Towards Differentially Private Inference on Social Network Data

Alexander Goldberg

Abstract

This is my abstract.

Acknowledgements

Thanks everyone!

Contents

Chapter 1: Introduction

This is my intro...

Chapter 2: Statistical Modeling of Networks

An increasingly popular approach in quantitative analysis of networks is to fit statistical models to realized network data. Many of these models have generative interpretations, allowing researchers to understand the relative importance of multiple endogenous processes to the resulting structure of the network. The advantage of such an approach is best illustrated in contrast to computing statistics – like degree distributions or clustering coefficients – to describe the network structure, without an explicit model of the network. While such metrics are useful in summarizing the structural properties of a given network, they cannot tease out the underlying processes that may give rise to such structures.

For example, one of the distinguishing characteristics of many real-world social networks is the tendency to have more triangles (sets of three connected nodes) than would be expected by drawing random edges of a graph [GKM09]. There are a number of different processes in the formation of a friend network that could give rise to this outcome. One potential explanation is the notion of "triangle closure," or the tendency for people to become friends with friends-of-friends, since they are easier to meet. Another subtly different explanation is that triangles arise out of "assortative matching," the propensity for people with the same attributes to become friends with one another, leading to clustering in the network. Finally, a high number of triangles in a social network could arise for reasons of "sociality," the presence of only a few highly social individuals in the network, who are mutual friends to many people.

In order to consider what global or local processes best explain particular structures of a network, a statistical model of network data posits a probability distribution over the space of possible networks. The goal of inference is to tune parameters of the distribution, such that the realized network is likely to be observed under the probability distribution.

A simple example of such a model is the Erdös-Rényi Random Graph Model, known as the $G(n, p)$ model, which proposes that edges are drawn independently with probability p between any two nodes of a network with n nodes. While this model has been studied in great depth by graph theorists, it does not capture many important features of real world networks, like the tendency for clustering or the power-law distribution of degrees. In order to model such structures in networks, a more general class of random graph models are Exponential Random Graph Models. While these models arose out of the sociology literature, particularly in studying social networks, they have been applied to a broad range of problems, including analysis of interactions between proteins in the human body [RAS10], networks of neurons in the brain as people age [SDC+16], corporate management structures at Enron [UHH13], and the demographics of high school friendships [GKM09].

2.1 Exponential Random Graph Models

Formally, a graph $G = (V, E)$ is defined by a set of nodes (or vertices) V, with $|V| = n$ and edges E, representing the presence or absence of relationships between nodes. We will use the "adjacency matrix" representation of a graph, which we denote x, where $x_{ij} = 1$ if an edge exists between nodes i and j and $x_{ij} = 0$ otherwise. The models we consider are defined over undirected graphs, where all the edges are bidirectional, and the adjacency matrix is therefore symmetric. We refer to the number of edges adjacent to node i as the *degree* of node i so $d_i = \sum_{j=1}^{n} x_{ij}$. Then, the *degree distribution* is $D = (D_0, ..., D_{n-1})$ where $D_k = |\{i \in V : d_i = k\}|$.

Definition 2.1 (Exponential Random Graph [WP96])**.** A probability distribution over graphs of n vertices belongs to the family of *exponential random graph models* (henceforth referred to as ERGMs) if it takes the form:

$$\Pr(x|\theta) = \exp\left\{\theta^T u(x) - \psi(\theta)\right\}$$

where θ is a vector of parameters of the model, $u(x)$ is a vector of sufficient statistics computed on graph x, and $\psi(\theta)$ is a normalization constant needed to ensure a valid probability distribution so:

$$\psi(\theta) = \log \sum_{x'} \exp\left\{\theta^T u(x')\right\}$$

ERGMs describe a broad class of random graphs, with varying conditional dependence relationships between edges. For instance, the $G(n, p)$ graph can be viewed as an ERGM:

Example 2.1 ($G(n, p)$ graphs)**.** We can represent the Erdös-Rényi Random Graph ($G(n, p)$) model as an ERGM, by taking

$$u(x) = |E|, \quad \theta = \log \frac{p}{1-p}$$

$$\psi(\theta) = -\binom{n}{2} \log(1-p) = -\binom{n}{2} \log \frac{e^{-\theta}}{1+e^{-\theta}}$$

Then,

$$\begin{aligned}
\Pr(x|\theta) &= \exp\left\{|E| \log \frac{p}{1-p} + \binom{n}{2} \log(1-p)\right\} \\
&= p^{|E|}(1-p)^{\binom{n}{2}-|E|} \\
&= \prod_{i<j} p^{x_{ij}}(1-p)^{1-x_{ij}}
\end{aligned}$$

so each possible edge is included independently with probability p as specified by the Erdös-Rényi Model.

