.

4.2.2. Maximum entropy property of exponential families. Let us now introduce a key property of ERGMs, in order to motivate the functional form of the distribution. This key property lies in its maximization of a quantity known as entropy.

Entropy is a concept with a long history and deep connections to many fields (an entrance to this field can be found in the classic [**34**]). For the sake of brevity of exposition here, we will rely on just a few key observations. Given a distribution $p(k)$ over a discrete sample space Ω, entropy is defined as the following quantity

$$H(p) \coloneqq -\sum_{k \in \Omega} p(k) \log p(k).$$

We define the entropy $H(X)$ of a random variable X with distribution p to be $H(p)$. This quantity has a number of properties: it is always non-negative, it is maximized on a uniform distribution (i.e., $H(p) = \log m$ for a uniform distribution over a sample space of size m), and it is minimized for Dirac delta distributions (i.e., $H(p) = 0$ if $p(k) = 1$ for some $k \in \Omega$). With these two cases in mind, we can intuitively see that entropy measures the "spread" of a distribution versus its concentration on a few elements. If a distribution concentrates, then it becomes more predictable, and, on the other hand, the spread of a distribution leads to more unpredictability. Therefore entropy increases as the "randomness" of a distribution increases.

The maximum entropy principle is one proposed by Jaynes in 1950s [**62**] as a principled way of encoding one's knowledge and ignorance in a probability distribution, in addition to attempting to unify the concepts of entropy in statistical mechanics and in information theory. Namely, the maximum entropy principle states that given some information constraints (such as data), the most impartial distribution P that satisfies the data is one that also maximizes the Shannon entropy. It has since been applied in many fields as a distribution fitting method [**65**]. While there is a rich literature on the soundness of the principle which borders on philosophy [**63**], we will not go into it here. Let us instead define the precise mathematical problem of maximum entropy.

Given a function $\boldsymbol{\phi} : \Omega \to \mathbb{R}^d$ and a vector $\boldsymbol{\alpha} \in \mathbb{R}^d$, the *maximum entropy problem* is to solve

$$\max_p H(p)$$
$$\text{subject to } \mathbb{E}_p[\boldsymbol{\phi}(X)] = \boldsymbol{\alpha},$$

where X is a random variable with distribution p, $\mathbb{E}_p[\boldsymbol{\phi}(X)] = \sum_{k\in\Omega} \boldsymbol{\phi}(k)p(k)$ is the expectation of $\boldsymbol{\phi}(X)$ over p, and the maximization is taken over all distributions p on Ω. Rewriting, we can see

that the problem is equivalent to

$$\begin{aligned} \max_p \sum_{k\in\Omega} p(k)\log p(k) \\ \text{subject to } \sum_{k\in\Omega} p(k)\phi_i(k) = \alpha_i, \\ \forall k \in \Omega, p(k) \geq 0, \\ \sum_{k\in\Omega} p(k) = 1. \end{aligned}$$

Now that we have defined entropy, which we will show that ERGMs maximize (with some constraints), let us introduce another concept which will help us prove our propositions. The particular property of ERGMs that we're interested in is in fact a consequence of a more general feature of the model and that is the fact that ERGMs belong to the exponential family. In defining an exponential family, let $\boldsymbol{\beta} \in \mathbb{R}^d$ and let x denote a general object from the family Ω over which we have the distribution

$$\begin{aligned} p(x;\boldsymbol{\beta}) &= \exp(\langle \boldsymbol{\beta}, \boldsymbol{t}(x)\rangle - \psi(\boldsymbol{\beta})) \\ \psi(\boldsymbol{\beta}) &= \log\left(\sum_{k\in\Omega} \exp(\langle \boldsymbol{\beta}, \boldsymbol{t}(x)\rangle)\right). \end{aligned} \tag{4.5}$$

It is easy to see that ERGMs already have this form. Let us demonstrate some other common distributions in this family.

Example 4.2.1 (Binomial distribution). The *binomial distribution* with parameters $n \in \mathbb{N}$ and $p \in [0,1]$ is defined as the pmf

$$p(k;n,p) = p^k(1-p)^{n-k}.$$

With a little algebra and by setting $\beta = \log\frac{p}{1-p}$, we can show that

$$p(k;n,\beta) = \exp(k\beta - n\log(1+e^\beta)).$$

Example 4.2.2 (Poisson distribution). The *Poisson distribution* with parameter $\lambda > 0$ is defined as the pmf

$$p(k;\lambda) = e^{-\lambda}\frac{\lambda^k}{k!}.$$

With some algebra and by setting $\beta = \log \lambda$, we can work this into the exponential family form

$$p(k; \beta) = \exp(k\beta - \log k! - e^{\beta}).$$

EXAMPLE 4.2.3 (Normal distribution). The *normal distribution* $N(\boldsymbol{\mu}, \Sigma)$, with $\boldsymbol{\mu} \in \mathbb{R}^k$ and $\Sigma \in \mathbb{R}^{k \times k}$, is typically defined as

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det(\Sigma)}} \exp\left(-\frac{1}{2}\langle(\boldsymbol{x} - \boldsymbol{\mu}), \Sigma^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\rangle\right).$$

This can be reparametrized to

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) \propto \exp\left(\langle \Sigma^{-1}\boldsymbol{\mu}, \boldsymbol{x}\rangle + \frac{1}{2}\langle \boldsymbol{x}\boldsymbol{x}^T, \Sigma^{-1}\rangle\right),$$

which illuminates the exponential family form.

EXAMPLE 4.2.4 (Erdös-Rényi model). Recall that an ER model $G(n, p)$, with n nodes and edge probability p, assigns to an E-edge graph G the probability

$$p(G; p) = p^E (1 - p)^{N - E},$$

where $N = \binom{n}{2}$ is the maximum possible number of edges in G. As the binomial case shows, setting $\beta = \operatorname{logit}(p)$ gives

$$p(G; \beta) = \exp(\beta E - N \log(1 + e^{\beta})),$$

the exponential family form we seek.

Two very important properties of the exponential family are that they naturally arise as solutions in optimization problems and that they are analytically tractable, leading to a long history of study.

Let us prove a couple key properties of exponential families.

PROPOSITION 4.2.1. *The log-partition function $\psi(\boldsymbol{\beta})$, $\boldsymbol{\beta} \in \mathbb{R}^d$, of an exponential family is convex as a function $\boldsymbol{\beta} \to \psi(\boldsymbol{\beta})$.*

PROOF. Let us recall that a convex function $f(x)$ on a convex domain Ω is one where $f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda) f(y)$ for all $x, y \in \Omega$ and $\lambda \in [0, 1]$. Let $\boldsymbol{\beta}_\lambda = \lambda \boldsymbol{\beta}_1 + (1 - \lambda)\boldsymbol{\beta}_2$, with

$\boldsymbol{\beta}_1, \boldsymbol{\beta}_2 \in \mathbb{R}^d$. Then $\frac{1}{\lambda} \geq 1$ and $\frac{1}{1-\lambda} \geq 1$ and so by Hölder's inequality[5] we have

$$\begin{aligned}
\log \sum_{k \in \Omega} \exp(\langle \boldsymbol{\beta}_\lambda, T(k) \rangle) &= \log \sum_{k \in \Omega} \exp(\langle \boldsymbol{\beta}_1, T(k) \rangle)^\lambda \exp(\langle \boldsymbol{\beta}_2, T(k) \rangle)^{1-\lambda} \\
&\leq \log \left(\sum_{k \in \Omega} \exp(\langle \boldsymbol{\beta}_1, T(k) \rangle) \right)^\lambda \left(\sum_{k \in \Omega} \exp(\langle \boldsymbol{\beta}_2, T(k) \rangle) \right)^{1-\lambda} \\
&= \lambda \log \left(\sum_{k \in \Omega} \exp(\langle \boldsymbol{\beta}_1, T(k) \rangle) \right) + (1-\lambda) \left(\sum_{k \in \Omega} \exp(\langle \boldsymbol{\beta}_2, T(k) \rangle) \right),
\end{aligned}$$

which shows convexity. □

Note that this convexity substantially reduces the complexity of maximum likelihood estimation for exponential families. For instance, suppose we have a sample $X_1, \ldots, X_n$ from an exponential family and we wish to estimate $\boldsymbol{\beta}$ by maximum likelihood, then we would need to solve

$$\max_{\boldsymbol{\beta}} \sum_{i=1}^n \log p(X_i; \boldsymbol{\beta}) = \sum_{i=1}^n \left(\langle \boldsymbol{\beta}, \boldsymbol{\phi}(X_i) \rangle + \psi(\boldsymbol{\beta}) \right)$$

which is a convex problem in $\boldsymbol{\beta}$. Furthermore, though we do not prove it here, $\psi(\boldsymbol{\beta})$ can be shown to be a smooth (infinitely differentiable) function in $\boldsymbol{\beta}$. This fact implies that there are no local minima and the whole suite of tractable convex optimization algorithms can be brought to bear on these problems [**19**] (as long as the partition function remains tractable, of course). For a deeper exploration of the properties of exponential families see [**125**]. For connections between exponential families and information geometry see [**4**].

Returning to the maximum entropy problem stated above, we can now state the following theorem.

Proposition 4.2.2. *Let*

$$\Delta_{\boldsymbol{\alpha}} \coloneqq \{p \mid \mathbb{E}_p[\boldsymbol{\phi}(X)] = \boldsymbol{\alpha}\}$$

be the set of all distributions over Ω satisfying the expectation constraint $\mathbb{E}[\boldsymbol{\phi}(X)] = \boldsymbol{\alpha}$.

For $\boldsymbol{\beta} \in \mathbb{R}^d$, let $p_{\boldsymbol{\beta}}$ have the exponential family form as in Equation (4.5). If $\mathbb{E}_{p_{\boldsymbol{\beta}}}[\boldsymbol{\phi}(X)] = \boldsymbol{\alpha}$, then $p_{\boldsymbol{\beta}}$ maximizes $H(p)$ over $\Delta_{\boldsymbol{\alpha}}$; furthermore, the distribution $p_{\boldsymbol{\beta}}$ is unique.

[5] See Chapter 2 in [**72**].

PROOF. We proceed by setting up a Lagrangian for the problem.[6] Introducing the Lagrange multipliers $\lambda(k) \geq 0$ for the constraint $p(k) \geq 0$, $\beta_0 \in \mathbb{R}$ for the normalization constraint, and $\boldsymbol{\beta}_i$ for the constraints that $E_p[\phi_i(X)] = \alpha_i$, we obtain

$$\begin{aligned}\mathcal{L}(p, \boldsymbol{\beta}, \beta_0, \lambda) = \sum_{k\in\Omega} p(k) \log p(k) + \sum_{i=1}^{d} \beta_i \left(\alpha_i - \sum_{k\in\Omega} p(k)\phi_i(k) \right) \\ + \beta_0 \left(1 - \sum_{k\in\Omega} p(k) \right) - \left(\sum_{k\in\Omega} \lambda(k)p(k) \right).\end{aligned}$$

Taking derivatives, we obtain

$$\frac{\partial}{\partial p(k)} \mathcal{L}(p, \boldsymbol{\beta}, \beta_0, \lambda) = 1 + \log p(k) - \sum_{i=1}^{d} \beta_i \phi_i(k) + \beta_0 - \lambda(k) = 1 + \log p(k) - \langle \boldsymbol{\beta}, \boldsymbol{\phi}(k) \rangle + \beta_0 - \lambda(k).$$

Setting this to zero and solving for $p(k)$, we obtain

$$p(k) = \exp(\langle \boldsymbol{\beta}, \boldsymbol{\phi}(k) \rangle - 1 - \beta_0 - \lambda(k)).$$

Note that with this setting, $p(x) > 0$, thus the constraint $p(x) \geq 0$ is unnecessary and complementary slackness implies $\lambda(k) = 0$. Finally, the normalization constraint implies that $\beta_0 = -1 + A(\boldsymbol{\beta})$, which shows that the optimal distribution has the desired form

$$p(k) = \exp(\langle \boldsymbol{\beta}, \boldsymbol{\phi}(k) \rangle - A(\boldsymbol{\beta})).$$

Now to show uniqueness, let us label the distribution above as $p_{\boldsymbol{\beta}}$ and consider another distribution $p \in \Delta_{\boldsymbol{\alpha}}$. Then we have

$$\begin{aligned}H(p) &= -\sum_{k\in\Omega} p(k) \log p(k) = -\sum_{k\in\Omega} p(k) \log \frac{p(k)}{p_{\boldsymbol{\beta}}(k)} + \sum_{k\in\Omega} p(k) \log p_{\boldsymbol{\beta}}(k) \\ &= -D_{KL}(p \| p_{\boldsymbol{\beta}}) + \sum_{k\in\Omega} p(k)(\langle \boldsymbol{\beta}, \boldsymbol{\phi}(k) \rangle - A(\boldsymbol{\beta})) \\ &= -D_{KL}(p \| p_{\boldsymbol{\beta}}) + \sum_{k\in\Omega} p_{\boldsymbol{\beta}}(k)(\langle \boldsymbol{\beta}, \boldsymbol{\phi}(k) \rangle - A(\boldsymbol{\beta})) \\ &= -D_{KL}(p \| p_{\boldsymbol{\beta}}) + H(p_{\boldsymbol{\beta}}),\end{aligned}$$

[6] See Chapter 5 [**19**] for details on the method.

where in line 3 we used the fact that both distributions have the same expectation $\boldsymbol{\alpha}$ against $\boldsymbol{\phi}(X)$. The quantity D_{KL} is known as the Kullback-Leibler divergence and it is known to be strictly positive unless $p = p_{\boldsymbol{\beta}}$, hence we have shown that $p_{\boldsymbol{\beta}}$ is unique.[7] □

4.2.3. Markov chain Monte Carlo Approaches to ERGMs. The functional form of ERGMs allows for graphs to be sampled from the models through Markov chain Monte Carlo (MCMC) methods. MCMCs are well-studied and extensively used to simulate networks [**47**, **53**]. As is typical of MCMC methods, the goal is to construct a Markov chain on the sample space Ω with $p(k; \boldsymbol{\beta})$ as the equilibrium distribution. This is done by starting from an arbitrary sample and making a large number of appropriate Markov transitions until an approximate distributional convergence is reached (additional transitions can then be made to obtain additional samples). Many Markov chains can be used for a given model, with the choice of chain affecting their mixing properties. A basic result in Markov chain theory shows that convergence is guaranteed under mild conditions (irreducibility and aperiodicity) in the limit of infinitely many transitions. For ERGMs, this convergence has been studied by [**110**]. There are two common methods to produce such a Markov chain: Gibbs sampling and the Metropolis algorithm.

Gibbs sampling:

- At the start of every iteration select an edge to turn on or off at random, say the edge (i, j).
- Set the edge on or off, based on the conditional distribution $p(G_{ij}^{+}|G_{ij}^{c}; \boldsymbol{\beta})$, as already calculated in Equation (4.4).

Metropolis-Hastings algorithms:

- Propose a transition at iteration t from G^t to G^{t+1} based on an auxiliary distribution $q(G^t, G^{t+1})$.
- Make the transition to the proposed graph G^{t+1} with probability

$$\pi = \min\{1, \frac{p(G^{t+1}; \boldsymbol{\beta})}{p(G^t; \boldsymbol{\beta})} \frac{q(G^t, G^{t+1})}{q(G^{t+1}, G^t)}\}.$$

[7] See Chapter 2 of [**34**] for properties of the Kullback-Leibler divergence.

The *Metropolis algorithm* is the special case in which the auxiliary distribution is symmetric $q(x, y) = q(y, x)$. Different choices of $q(\cdot, \cdot)$ can be chosen to focus the transitions of the Markov chain. Skillful choice of q can lead to convergence more efficient than Gibbs sampling. Note still, that the calculation of π can be done efficiently by using Equation (4.4).

In this work we rely on the R software package `ergm` to produce ERGM samples, which includes many other MCMC sampling schemes [**61**]. This package also implements a number of fitting methods, including the common maximum likelihood estimation method based on [**46**]. Let us briefly sketch this method.

Recall the exponential family form as defined in Equation (4.5). Define the loglikelihood

$$l(x; \boldsymbol{\beta}) \coloneqq \langle \boldsymbol{\beta}, \boldsymbol{t}(x) \rangle - \psi(\boldsymbol{\beta}).$$

Since a direct maximum likelihood approach on the problem would run into difficulties calculating the normalizing constant, let us make a small variation on the problem by choosing an initial starting point $\boldsymbol{\beta}_0$ and taking ratios. The resulting problem can be transformed as follows

$$\begin{aligned}
\max_{\boldsymbol{\beta}} l(x; \boldsymbol{\beta}) - l(x; \boldsymbol{\beta}_0) &= \max_{\boldsymbol{\beta}} \langle \boldsymbol{\beta} - \boldsymbol{\beta}_0, \boldsymbol{t}(x) \rangle - \psi(\boldsymbol{\beta}) + \psi(\boldsymbol{\beta}_0) \\
&= \max_{\boldsymbol{\beta}} \langle \boldsymbol{\beta} - \boldsymbol{\beta}_0, \boldsymbol{t}(x) \rangle - \log \left(\mathbb{E}_{\boldsymbol{\beta}_0} [\exp(\langle \boldsymbol{\beta} - \boldsymbol{\beta}_0, \boldsymbol{t}(x) \rangle)] \right) \\
&\approx \max_{\boldsymbol{\beta}} \langle \boldsymbol{\beta} - \boldsymbol{\beta}_0, \boldsymbol{t}(x) \rangle - \log \left(\frac{1}{M} \sum_{i=1}^{M} \exp(\langle \boldsymbol{\beta} - \boldsymbol{\beta}_0, t(x_i) \rangle \right),
\end{aligned}$$

where the second line follows through a little algebra[8] and the third line relies on the law of large numbers to approximate the expectation with a series of samples from p_{β_0}. It is important to be aware that the choice of β_0 is critical and can cause the optimization to fail.

4.2.4. Scalable Graphlet Counting via Lifting. Here we will briefly describe the graphlet sampling procedure from our work in [**86**].

[8]This can also be seen as a result of the well-known formula for the moment-generating function of the canonical statistic $\boldsymbol{t}(X)$ induced by $p_{\boldsymbol{\beta}}$

$$\log m_{\boldsymbol{\beta}}(t) = \log \mathbb{E}_{\boldsymbol{\beta}} [\exp(\langle \boldsymbol{\beta}, \boldsymbol{t}(X) \rangle)] = \psi(t + \boldsymbol{\beta}) - \psi(\boldsymbol{\beta}).$$

See the notes [**45**] by Geyer.

The vertex neighborhood of S will be denoted by $N_v(S)$, defined as the set of all vertices adjacent to S but not in S. The edge neighborhood of S will be denoted by $N_e(S)$, defined as the set of all edges connecting a vertex in S with a vertex outside S. By $G|\{v_1, \ldots, v_k\}$ we will denote the induced subgraph in G formed by selecting the nodes $\{v_1, \ldots, v_k\} \subset V(G)$. Let $V(H, G)$ be the set of all induced subgraphs in G that are isomorphic to H. Furthermore, let $V_k(G) = \cup_{i=1}^{l} V(H_i, G)$ where $H_1, H_2, \ldots, H_l$ are all non-isomorphic connected graphlets of size k. The ideal sampling procedure would sample graphlets uniformly randomly from the set $V_k(G)$. For $T \in V_k(G)$, we will say that it is of "type m" if it is isomorphic to H_m and we will denote this by $T \sim H_m$. Unfortunately, this is a quite difficult task, since random subsets of nodes in G are unlikely to be connected. Therefore a more elaborate Markov chain procedure will be required. First however, let us describe how we would estimate the graphlet counts given a sequence of sampled graphlets $T_1, T_2, \ldots, T_m$. Applying Horvitz-Thompson weighting, we arrive at the estimator

$$\hat{N}_m(G) \coloneqq \frac{1}{m} \sum_{i=1}^{m} \frac{\mathbf{1}(T_i \sim H_m)}{\pi(T_i)}.$$

This estimator is clearly unbiased.

A naïve method for sampling graphlets would be to perform a random walk on the graph G and sample the most recent k vertices. This scheme would have an easy-to-compute stationary distribution and would also 'inherit' the relatively fast mixing rate from the random walk on G. Despite the simplicity, the method would never sample certain graphlets such as stars, so modifications are necessary to fix this.