In order to model more complex structures in a network, researchers have proposed higher order sufficient statistics of ERGMs that imply more general conditional independence

assumptions than the Erdös-Rényi Model. For instance, "Markov" graphs, allow the probabilities of any two possible edges in a graph to be conditionally dependent if the edges share a common endpoint. This dependency allows for node level effects on edge formation. In fact, Markov dependencies are captured by ERGMs of the following form:

Example 2.2 (Markov graphs [FS86])**.** Any undirected *Markov graph* has probability distribution:

$$\Pr(x|\theta,\tau) = \exp\left\{\sum_{k=1}^{n-1}\theta_k S_k(x) + \tau T(x) - \psi(\theta,\tau)\right\}$$

where the sufficient statistics are

$$\begin{aligned}
&\text{number of edges:} && S_1(x) = \sum_{1\le i<j\le n} x_{ij} = |E| \\
&\text{number of } k\text{-stars } (k\ge 2): && S_k(x) = \sum_{i=1}^{n-1}\binom{i}{k} D_i(x) \\
&\text{number of triangles:} && T(x) = \sum_{1\le h<i<j<\le n} x_{hi}x_{ij}x_{hj}
\end{aligned}$$

and the parameters are $\{\theta_k\}_{k=1}^n$ and τ.[1]

2.2 Sufficient Statistics of ERGMs

In practice, due to its simplicity, the $G(n,p)$ model is used only as a starting point in inference over real-world data, while the full Markov graph model is infrequently used as it suffers from poor statistical properties. In particular, the Markov graph model is degenerate for many parameter configurations, representing only distributions that put all of their probability mass on either nearly-complete graphs (graphs with all edges present) or on $G(n,p)$ graphs [Jon99]. In response to these problems of degeneracy with Markov graphs, more robust "alternating" sufficient statistics are generally used in ERGMs to capture structural properties of networks. We will first provide definitions of these statistics and then expand on mathematical motivation behind them.

2.2.1 Alternating Sufficient Statistics

Definition 2.2 (Alternating k-star statistic [SPRH06])**.** The *alternating k-star* statistic on graph x with weighting parameter $\lambda \ge 1$ is defined as

$$\begin{aligned}
u_\lambda^{(s)}(x) &= S_2 - \frac{S_3}{\lambda} + \frac{S_4}{\lambda^2} - \cdots + (-1)^{n-2}\frac{S_{n-1}}{\lambda^{n-3}} \\
&= \sum_{k=2}^{n-1}\frac{S_k}{\lambda^{k-2}}
\end{aligned}$$

[1]Note that setting $\theta_2 = ... = \theta_k = \tau = 0$ in the Markov model, we recover the $G(n,p)$ model, which is an instance of a Markov graph since any two edges are conditionally independent in the $G(n,p)$ model.

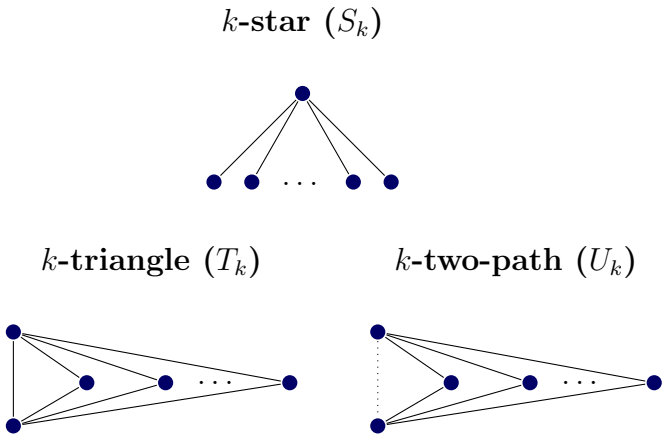

Figure 2.1: Subgraphs used in sufficient statistics of ERGMs.

We introduce the notion of "shared partners" of two nodes – the number of common neighbors that two nodes share – which give a clean way to count k-triangles and k-two-paths.

Definition 2.3 (Shared partners)**.** We denote the *shared partner count* of nodes i and j:

$$P_{ij}(x) = \sum_{\ell \in V} x_{i\ell} x_{j\ell} \tag{2.1}$$

We define *k-triangles* analogously to k-stars, so that a k-triangle consists of k triangles that all share an edge. We can count the total number of k-triangles in a graph using the number of shared partners:

$$T_k(x) = \sum_{1 \le i < j \le n} x_{ij} \binom{P_{ij}}{k} \quad \text{for } (k \ge 2), \quad \text{and } T_1 = \frac{1}{3} \sum_{1 \le i < j \le n} x_{ij} P_{ij} \tag{2.2}$$

Definition 2.4 (Alternating k-triangle statistic [SPRH06])**.** The *alternating k-triangle* statistic on graph x with weighting parameter $\gamma \ge 1$ is defined as

$$\begin{aligned} u_\gamma^{(t)}(x) &= 3T_1 - \frac{T_2}{\gamma} + \frac{T_3}{\gamma^2} - \cdots + (-1)^{n-3} \frac{T_{n-2}}{\gamma^{n-3}} \\ &= 3T_1 + \sum_{k=2}^{n-2} \left(\frac{-1}{\gamma} \right)^{k-1} T_k \end{aligned}$$