The lifting procedure is based on a random protocol that attaches a vertex to a given graphlet. First we select a node at random, either uniformly or through a random walk. Then for any $(k-1)$-graphlet, S, we lift it to a k-graphlet by adding a vertex from its neighborhood, $N_v(S)$, at random. Note that this procedure can sample any possible graphlet in $V_k(G)$. We can then express the probability of sampling a k-graphlet T recursively as follows

$$\pi_U(T) = \sum_{S \subset T} \pi_U(S) \frac{\deg_S(V_T \setminus V_S)}{|N_e(S)|} = \sum_{S \subset T} \pi_U(S) \frac{|E_T| - |E_S|}{\sum_{u \in S} \deg(u) - 2|E_S|},$$

where the sum is taken over all connected $(k-1)$ graphlets $S \subset T$. The probability of a single node $\pi(v)$, as mentioned above, could then be either a uniform probability over nodes or a polynomial

function of the degree of v (in accordance with the stationary distribution of a random walk on a graph), depending on the initialization method chosen.

Now consider a sampled k-graphlet $T \coloneqq S_k$. Denote the set of possible sequences $A = [v_1, \ldots, v_k]$ that would form T in the lifting process as $\mathrm{co}(T)$. Note that at every step of the sequence, the induced subgraph must be connected, thus $\mathrm{co}(T) = \{[v_1, \ldots, v_k] \in V(G)^k \mid \{v_1, \ldots, v_k\} = V_T, T|\{v_1, \ldots, v_r\} \text{ is connected}\}$. Then

$$\pi_U(T) = \sum_{A \in \mathrm{co}(T)} \pi(A[1]) \prod_{r=1}^{k-1} \frac{|E_{S_{r+1}(A)}| - |E_{S_r(A)}|}{\sum_{i=1}^{r} \deg(A[i]) - 2|E_{S_r(A)}|},$$

where $A[i]$ is the ith vertex in A and $S_r(A) = G|\{v_1, \ldots, v_r\}$. Note that $\mathrm{co}(T)$ depends on the isomorphism class of T and can be precomputed. Furthermore, note that the probability function $\pi_U(T)$ is a function of the degrees of the vertices involved. Hence, letting $[v_1, \ldots, v_k]$ be an arbitrary labeling of the vertices of T with $d_i = \deg(v_i)$, the function

$$\pi_U(T) = \frac{1}{K} F_m(d_1, \ldots, d_k)$$

can be pre-computed and cached for every graphlet isomorphism class.

EXAMPLE 4.2.5. A triangle is a 3-graphlet with edges $(v_1, v_2), (v_2, v_3), (v_3, v_1)$. Given the degrees d_1, d_2, d_3 of sampled vertices in a graph, the probability function for sampling the triangle with these vertices is

$$\begin{aligned}
\pi_U(\text{triangle}) = &\left(\frac{\pi_1(d_1)}{d_1} + \frac{\pi_1(d_2)}{d_2}\right) \frac{2}{d_1 + d_2 - 2} \\
&+ \left(\frac{\pi_1(d_2)}{d_2} + \frac{\pi_1(d_3)}{d_3}\right) \frac{2}{d_2 + d_3 - 2} \\
&+ \left(\frac{\pi_1(d_3)}{d_3} + \frac{\pi_1(d_1)}{d_1}\right) \frac{2}{d_3 + d_1 - 2}
\end{aligned}$$

(recall that $\pi_1(d_i)$ depends on the initial vertex sampling scheme).

4.2.5. Graph Limit Theory. Graph limit theory was introduced and developed by Lovász and others in [**18**, **75**]. In this section, we develop the basics of the theory to state key large deviation results for ER graphs.

A sequence of simple graphs $\{G_n\}$ is said to be convergent if the sequence $t(H, G_n)$ exists for all simple graph H. In this sense, the density of the subgraphs in the G_n limit to the same value as n tends to infinity. A key result of [**75**] is that the limit object of convergent graph sequences can be represented as a measurable function. Let $\mathcal{W}$ be the set of all measurable functions $w : [0,1]^2 \to [0,1]$ with the symmetry $w(x,y) = w(y,x)$ for all $x, y \in [0,1]$. Finite graphs G can be naturally embedded in $\mathcal{W}$ by interpreting the adjacency matrix of G as a piece-wise function on $[0,1]^2$. For example, one such possible identification is to split the interval $[0,1]$ into equal segments of size $1/n$ and defining $w_G(x,y) = \mathbf{1}((\lceil nx \rceil, \lceil ny \rceil) \in G)$.

For every simple subgraph H and $w \in \mathcal{W}$, define the density of H in w to be[9]

$$t(H,w) \coloneqq \int_{[0,1]^{|V(H)|}} \prod_{ij \in E(H)} w(x_i, x_j) d\boldsymbol{x}.$$

We say that G_n converges to w if $\lim_{n\to\infty} t(H, G_n) = d(H, w)$ for every finite simple graph H.

The cut-distance pseudometric[10] can be defined on $\mathcal{W}$

$$\delta_\square(f,g) \coloneqq \inf_\sigma \sup_{S,T \subset [0,1]} \left| \int_{S\times T} [f(\sigma x, \sigma y) - g(x,y)] dx dy \right|,$$

where the infimum is taken over all measure-preserving bijections $\sigma : [0,1] \to [0,1]$ and the supremum is taken over all measurable subsets S, T of $[0,1]$. This pseudometric measures the maximum difference between the integrals of two graphons over measurable boxes and then minimizes the difference over all possible "relabelings" of the graphon's vertices. To make this pseudometric into a metric, we must quotient the space $\mathcal{W}$ by identifying all graphs that are weakly isomorphic (i.e., $\delta_\square(f,g) = 0$). Denote the resulting quotient space by $\tilde{\mathcal{W}}$. Three fundamental results on graphons:

(1) every graphon is the $\delta_\square$-limit of a sequence of finite graphs,

(2) the space $(\tilde{\mathcal{W}}, \delta_\square)$ is compact,

(3) for every finite simple graph H, the map $t(H, \cdot) : \mathcal{W} \to [0,1]$ is Lipschitz continuous.

[9]This formula can be seen as an integral limit of the subgraph density

$$t(H,G) = \frac{1}{|V(G)|^{|V(H)|}} \sum_{\boldsymbol{x} \in V(G)^{|V(H)|}} \prod_{(i,j)\in E(H)} \mathbf{1}((x_i, x_j) \in G)$$

for finite simple graphs H, G.

[10]A pseudometric is a metric $d(\cdot,\cdot)$ where $d(x,y) = 0$ does not imply that $x = y$.

As we do not wish to distract from the chapter's narrative, we leave the interest reader to consult [**74**] for details.

4.2.6. Large Deviations Principles for ERGMs. In [**28**], the authors formulate a large deviation principle for the ER random graph $G(n,p)$. First, define the function $I_p : [0,1] \to \mathbb{R}$ to be the function

$$I_p(u) \coloneqq \frac{1}{2}u\log\frac{u}{p} + \frac{1}{2}(1-u)\log\frac{1-u}{1-p}.$$

This function's domain can be extended to $\mathcal{W}$ via

$$I_p(h) \coloneqq \int_{[0,1]^2} I_p(h(x,y))dxdy$$

and then naturally extended to $\tilde{\mathcal{W}}$ by identifying the value of the integral with the integral $I_p(h)$ for a member of an equivalence class (this was shown to be well-defined). Now the ER random graph $G(n,p)$ induces the distributions $P(\cdot;n,p)$ on $\mathcal{W}$ through the map $G \to w_G$ and in turn on $\tilde{\mathcal{W}}$ through the natural injection of $\mathcal{W}$ into $\tilde{\mathcal{W}}$. This distribution was shown to obey the following large deviation principle.

THEOREM 4.2.1 (Chatterjee and Varadhan [**28**]). *For each fixed $p \in (0,1)$, the sequence $P(\cdot;n,p)$ obeys a large deviation principle with rate function I_p. Namely, for any closed set $S \subset \tilde{\mathcal{W}}$*

$$\limsup_{n\to\infty} \frac{1}{n^2}\log P(S;n,p) \leq -\inf_{w\in S} I_p(w)$$

and for any open set $T \subset \tilde{\mathcal{W}}$

$$\liminf_{n\to\infty} \frac{1}{n^2}\log P(T;n,p) \geq -\inf_{w\in S} I_p(w).$$

This result is extended to ERGM models in [**26**] (see Theorem 3.1). A consequence of this is the following theorem

THEOREM 4.2.2 (Theorem 4.1 in [**26**]). *For $\beta_1,\ldots,\beta_k$, $H_1,\ldots,H_k$, and ψ_n as above,*

$$\lim_{n\to\infty}\psi_n = \sup_{0\leq u\leq 1}\left(\sum_{i=1}^{k}\beta_i u^{e(H_i)} + \frac{1}{2}I(u)\right),$$

where $I(u) \coloneqq -u \log u - (1-u)\log(1-u)$ *is the binary entropy function and* $e(H_i)$ *is the number of edges in* H_i.

For example, consider the following model on a graph with n vertices

$$p(G; \beta_1, \beta_2) = \exp\left(2\beta_1 E + \frac{6\beta_2}{n}\Delta - n^2 \psi_n(\beta_1, \beta_2)\right)$$

where E, Δ denote the number of edges and triangles in the graph G (this is the model Equation (4.2) with H_1 being an edge and H_2 being a triangle). The normalization allows us to take non-trivial model limits (without scaling, for large n, almost all graphs are either completely empty or full). Theorem 4.2.2 states that for large n

$$\psi_n \simeq \sup_{0 \le u \le 1} \left(\beta_1 u + \beta_2 u^3 - \frac{1}{2} u \log u - \frac{1}{2}(1-u)\log(1-u)\right).$$

We leverage the asymptotic relationship above to produce a pseudolikelihood. Labeling the right-hand side of Theorem 4.2.2 by $\hat{\psi}_n$ we obtain

$$\hat{l}(G; \boldsymbol{\beta}) \coloneqq n^2(\langle \boldsymbol{\beta}, t(G)\rangle - \hat{\psi}_n(\boldsymbol{\beta})). \tag{4.6}$$

4.3. Results

For our simulation studies, we used an ERGM model where H_1 is an edge, H_2 is a two-star, and H_3 is a triangle. We specify a few true values of the parameters to $\boldsymbol{\beta}_1 = (-1, 0.4, 0.1)$, $\boldsymbol{\beta}_2 = (-0.5, 0.1, 0.5)$, and $\boldsymbol{\beta}_3 = (-0.5, 0.1, 0.5)$, which were chosen to get a range of parameter values across graph density (low to high, respectively). We used the R function `simulate.ergm` from [**61**] to generate ERGM graphs of sizes ($n = 50, 100, 200, 500$). For each size instance, we generated 5 graphs, counted the subgraphs with the `ergm` package, and for each graph produced 5 noised counts $\tilde{t}(G)$ (while the sample sizes may appear low, the size of the graphs used ensures that with a high probability will have similar subgraph statistics). The noised counts $\tilde{t}_i(G)$ are obtained as samples from a scaled binomial distribution $\frac{1}{p}\text{Bino}(t_i(G), p)$. We use a range of noised values $p \in \{1, 8/10, 1/2\}$ (which correspond to standard deviations of size $0, \sqrt{t_i(G)}/2, \sqrt{t_i(G)}$). We then applied the inference method Equation (4.6) to these counts.

We measured the performance of the approaches in terms of the average relative errors of the subgraph statistics between the inferred and the graph used for fitting. More precisely, for each graph that was noised for fitting, we obtain an estimate of the average subgraph counts from the resulting fitted betas $\hat{\boldsymbol{\beta}}$ by simulating 5 graphs from the model with the fitted parameters and then find the average absolute difference between this estimated average and the counts of the unnoised graph used for fitting. This method of comparison avoids errors in goodness of fit testing arising from identifiability issues in the model as well as noise issues arising from sampling. Note that in addition to comparing the subgraph counts of the in-model subgraphs H_1, H_2, H_3, we also compare the fit on the out-of-model 4-cycle H_4 and 3-star H_5 subgraphs (as suggested in [**59**]).

The results can be seen in Table 4.1, Table 4.2, Table 4.3. In the low density regime, the error rate is overall quite good. The effects of the noise do not appear to be not large. While intuition might suggest that high sparsity data should be more heavily effected by noise (as the signal is easily covered), the effects of the ERGM model approximation dominate (that is, the ERGM model is likely very close to an ER model, which is robust to counts perturbations). The out-of-model 4-cycle and the in-model triangle subgraphs have the worst performance, though they converge quite quickly as the number of nodes increases. This is likely due to the convergence of the normalizing constant approximation. Interestingly, the 2-star, which presents a subgraph of the triangle, has a factor of two improved performance. It is not entirely clear why this occurs.

In the medium density regime, we can see a marked change in the baseline error rates, with all subgraphs having an error rate of almost 10 percent even before noising. The edge counts appear to be relatively well reproduced, even as the noise and graph size increases. The rest of the subgraphs increasing the size of the graph appears to either keep the error rate constant or magnify it, which is unexpected. This may also be due to a larger variance in the graph samples than expected or due to a poor approximation of normalizing constant in this regime. Investigating further reveals that the variance in the edge density is on the order of 10 percent. This suggests that the model has not converged to an ER model with a fixed edge density yet, leading to a poor approximation in the normalizing constant.

Finally, in the high density region, we should expect to see similar performance to the medium regime. Indeed we do, with error rates beginning on the order of 50 percent. However, there is a

marked improvement as the size of the graph increases, which is puzzling. This could be due to a fast convergence available to this parameter choice than it is available to the medium density parameters. In this high density regime, the noise does not seem to affect the error rate as much as in the medium or low density regimes. This may be due to greater variability in the model, which is able to subsume the noised subgraph counts.

Relative error for $\boldsymbol{\beta}_1 = (-1, 0.4, 0.1)$ with no subgraph noising					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.0340	0.0657	0.115	0.129	0.094
100	0.0221	0.0407	0.0543	0.0727	0.0562
200	0.0144	0.0293	0.0437	0.0589	0.0444
500	0.00745	0.01490	0.02140	0.02980	0.02250
Relative error for $\boldsymbol{\beta}_1 = (-1, 0.4, 0.1)$ with $p = 8/10$					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.0372	0.0702	0.1290	0.1430	0.1130
100	0.0244	0.0479	0.0806	0.0949	0.0728
200	0.0158	0.0309	0.0435	0.0600	0.0458
500	0.00881	0.01740	0.02640	0.03450	0.02580
Relative error for $\boldsymbol{\beta}_1 = (-1, 0.4, 0.1)$ with $p = 1/2$					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.0404	0.0673	0.0983	0.1550	0.1110
100	0.0226	0.0452	0.0800	0.0968	0.0696
200	0.0195	0.0388	0.0585	0.0782	0.0593
500	0.00749	0.01470	0.02310	0.02880	0.02160
True count density	0.146	0.0216	0.00319	0.000468	0.00318

TABLE 4.1

4.4. Conclusion

Overall, our analysis suggests that regimes of good approximation for the ERGM model will be robust to noisy counts. While the graph limit approximation does not always yield accurate estimates of the ERGM parameters, there seems to be little degradation in performance when we use noisy graphlet counts in regimes with low edge density. Since this is a common feature in real-world social networks, this means the method may have utility in fitting real-world social networks. Furthermore, if applied to social networks on a size far greater than what was tested here (tens of thousands of nodes), the asymptotics of the normalizing constant may behave better

Relative error for $\boldsymbol{\beta}_2 = (-0.5, 0.1, 0.5)$ with no subgraph noising					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.0968	0.1820	0.2520	0.3240	0.2580
100	0.0627	0.1210	0.1750	0.2220	0.1750
200	0.0537	0.1030	0.1460	0.1910	0.1480
500	0.138	0.256	0.356	0.444	0.357
Relative error for $\boldsymbol{\beta}_2 = (-0.5, 0.1, 0.5)$ with $p = 8/10$					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.0742	0.1380	0.1930	0.2470	0.1960
100	0.0536	0.1010	0.1430	0.1820	0.1440
200	0.0797	0.1500	0.2120	0.2670	0.2120
500	0.121	0.223	0.311	0.387	0.311
Relative error for $\boldsymbol{\beta}_2 = (-0.5, 0.1, 0.5)$ with $p = 1/2$					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.112	0.199	0.270	0.326	0.268
100	0.0704	0.1340	0.1890	0.2410	0.1900
200	0.0788	0.1490	0.2110	0.2680	0.2120
500	0.135	0.247	0.341	0.420	0.341
True count density	0.398	0.159	0.0641	0.0255	0.0639

TABLE 4.2

Relative error for $\boldsymbol{\beta}_3 = (-0.5, 0.1, 0.5)$ with no subgraph noising					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.634	0.823	0.887	0.915	0.888
100	0.529	0.675	0.722	0.743	0.723
200	0.385	0.560	0.649	0.703	0.649
500	0.239	0.419	0.554	0.657	0.554
Relative error for $\boldsymbol{\beta}_3 = (-0.5, 0.1, 0.5)$ with $p = 8/10$					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.549	0.721	0.780	0.806	0.781
100	0.596	0.759	0.808	0.827	0.808
200	0.334	0.493	0.579	0.633	0.579
500	0.220	0.384	0.508	0.604	0.508
Relative error for $\boldsymbol{\beta}_3 = (-0.5, 0.1, 0.5)$ with $p = 1/2$					
n	1-star	2-star	triangle	4-cycle	3-star
50	0.579	0.741	0.790	0.809	0.791
100	0.480	0.611	0.653	0.670	0.653
200	0.429	0.605	0.686	0.730	0.686
500	0.243	0.416	0.542	0.635	0.542
True count density	0.713	0.508	0.363	0.259	0.363

TABLE 4.3

resulting in a better fit. Nonetheless, this analysis would require a closer investigation of the conditions of the large deviation approximation.

This study was constrained by the size of the graphs that the `ergm` package could simulate in a reasonable amount of time, however our method can scale to very large real world graphs (with millions of vertices). In this regime, the behavior of the methods may change, and extending to massive graphs is reserved for future study.

Thus our simulations give preliminary evidence that a new pseudolikelihood approach for fitting ERGMs under noised subgraph counts may be plausible in massive graphs. In low density parameter regimes, the approximation attained good fits and had greater robustness than for higher density parameter regimes. We suspect this is due to either the indistinguishability of the model or slow convergence in the approximation, but more investigation is needed. While it may be limited in applicability only to graphs that are known to belong to a particular class ahead of time, the computational advantage should not be understated – the normalizing constant can be approximated orders of magnitude faster especially for larger graphs. To the best of our knowledge, combining approximate graphlet counting with the graph-limit pseudolikelihood is the only ERGM inference approach that can scale to massive graphs.

In future work, we would like to compare the robustness of the pseudolikelihood method against the robustness of the MCMLE method, which is a commonly used. Preliminary attempts showed poor performance from MCMLE, which we were unable to diagnose as inherent to the method, the model, or the Monte Carlo settings. Furthermore, more compute time would help increase the scale of the tests above to reduce the effects of noise.

4.5. Acknowledgements

We relied on the **ergm** and **ergm.graphlets** R-packages extensively for the simulations [**61**, **133**].

APPENDIX A

Remarks Relating to Chapter 2

A.1. Classes of Decision Strategies

In this section, we describe the main classes of decision strategies considered in our work. Most of the information here is a distillation of the results found in [**119**].

A decision strategy is the composite function transforming the data from all nodes to the decision made by the root. Naturally, such a function is a composition of the functions, or decision rules, used by the intermediate nodes. Much of the work done in the area of decentralized detection depends on restricting attention to a particular class of functions. Here we will try to capture the main classes.

Our setup is as before. Consider a perfect m-ary tree of depth D where each node corresponds to a sensor. Each sensor uses a b-bit decision rule $f : \{0,1\}^m \to \{0,1\}^b$ to communicate information up the tree. The decision rules may be stochastic or deterministic (we avoid the measure-theoretic formulation of randomized functions to simplify the presentation; when applicable, deterministic rules can be considered as stochastic rules by appropriately setting probabilities to 1).

The class of all *deterministic decision strategies*, where all the sensors use a deterministic decision rule, will be denoted by $\mathcal{F}^3$. The class of all *stochastic decision strategies*, where all the sensor nodes use a stochastic decision rule, will be denoted by $\mathcal{F}^4$. These two classes form the two most general classes available. A further restriction requires all the sensor to use an identical decision rule. This gives rise to the class of all *identical deterministic decision strategies* $\mathcal{F}^1$, and the class of all *identical stochastic decision strategies* $\mathcal{F}^2$ (for identical stochastic functions, the randomization of each sensor is separate). A yet further restriction requires the decision rules to be monotone threshold rules

$$\mathcal{M}^i \coloneqq \{F \in \mathcal{F}^i | F \text{ consists of monotone threshold rules}\}.$$

A decision rule is a *monotone threshold rule* if it can be written as a sum

$$f(x) = \sum_{i=1}^{K} [L(x) \leq t_i]$$

where we used Iverson bracket notation (i.e., $[E] = 1$ if E is true and 0 otherwise) and

$$L(x) \coloneqq \frac{P(x|H_0)}{P(x|H_1)}$$

is the likelihood ratio and $0 \leq t_1 \leq t_2 \leq \cdots \leq t_K$ is a sequence of thresholds. A simple way to describe these functions is that they are non-decreasing in the likelihood of x.

The inclusion relationships between the sets defined above are as follows (omitting the obvious inclusion of $\mathcal{M}^i$ in $\mathcal{F}^i$):

$$\begin{array}{ccc} \mathcal{F}^1 & & \mathcal{F}^2 \\ \cap & & \cap \\ \mathcal{F}^3 & \subset & \mathcal{F}^4 \end{array}$$

and

$$\begin{array}{ccc} \mathcal{M}^1 & & \mathcal{M}^2 \\ \cap & & \cap \\ \mathcal{M}^3 & \subset & \mathcal{M}^4 \end{array} .$$

Note that $\mathcal{F}^1 \subset \mathcal{F}^2$ only holds if the identical stochastic strategies are all correlated (all nodes choose the same strategy at random); otherwise the stochastic strategies can choose strategies that do not belong to $\mathcal{F}^1$.

We are interested in characterizing the relative optimal probability of error over the classes we have defined so far, so let us define the class error probability to be

$$P_e(S) \coloneqq \min_{F \in S} P(F(\mathbf{x}) \neq H)$$

where $\mathbf{x}$ is all the data at the leaf nodes, we have some distribution $\Pr(\boldsymbol{x}|H)$, and H is the hypothesis random variable (uniform over the hypotheses). A number of results are available in [**119**]. We summarize some relevant ones here. One fundamental fact is that randomized strategies do not

have an error advantage over deterministic strategies. That is to say, we have

$$P_e(\mathcal{F}^4) = P_e(\mathcal{F}^3)$$

$$P_e(\mathcal{M}^4) = P_e(\mathcal{M}^3)$$

The reason for this is that for a stochastic decision strategy F, that uses the deterministic decision strategy G_i with probability p_i, we have

$$P_e(F) = \sum_{i=1}^{K} p_i P_e(G_i)$$

which follows by conditioning. Minimizing the sum on the right over the p_i amounts to either finding the $P_e(G_i)$ with the smallest value and putting all the probability there or choosing one to break a tie (this follows from the nature of the simplex method). In either case, we have that for every stochastic strategy there exists a deterministic strategy that does at least as well.

Surprisingly, the strict inequality between identical deterministic decision strategies and non-identical deterministic decision strategies

$$P_e(\mathcal{F}^1) < P_e(\mathcal{F}^3)$$

is possible to achieve. Even more so, strict inequality between identical and non-identical strategies can be achieved even for monotone threshold strategies

$$P_e(\mathcal{M}^1) < P_e(\mathcal{M}^3).$$

(Analogous statements naturally hold for their randomized counterparts.) This stems from the existence of optimal strategies where the sensors use non-identical monotone threshold functions (see the appendix of [**118**] for a counter-example; we verified through numerical search that there are non-degenerate regions of hypotheses where non-identical strategies improve upon identical strategies). The fact that non-identical tree strategies are capable of a strictly better probability of error than identical strategies presents an interesting conceptual obstacle to deriving optimality. Nonetheless, we content ourselves, as done many decades ago, with a few facts. Firstly, we have numerical evidence suggesting that the optimality gap between the two is not large (see [**119**], section

2.3.2). Secondly, it can be shown that asymptotically, in the fusion center case, identical decision strategies can achieve the same exponent in their sample rate function exponent as non-identical decision strategies (see [**119**], section 5.1). These two facts can give us some comfort that the study of optimal identical decision strategies can be used to get approximate into the performance of general decision strategies.

The importance of monotone threshold rules stems from the fact that in the case of a two hypothesis test, the set of monotone threshold rules achieves optimality. This fact was proved (even in the case of tree strategies!) in [**119**] (see Proposition 2.4 and Proposition 4.1).

$$P_e(\mathcal{M}^3) = P_e(\mathcal{F}^3). \tag{A.1}$$

However, the same fact is not true when we restrict to identical strategies

$$P_e(\mathcal{M}^1) < P_e(\mathcal{F}^1). \tag{A.2}$$

A simple example of this inequality can be found in the binary tree case: no single monotone threshold rule can handle the case of a tie, introducing a bias that degrades performance in the whole tree, while non-monotone threshold rules can use the "choose left" decision rule (i.e., each parent uses the function $f(x, y) = x$) to break the tie and bring single sample up to the root node.

Let us come back and summarize a key takeaway of this section. While the question of optimality allows for some reductions in the function classes considered, these are sensitive to small changes in the problem and the function class. Thus, care must be taken in performing these reductions. While using all identical decision strategies is provably suboptimal compared to non-identical decision strategies, their asymptotic optimality and the bounded optimality gap makes restriction to the study of identical strategies reasonable. Additionally, while identical monotone threshold rules suffer from the same suboptimality in the general case, we see them as an analytically tractable subclass that can serve as a guide to the form of optimal results in the case of general decision strategies.

A.2. Chernoff Information in the Symmetric Binary Case

The Chernoff information between two Bernoulli hypotheses $p_0(x) \sim \text{Bern}(\frac{1}{2} - \delta)$ and $p_1(x) \sim \text{Bern}(\frac{1}{2} + \delta)$ can be found by solving

$$-\min_{0 \leq \lambda \leq 1} \log \left(\left(\frac{1}{2} - \delta\right)^{\lambda} \left(\frac{1}{2} + \delta\right)^{1-\lambda} \right) + \left(\left(\frac{1}{2} + \delta\right)^{\lambda} \left(\frac{1}{2} - \delta\right)^{1-\lambda} \right).$$

Through some elementary calculus, it can be seen that the optimum takes place at $\lambda = \frac{1}{2}$.

A.3. Constructing A Majority Voting Channel With Stochastic Channels

It is possible to construct stochastic rules with a fixed point at $p^* = 1/2$, whose parameter maps are different from the majority threshold rule. For example, the parameter map $1/2 f(p; m, \lfloor (m+1)/2 \rfloor - 1) + 1/2 f(p; m, \lfloor (m+1)/2 \rfloor + 1)$. However, the derivative can be shown to be just $5/4$ compared to the majority channel's $3/2$, suggesting that the majority channel is optimal in the sense of having maximal derivative at the fixed point. In turn, this suggests that the non-stochastic monotone threshold rules, when available for a given fixed point, are optimal (naturally, when the fixed point is such that it cannot be addressed by a non-stochastic monotone threshold rule, a stochastic mapping will need to be used and will then be optimal).

It is also possible to reconstruct the majority channel parameter map $f(p; m, \lfloor (m+1)/2 \rfloor)$ using a mixture of other stochastic rules. One way to do so is $p f(p; m, \lfloor (m+1)/2 \rfloor - 1) + (1-p) f(p; m, \lfloor (m+1)/2 \rfloor + 1)$. Note, however, that the mixture weight p is dynamic, which requires the ability to change the weights from layer to layer. Thus this option is only available if we allow the stochastic mapping to change from layer to layer. On a practical level this is implausible however, as the nodes would need to be aware of their position in the tree. This awareness would need to be communicated in some way and since we are already communication constrained, it is not a reasonable assumption to make.

A.4. Derivatives of Binomial Tail Sums

For odd $m = 2n + 1$, we have that

$$\frac{\partial f(p; m, \lfloor \frac{m+1}{2} \rfloor)}{\partial p}\Big|_{p=1/2} = \frac{4n \, {}_2\tilde{F}_1(2, 1-n; n+3; -1)\Gamma\left(n + \frac{3}{2}\right)}{\sqrt{\pi}} + 1$$

and for even $m = 2n$, we have

$$\frac{\partial f(p; m, \lfloor \frac{m+1}{2} \rfloor)}{\partial p}\Big|_{p=1/2} = \frac{2\Gamma\left(n + \frac{1}{2}\right)}{\sqrt{\pi}\Gamma(n)}$$

where ${}_2\tilde{F}_1(a, b, c; z)$ is the regularized hypergeometric function and $\Gamma(n)$ is the standard gamma function. It can be seen numerically that the expressions on the right are both asymptotically equivalent to $g(m) = 2^{-m} m \binom{m}{\frac{m}{2}}$ (in the sense that the ratio of the two functions tends to 1 as $m \to \infty$). This allows us to, as a sanity check, recover the parametric sample-variance tradeoff (i.e., $\delta \sim n^{-1/2}$) in the limit of an infinite width tree by applying a second order Stirling's approximation to $\log_m g(m)$.

A.5. Probability of error and MAP

Here we develop an expression for the probability of error for a root node that uses the maximum a posteriori (MAP) rule (which by Neymann-Pearson is the optimal choice).

Suppose we perform the generalized hypothesis test Equation (2.2) in a D depth m-ary tree with the children using a decision rule with parameter map $f(p)$. Define the MAP rule by

$$\mathrm{MAP}(\boldsymbol{x}) = \mathbf{1}_{\boldsymbol{x} \in \mathcal{R}}$$

where the rejection region $\mathcal{R}$ is defined as

$$\begin{aligned}
\mathcal{R} &= \{\boldsymbol{x} | \Pr(\boldsymbol{x}|H_0) < \Pr(\boldsymbol{x}|H_1)\} \\
&= \left\{\boldsymbol{x} \mid j = \|\boldsymbol{x}\|_0,\ \bar{p}_0^j \bar{q}_0^{m-j} < \bar{p}_1^j \bar{q}_1^{m-j}\right\}, \\
&= \left\{\boldsymbol{x} \mid j = \|\boldsymbol{x}\|_0,\ j > m \frac{\log\left(\frac{\bar{q}_1}{\bar{p}_0}\right)}{\log\left(\frac{\bar{p}_0 \bar{q}_1}{\bar{p}_1 \bar{q}_0}\right)}\right\},
\end{aligned}$$

($\boldsymbol{x} \in \{0,1\}^m$, $\|\boldsymbol{x}\|_0 = \sum_{i=1}^m x_i$, and $\bar{p}_i = f^{D-1}(p_i)$ and $\bar{q}_i = 1 - f^{D-1}(p_i)$). We also note that we have assumed that $\Pr(H_0) = \Pr(H_1) = \frac{1}{2}$). We can write the probability of error for this rule as

$$2p_e = \Pr(\mathcal{R}|H_0) + \Pr(\mathcal{R}^c|H_1).$$

Analyzing this quantity is challenging. This is due to the complexity of the rejection region $\mathcal{R}$, which depends on D and the convergence rates of f. Note that this stems from the fact that the MAP threshold for a regular Binomial is quite complicated (as can be seen in the definition of the rejection region above). However, intuitively we should have that as $D \to \infty$, $\mathcal{R} \to \{\boldsymbol{x} \mid j > \frac{m}{2}\}$, because the root will expect the induced distribution below it to be well separated. Using Mathematica, we were able to show that the threshold in $\mathcal{R}$ limits to $j > m/2$ if $p_0 = \delta$ and $p_1 = 1 - c_1\delta$, for some $c_1 > 0$ and as $\delta \to 0$. Hence, asymptotically, we can treat MAP as being identical to the majority vote.

A.6. Majority Rules in Ternary Trees: Approximations to the Kullback-Leibler Contraction Ratio

Consider the Kullback-Leibler divergence for the majority channel in the ternary case with $(h_0, h_1) = (\frac{1}{2}, \frac{1}{2} + \delta)$. It can be shown that the KL contraction ratio for the majority channel

$$\frac{KL(T(1/2), T(1/2+\delta)}{KL(1/2, 1/2+\delta)} = 3/4 - 11/8\delta^2 - O(\delta^4)$$

with only even power terms remaining and hence is maximized at $3/4$. Combining this with the well-known lower bound (see chapter 2 of [**120**])

$$\begin{aligned}
\Pr(\text{err}) &\geq \frac{1}{4}\exp\{-KL(A^D h_0, A^D h_1)\} \\
&\geq \frac{1}{4}\exp\left\{-\left(\frac{3}{4}\right)^D 3^D KL\left(\frac{1}{2}, \frac{1}{2}+\delta\right)\right\} \\
&\geq \frac{1}{4}\exp\left\{-\left(\frac{9}{4}\right)^D \delta^2\right\}
\end{aligned}$$

which leads to the sample-bias tradeoff[1]

$$\delta \sim n^{-\frac{1}{2}\log_3 \frac{9}{4}} = n^{-\log_3 \frac{3}{2}}.$$

[1]We have used

$$\begin{aligned}
KL\left(\frac{1}{2}, \frac{1}{2}+\delta\right) &= \log(1-4\delta^2) = O(\delta^2) \\
KL\left(\frac{1}{2}+\delta, \frac{1}{2}\right) &= \frac{1}{2}\log(1-4\delta^2) + \delta\log\left(\frac{1+2\delta}{1-2\delta}\right) = O(\delta^2).
\end{aligned}$$

Bibliography

[1] C. C. Aggarwal, *An Introduction to Social Network Data Analytics*, in Social Network Data Analytics, C. C. Aggarwal, ed., Springer US, Boston, MA, 2011, pp. 1–15.

[2] N. K. Ahmed, J. Neville, R. A. Rossi, N. G. Duffield, and T. L. Willke, *Graphlet decomposition: Framework, algorithms, and applications*, Knowledge and Information Systems, 50 (2017), pp. 689–722.

[3] E. Albina, *Epidemiology of porcine reproductive and respiratory syndrome (PRRS): An overview*, Veterinary Microbiology, 55 (1997), pp. 309–316.

[4] S.-i. Amari, *Information Geometry and Its Applications*, Springer, Feb. 2016.

[5] E. Arias-Castro, E. J. Candès, and A. Durand, *Detection of an anomalous cluster in a network*, The Annals of Statistics, 39 (2011), pp. 278–304.

[6] E. Arias-Castro, D. L. Donoho, and X. Huo, *Near-optimal detection of geometric objects by fast multiscale methods*, IEEE Transactions on Information Theory, 51 (2005), pp. 2402–2425.

[7] S. A. Bagloee, M. Tavana, M. Asadi, and T. Oliver, *Autonomous vehicles: Challenges, opportunities, and future implications for transportation policies*, Journal of Modern Transportation, 24 (2016), pp. 284–303.

[8] D. Bajovic, D. Jakovetic, J. Xavier, B. Sinopoli, and J. M. F. Moura, *Distributed Detection via Gaussian Running Consensus: Large Deviations Asymptotic Analysis*, IEEE Transactions on Signal Processing, 59 (2011), pp. 4381–4396.

[9] M. Baker, *1,500 Scientists Lift the Lid on Reproducibility*, Nature, 533 (2016), pp. 452–454.

[10] Y. Benjamini and Y. Hochberg, *Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing*, Journal of the Royal Statistical Society: Series B (Methodological), 57 (1995), pp. 289–300.

[11] Y. Benjamini and D. Yekutieli, *The control of the false discovery rate in multiple testing under dependency*, Annals of Statistics, 29 (2001), pp. 1165–1188.

[12] J. Bennett, S. Lanning, and N. Netflix, *The Netflix Prize*, in In KDD Cup and Workshop in Conjunction with KDD, 2007.

[13] J. Besag, *Spatial Interaction and the Statistical Analysis of Lattice Systems*, Journal of the Royal Statistical Society: Series B (Methodological), 36 (1974), pp. 192–225.

[14] M. A. Bhuiyan, M. Rahman, M. Rahman, and M. A. Hasan, *GUISE: Uniform Sampling of Graphlets for Large Graph Analysis*, in 2012 IEEE 12th International Conference on Data Mining, Dec. 2012, pp. 91–100.

[15] P. J. Bickel, A. Chen, and E. Levina, *The method of moments and degree distributions for network models*, The Annals of Statistics, 39 (2011), pp. 2280–2301.

[16] N. Biggs, E. K. Lloyd, and R. J. Wilson, *Graph Theory, 1736-1936*, Clarendon Press, 1986.

[17] G. Blanchard, S. Delattre, E. Roquain, et al., *Testing over a continuum of null hypotheses with false discovery rate control*, Bernoulli, 20 (2014), pp. 304–333.

[18] C. Borgs, J. T. Chayes, L. Lovász, V. T. Sós, and K. Vesztergombi, *Convergent sequences of dense graphs I: Subgraph frequencies, metric properties and testing*, Advances in Mathematics, 219 (2008), pp. 1801–1851.

[19] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Mar. 2004.

[20] P. Braca, S. Marano, V. Matta, and P. Willett, *Asymptotic Optimality of Running Consensus in Testing Binary Hypotheses*, IEEE Transactions on Signal Processing, 58 (2010), pp. 814–825.

[21] M. Braverman, A. Garg, T. Ma, H. L. Nguyen, and D. P. Woodruff, *Communication lower bounds for statistical estimation problems via a distributed data processing inequality*, in Proceedings of the forty-eighth annual ACM symposium on Theory of Computing, ACM, 2016, pp. 1011–1020.

[22] S. M. Brennan, A. B. Maccabe, A. M. Mielke, and D. C. Torney, *Radiation Detection with Distributed Sensor Networks*, IEEE COMPUTER, 37 (2004), p. 10.

[23] M. Bressan, F. Chierichetti, R. Kumar, S. Leucci, and A. Panconesi, *Counting Graphlets: Space vs Time*, in Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, Cambridge United Kingdom, Feb. 2017, ACM, pp. 557–566.

[24] ——, *Motif Counting Beyond Five Nodes*, ACM Transactions on Knowledge Discovery from Data, 12 (2018), pp. 48:1–48:25.

[25] R. M. Chang, R. J. Kauffman, and Y. Kwon, *Understanding the paradigm shift to computational social science in the presence of big data*, Decision Support Systems, 63 (2014), pp. 67–80.

[26] S. Chatterjee and P. Diaconis, *Estimating and understanding exponential random graph models*, The Annals of Statistics, 41 (2013), pp. 2428–2461.

[27] S. Chatterjee and S. R. S. Varadhan, *The large deviation principle for the Erdős-Rényi random graph*, European Journal of Combinatorics, 32 (2011), pp. 1000–1017.

[28] ——, *The large deviation principle for the Erdős-Rényi random graph*, European Journal of Combinatorics, 32 (2011), pp. 1000–1017.

[29] H. Chernoff et al., *A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations*, The Annals of Mathematical Statistics, 23 (1952), pp. 493–507.

[30] S. K. Chong, M. M. Gaber, S. Krishnaswamy, and S. W. Loke, *Energy-Aware Data Processing Techniques for Wireless Sensor Networks: A Review*, in Transactions on Large-Scale Data- and Knowledge-Centered Systems III: Special Issue on Data and Knowledge Management in Grid and P2P Systems, A. Hameurlain, J. Küng, and R. Wagner, eds., Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2011, pp. 117–137.

[31] A. Ciancio, S. Pattem, A. Ortega, and B. Krishnamachari, *Energy-efficient data representation and routing for wireless sensor networks based on a distributed wavelet compression algorithm*, in Proceedings of the 5th international conference on Information processing in sensor networks, ACM, 2006, pp. 309–316.

[32] J. Cohen, J. H. B. Kempermann, and G. Zbaganu, *Comparisons of Stochastic Matrices with Applications in Information Theory, Statistics, Economics and Population*, Springer Science & Business Media, Sept. 1998.

[33] O. S. Collaboration, *Estimating the reproducibility of psychological science*, Science, 349 (2015).

[34] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, Nov. 2012.

[35] D. Culler, D. Estrin, and M. Srivastava, *Guest Editors' Introduction: Overview of Sensor Networks*, Computer, 37 (2004), pp. 41–49.

[36] J. C. Duchi, M. I. Jordan, M. J. Wainwright, and Y. Zhang, *Optimality guarantees for distributed statistical estimation*, arXiv preprint arXiv:1405.0782, (2014).

[37] J. C. Duchi and M. J. Wainwright, *Distance-based and continuum Fano inequalities with applications to statistical estimation*, arXiv:1311.2669 [cs, math, stat], (2013). arXiv: 1311.2669.

[38] L. Duczmal, M. Kulldorff, and L. Huang, *Evaluation of Spatial Scan Statistics for Irregularly Shaped Clusters*, Journal of Computational and Graphical Statistics, 15 (2006), pp. 428–442.

[39] J. Fang, H. Li, Z. Chen, and S. Li, *Optimal Precoding Design and Power Allocation for Decentralized Detection of Deterministic Signals*, IEEE Transactions on Signal Processing, 60 (2012), pp. 3149–3163.

[40] T. Fawcett, *An introduction to ROC analysis*, Pattern Recognition Letters, 27 (2006), pp. 861–874.

[41] P. J. Feibelman, *A PhD Is Not Enough!: A Guide to Survival in Science*, Basic Books, Jan. 2011.

[42] S. E. Fienberg, *Introduction to papers on the modeling and analysis of network data*, The Annals of Applied Statistics, 4 (2010), pp. 1–4.

[43] R. A. Fisher, *Statistical Methods and Scientific Inference*, Statistical Methods and Scientific Inference, Hafner Publishing Co., Oxford, England, 1956.

[44] O. Frank and D. Strauss, *Markov Graphs*, Journal of the American Statistical Association, 81 (1986), pp. 832–842.

[45] C. J. Geyer, *Stat 8053 Lecture Notes: Exponential Families*, (2014).

[46] C. J. Geyer and E. A. Thompson, *Constrained Monte Carlo Maximum Likelihood for Dependent Data*, Journal of the Royal Statistical Society. Series B (Methodological), 54 (1992), pp. 657–699.

[47] W. R. Gilks, *Markov Chain Monte Carlo*, in Encyclopedia of Biostatistics, American Cancer Society, 2005.

[48] J. Glaz, J. Naus, and S. Wallenstein, *Scan Statistics*, Springer Series in Statistics, Springer-Verlag, New York, 2001.

[49] M. G. G'Sell, S. Wager, A. Chouldechova, and R. Tibshirani, *Sequential selection procedures and false discovery rate control*, Journal of the Royal Statistical Society. Series B (Statistical Methodology), 78 (2016), pp. 423–444.

[50] A. Hagberg, P. Swart, and D. S Chult, *Exploring network structure, dynamics, and function using networkx*, Tech. Rep. LA-UR-08-05495; LA-UR-08-5495, Los Alamos National Lab. (LANL), Los Alamos, NM (United States), Jan. 2008.

[51] G. Han and H. Sethu, *Waddling Random Walk: Fast and Accurate Mining of Motif Statistics in Large Graphs*, in 2016 IEEE 16th International Conference on Data Mining (ICDM), Dec. 2016, pp. 181–190.

[52] M. S. Handcock, G. Robins, T. Snijders, J. Moody, and J. Besag, *Assessing Degeneracy in Statistical Models of Social Networks*, Journal of the American Statistical Association, 76 (2003), pp. 33–50.

[53] W. K. Hastings, *Monte Carlo sampling methods using Markov chains and their applications*, Biometrika, 57 (1970), pp. 97–109.

[54] R. He and T. Zheng, *GLMLE: Graph-limit enabled fast computation for fitting exponential random graph models to large social networks*, Social Network Analysis and Mining, 5 (2015), p. 8.

[55] R. Heffernan, F. Mostashari, D. Das, M. Besculides, C. Rodriguez, J. Greenko, L. Steiner-Sichel, S. Balter, A. Karpati, P. Thomas, M. Phillips, J. Ackelsberg, E. Lee, J. Leng, J. Hartman, K. Metzger, R. Rosselli, and D. Weiss, *New York City Syndromic Surveillance Systems*, Morbidity and Mortality Weekly Report, 53 (2004), pp. 25–27.

[56] Y. Hochberg, *A sharper Bonferroni procedure for multiple tests of significance*, Biometrika, 75 (1988), pp. 800–802.

[57] P. W. Holland and S. Leinhardt, *An Exponential Family of Probability Distributions for Directed Graphs*, Journal of the American Statistical Association, 76 (1981), pp. 33–50.

[58] D. J. Holtkamp, *Assessment of the economic impact of porcine reproductive and respiratory syndrome virus on United States pork producers*, Journal of Swine Health and Production, 21 (2013), pp. 72–84.

[59] D. R. Hunter, S. M. Goodreau, and M. S. Handcock, *Goodness of Fit of Social Network Models*, Journal of the American Statistical Association, 103 (2008), pp. 248–258.

[60] D. R. Hunter and M. S. Handcock, *Inference in Curved Exponential Family Models for Networks*, Journal of Computational and Graphical Statistics, 15 (2006), pp. 565–583.

[61] D. R. Hunter, M. S. Handcock, C. T. Butts, S. M. Goodreau, and M. Morris, *Ergm: A Package to Fit, Simulate and Diagnose Exponential-Family Models for Networks*, Journal of statistical software, 24 (2008), p. nihpa54860.

[62] E. T. Jaynes, *Information Theory and Statistical Mechanics*, Physical Review, 106 (1957), pp. 620–630.

[63] ——, *On the rationale of maximum-entropy methods*, Proceedings of the IEEE, 70 (1982), pp. 939–952.

[64] Z. Kabluchko, *Extremes of the standardized gaussian noise*, Stochastic Processes and their Applications, 121 (2011), pp. 515–533.

[65] J. N. Kapur and H. K. Kesavan, *Entropy Optimization Principles and Their Applications*, in Entropy and Energy Dissipation in Water Resources, V. P. Singh and M. Fiorentino, eds., Water Science and Technology Library, Springer Netherlands, Dordrecht, 1992, pp. 3–20.

[66] C. König, A. Munk, and F. Werner, *Multidimensional multiscale scanning in exponential families: Limit theory and statistical consequences*, The Annals of Statistics, 48 (2020), pp. 655–678.

[67] J. Krauth, *Discrete scan statistics for detecting change-points in binomial sequences*, in Classification in the Information Age, Springer, 1999, pp. 196–204.

[68] O. P. Kreidl and A. S. Willsky, *An Efficient Message-Passing Algorithm for Optimizing Decentralized Detection Networks*, IEEE Transactions on Automatic Control, 55 (2010), pp. 563–578.

[69] M. Kulldorff, R. Heffernan, J. Hartman, R. Assunção, and F. Mostashari, *A Space–Time Permutation Scan Statistic for Disease Outbreak Detection*, PLOS Medicine, 2 (2005), p. e59.

[70] M. Kulldorff, K. Rand, G. Gherman, G. Williams, and D. DeFrancesco, *Satscan v 2.1: Software for the spatial and space-time scan statistics*, Bethesda, MD: National Cancer Institute, (1998).

[71] E. L. Lehmann and J. P. Romano, *Testing statistical hypotheses*, Springer Science & Business Media, 2006.

[72] E. H. Lieb and M. Loss, *Analysis*, American Mathematical Society, Providence, RI, 2nd edition ed., Mar. 2001.

[73] C. R. Loader, *Large-deviation approximations to the distribution of scan statistics*, Advances in Applied Probability, 23 (1991), pp. 751–771.

[74] L. Lovász, *Large Networks and Graph Limits*, American Mathematical Soc., 2012.

[75] L. Lovász and B. Szegedy, *Limits of dense graph sequences*, Journal of Combinatorial Theory, Series B, 96 (2006), pp. 933–957.

[76] Y. Low, J. E. Gonzalez, A. Kyrola, D. Bickson, C. E. Guestrin, and J. Hellerstein, *GraphLab: A New Framework For Parallel Machine Learning*, arXiv:1408.2041 [cs], (2014). arXiv: 1408.2041.

[77] J. K. Lunney, D. A. Benfield, and R. R. Rowland, *Porcine reproductive and respiratory syndrome virus: An update on an emerging and re-emerging viral disease of swine*, Virus Research, 154 (2010), pp. 1–6.

[78] P. Martelli, S. Gozio, L. Ferrari, S. Rosina, E. De Angelis, C. Quintavalla, E. Bottarelli, and P. Borghetti, *Efficacy of a modified live porcine reproductive and respiratory syndrome virus (PRRSV) vaccine in pigs naturally exposed to a heterologous European (Italian cluster) field strain: Clinical protection and cell-mediated immunity*, Vaccine, 27 (2009), pp. 3788–3799.

[79] Y. Mei, *Asymptotic Optimality Theory for Decentralized Sequential Hypothesis Testing in Sensor Networks*, IEEE Transactions on Information Theory, 54 (2008), pp. 2072–2089.

[80] J. Michell, *XXVII. An inquiry into the probable parallax, and magnitude of the fixed stars, from the quantity of light which they afford us, and the particular circumstances of their situation, by the Rev. John Michell, B. D. F. R. S*, Philosophical Transactions of the Royal Society of London, 57 (1767), pp. 234–264.

[81] J. C. Miller and T. Ting, *EoN (Epidemics on Networks): A fast, flexible Python package for simulation, analytic approximation, and analysis of epidemics on networks*, Journal of Open Source Software, 4 (2019), p. 1731.

[82] M. Newman, *Networks*, Oxford University Press, July 2018.

[83] F. Nielsen, *An information-geometric characterization of chernoff information*, IEEE Signal Processing Letters, 20 (2013), pp. 269–272.

[84] R. Niu and P. K. Varshney, *Distributed Detection and Fusion in a Large Wireless Sensor Network of Random Size*, EURASIP Journal on Wireless Communications and Networking, 2005 (2005), p. 815873.

[85] I. S. U. of Science and Technology., *Porcine Reproductive and Respiratory Syndrome (PRRS)*, 2020.

[86] K. Paramonov, D. Shemetov, and J. Sharpnack, *Estimating Graphlet Statistics via Lifting*, in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19, New York, NY, USA, July 2019, Association for Computing Machinery, pp. 587–595.

[87] G. P. Patil and C. Taillie, *Upper level set scan statistic for detecting arbitrarily shaped hotspots*, Environmental and Ecological Statistics, 11 (2004), pp. 183–197.

[88] F. Picard, P. Reynaud-Bouret, and E. Roquain, *Continuous testing for Poisson process intensities: A new perspective on scanning statistics*, Biometrika, 105 (2018), pp. 931–944.

[89] E. Pileri and E. Mateu, *Review on the transmission porcine reproductive and respiratory syndrome virus between pigs and farms and impact on vaccination*, Veterinary research, 47 (2016), p. 108.

[90] A. Pinar, C. Seshadhri, and V. Vishal, *ESCAPE: Efficiently Counting All 5-Vertex Subgraphs*, in Proceedings of the 26th International Conference on World Wide Web, WWW '17, Republic and Canton of Geneva, CHE, Apr. 2017, International World Wide Web Conferences Steering Committee, pp. 1431–1440.

[91] C. E. Priebe, J. M. Conroy, D. J. Marchette, and Y. Park, *Scan Statistics on Enron Graphs*, Computational & Mathematical Organization Theory, 11 (2005), pp. 229–247.

[92] N. Pržulj, *Biological network comparison using graphlet degree distribution*, Bioinformatics, 23 (2007), pp. e177–e183.

[93] N. Pržulj, D. G. Corneil, and I. Jurisica, *Modeling interactome: Scale-free or geometric?*, Bioinformatics, 20 (2004), pp. 3508–3515.

[94] ——, *Efficient estimation of graphlet frequency distributions in protein–protein interaction networks*, Bioinformatics, 22 (2006), pp. 974–980.

[95] M. Raginsky, *Strong Data Processing Inequalities and ϕ -Sobolev Inequalities for Discrete Channels*, IEEE Transactions on Information Theory, 62 (2016), pp. 3355–3389.

[96] M. Rahman, M. A. Bhuiyan, and M. A. Hasan, *Graft: An Efficient Graphlet Counting Method for Large Graph Analysis*, IEEE Transactions on Knowledge and Data Engineering, 26 (2014), pp. 2466–2478.

[97] T. S. Rappaport, *The wireless revolution*, IEEE Communications Magazine, 29 (1991), pp. 52–71.

[98] G. Robins, P. Pattison, Y. Kalish, and D. Lusher, *An introduction to exponential random graph (p*) models for social networks*, Social Networks, 29 (2007), pp. 173–191.

[99] G. Robins, T. Snijders, P. Wang, M. Handcock, and P. Pattison, *Recent developments in exponential random graph (p*) models for social networks*, Social Networks, 29 (2007), pp. 192–215.

[100] I. Robinson, J. Webber, and E. Eifrem, *Graph Databases*, O'Reilly Media, Inc., 2013.

[101] O. Shamir, *Fundamental limits of online and distributed algorithms for statistical learning and estimation*, in Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1, NIPS'14, Cambridge, MA, USA, Dec. 2014, MIT Press, pp. 163–171.

[102] C. E. Shannon, R. G. Gallager, and E. R. Berlekamp, *Lower bounds to error probability for coding on discrete memoryless channels. i*, Information and Control, 10 (1967), pp. 65–103.

[103] J. Sharpnack and E. Arias-Castro, *Exact asymptotics for the scan statistic and fast alternatives*, Electronic Journal of Statistics, 10 (2016), pp. 2641–2684.

[104] J. Sharpnack, A. Krishnamurthy, and A. Singh, *Near-optimal anomaly detection in graphs using Lovász extended scan statistic*, in Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13, Red Hook, NY, USA, Dec. 2013, Curran Associates Inc., pp. 1959–1967.

[105] J. Sharpnack, D. Shemetov, K. Proksch, and B. Martínez-López, *Scan statistics and multiple testing in networks with applications to veterinary epidemiology*, Frontiers in Veterinary Science, 6 (2019).

[106] J. Sharpnack, A. Singh, and A. Krishnamurthy, *Detecting Activations over Graphs using Spanning Tree Wavelet Bases*, in Artificial Intelligence and Statistics, PMLR, Apr. 2013, pp. 536–544.

[107] N. Shervashidze, S. V. N. Vishwanathan, T. Petri, K. Mehlhorn, and K. Borgwardt, *Efficient graphlet kernels for large graph comparison*, in Artificial Intelligence and Statistics, PMLR, Apr. 2009, pp. 488–495.

[108] Z. Šidák, *Rectangular Confidence Regions for the Means of Multivariate Normal Distributions*, Journal of the American Statistical Association, 62 (1967), pp. 626–633.

[109] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, *Mastering the game of Go with deep neural networks and tree search*, Nature, 529 (2016), pp. 484–489.

[110] T. A. B. Snijders, *Markov chain monte carlo estimation of exponential random graph models*, Journal of Social Structure, 3 (2002).

[111] T. A. B. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock, *New Specifications for Exponential Random Graph Models*, Sociological Methodology, 36 (2006), pp. 99–153.

[112] S. Speakman and D. B. Neill, *Fast graph scan for scalable detection of arbitrary connected clusters*, in Proceedings of the 2009 international society for disease surveillance annual conference, 2010.

[113] J. D. Storey, *A direct approach to false discovery rates*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 64 (2002), pp. 479–498.

[114] D. Strauss and M. Ikeda, *Pseudolikelihood Estimation for Social Networks*, Journal of the American Statistical Association, 85 (1990), pp. 204–212.

[115] N. H. Tran, K. P. Choi, and L. Zhang, *Counting motifs in the human interactome*, Nature Communications, 4 (2013), p. 2241.

[116] Y. Tsai and L. Lin, *Sequential Fusion for Distributed Detection Over BSC Channels in an Inhomogeneous Sensing Environment*, IEEE Signal Processing Letters, 17 (2010), pp. 99–102.

[117] J. Tsitsiklis and M. Athans, *On the complexity of decentralized decision making and detection problems*, IEEE Transactions on Automatic Control, 30 (1985), pp. 440–446.

[118] J. N. Tsitsiklis, *Decentralized detection by a large number of sensors*, Mathematics of Control, Signals, and Systems, 1 (1988), pp. 167–182.

[119] J. N. Tsitsiklis and Center for Intelligent Control Systems (U.S.), *Decentralized Detection*, Center for Intelligent Control Systems, M.I.T., Cambridge, Mass., 1989.

[120] A. B. Tsybakov, *Introduction to Nonparametric Estimation*, Springer Series in Statistics, Springer-Verlag, New York, 2009.

[121] D. Ustebay, R. Castro, and M. Rabbat, *Efficient Decentralized Approximation via Selective Gossip*, IEEE Journal of Selected Topics in Signal Processing, 5 (2011), pp. 805–816.

[122] M. A. J. van Duijn, K. J. Gile, and M. S. Handcock, *A framework for the comparison of maximum pseudo-likelihood and maximum likelihood estimation of exponential family random graph models*, Social Networks, 31 (2009), pp. 52–62.

[123] P. K. Varshney, *Distributed detection and data fusion*, Springer Science & Business Media, 2012.

[124] M. M. Wagner, F.-C. Tsui, J. U. Espino, V. M. Dato, D. F. Sitting, R. A. Caruana, L. F. McGinnis, D. W. Deerfield, M. J. Druzdzel, and D. B. Fridsma, *The Emerging Science of Very Early Detection of Disease Outbreaks:*, Journal of Public Health Management and Practice, 7 (2001), pp. 51–59.

[125] M. J. Wainwright and M. I. Jordan, *Graphical Models, Exponential Families, and Variational Inference*, Now Publishers Inc, 2008.

[126] P. Wang, J. C. S. Lui, B. Ribeiro, D. Towsley, J. Zhao, and X. Guan, *Efficiently Estimating Motif Statistics of Large Networks*, ACM Transactions on Knowledge Discovery from Data, 9 (2014), pp. 8:1–8:27.

[127] P. Wang, G. Robins, and P. Pattison, *Pnet: A program for the simulation and estimation of exponetial random graph models*, University of Melbourne, (2006).

[128] Y. Wang and Y. Mei, *Quantization Effect on the Log-Likelihood Ratio and Its Application to Decentralized Sequential Detection*, IEEE Transactions on Signal Processing, 61 (2013), pp. 1536–1543.

[129] S. Wasserman and P. Pattison, *Logit models and logistic regressions for social networks: I. An introduction to Markov graphs andp*, Psychometrika, 61 (1996), pp. 401–425.

[130] D. Watts and S. Strogatz, *Collective dynamics of 'small-world' networks*, Nature, (1998).

[131] E. P. Xing, Q. Ho, P. Xie, and D. Wei, *Strategies and Principles of Distributed Machine Learning on Big Data*, Engineering, 2 (2016), pp. 179–195.

[132] Y. Yang and A. Barron, *Information-theoretic determination of minimax rates of convergence*, The Annals of Statistics, 27 (1999), pp. 1564–1599.

[133] Ö. N. Yaveroglu, S. M. Fitzhugh, M. Kurant, A. Markopoulou, C. T. Butts, and N. Przulj, *Ergm.graphlets : A Package for ERG Modeling Based on Graphlet Statistics*, Journal of Statistical Software, 65 (2015).

[134] D. Yekutieli, *Hierarchical false discovery rate–controlling methodology*, Journal of the American Statistical Association, 103 (2008), pp. 309–316.

[135] O. Younis and S. Fahmy, *HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks*, IEEE Transactions on Mobile Computing, 3 (2004), pp. 366–379.

[136] C. Zhang, J. Fan, and T. Yu, *Multiple testing via fdrl for large scale imaging data*, Annals of statistics, 39 (2011), p. 613.

[137] N. R. Zhang, B. Yakir, L. C. Xia, D. Siegmund, et al., *Scan statistics on poisson random fields with applications in genomics*, The Annals of Applied Statistics, 10 (2016), pp. 726–755.

[138] S. Zhang, A. E. Choromanska, and Y. LeCun, *Deep learning with elastic averaging sgd*, in Advances in Neural Information Processing Systems, 2015, pp. 685–693.

[139] Y. Zhang, J. Duchi, M. I. Jordan, and M. J. Wainwright, *Information-theoretic lower bounds for distributed statistical estimation with communication constraints*, in Advances in Neural Information Processing Systems 26, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds., Curran Associates, Inc., 2013, pp. 2328–2336.