We define an *independent k-two-path* as a pair of nodes (possibly connected or unconnected) with k paths of length 2 connecting them. We can think of a k-two-path as a precondition for a k-triangle, since every k-triangle must contain an independent k-two-path. In terms of shared partners, independent k-two-paths can be represented as:

$$U_k(x) = \sum_{1 \le i < j \le n} \binom{P_{ij}}{k} \text{ for } k \ne 2 \quad \text{and } U_2(x) = \frac{1}{2} \sum_{1 \le i < j \le n} \binom{P_{ij}}{2} \tag{2.3}$$

Definition 2.5 (Alternating k-two-path statistic [SPRH06])**.** The *alternating k-two-path* statistic on graph x with weighting parameter $\gamma \geq 1$ is defined as

$$\begin{aligned} u_\gamma^{(p)}(x) &= U_1 - \frac{2U_2}{\gamma} + \frac{U_3}{\gamma^2} - \cdots + (-1)^{n-3}\frac{U_{n-2}}{\gamma^{n-3}} \\ &= U_1 - \frac{2U_2}{\gamma} + \sum_{k=3}^{n-2}\left(\frac{-1}{\gamma}\right)^{k-1} U_k \end{aligned}$$

Now, having defined the "alternating" sufficient statistics, the proposed model has the form

$$\Pr(x|\theta) = \exp\left\{\theta_1 E(x) + \theta_2 u_\lambda^{(s)}(x) + \theta_3 u_\gamma^{(t)}(x) + \theta_4 u_\gamma^{(p)}(x) - \psi(\theta)\right\} \tag{2.4}$$

where $E(x)$ is the number of edges in graph x, the alternating k-two-path and k-triangle statistics generally use the same weighting parameter γ. In practice, a subset of the sufficient statistics can be used in the model, depending on what properties of a graph are pertinent to model for a given network.

The overarching motivation behind introducing "alternating" sufficient statistics of the ERGMs is that these statistics are robust to addition or removal of an edge adjacent to an individual node, alleviating degeneracies in the Markov graph model. For instance, consider adding an edge to a high degree node with degree k. This new edges contributes one $(k+1)$-star, $\binom{k}{k-1}$ k-stars, $\binom{k}{k-2}$ $(k-1)$-stars and so on. Therefore, the total number of additional stars in the graph resulting from adding this edge is $\sum_{i=0}^{k}\binom{k}{i} = 2^k$. For Markov graphs including all stars with arbitrary associated parameters, this could lead to a large increase (or decrease) in the likelihood of the graph making the model degenerate as it places almost all of its probability on either near-complete or near-empty graphs. However, by imposing constraints on the parameters θ_k, namely by alternating the signs of the k-star statistics, the additional $(k-1)$-stars and k-stars balance each-other out. The same general reasoning applies to the use of alternating statistics for k-triangles and k-two-paths – alternation prevents the probability distribution from putting all of its mass on graphs with many high degree nodes, preventing degeneracy of the model.

This interpretation of alternating statics as limiting the sensitivity of the likelihood to addition or removal of edges to high degree nodes can be understood by looking at an alternative representation of the statistics in terms of the degree distribution and the number of shared partners for nodes. Below, we present these equivalent representations of the statistics, which will also be helpful in proofs of privacy in Chapter 4.

Alternating k-star

Note that using the relationship between k-stars and degrees given in Example 2.2 along with the binomial theorem we can rewrite the *alternating k-star* statistic as:

$$u_\lambda^{(s)}(x) = \sum_{i=1}^{n-1} D_i(x) \sum_{k=2}^{n-1} \left(\frac{-1}{\lambda}\right)^{k-2} \binom{i}{k}$$
$$= \lambda^2 \sum_{i=0}^{n-1} \left(\frac{\lambda-1}{\lambda}\right)^i D_i + 2\lambda|E| - n\lambda^2 \tag{2.5}$$

The alternating k-star statistic is thus made up of the number of edges as well as a linear combination of the degree sequence where lower degree nodes are up-weighted exponentially compared to higher degree nodes, reflecting the tendency towards a power law degree distribution. Since a term representing the number of edges in the network is generally included along with this statistic, the model is mathematically equivalent to a model using a geometrically weighted average of the degree sequence. Sociologically, the coefficient of the k-star statistic can thus be interpreted as the propensity for high degree nodes in the network. If the coefficient of the statistic is positive, then networks with a few high degree "hubs" are observed, while if it is negative, high degree nodes are discouraged and the network consists of mostly low-degree nodes [SPRH06].

Alternating k-triangle

Similarly, for the alternating k-triangle statistic, we can gain insight by rewriting in terms of the number of shared partners for pairs of nodes. By using this representation of k-triangles from Equation (2.2) along with the binomial theorem, we can rewrite the *alternating k-triangle* statistic as:

$$u_\gamma^{(t)}(x) = \sum_{1\le i<j\le n} x_{ij} \sum_{k=1}^{n-2} \left(\frac{-1}{\gamma}\right)^{k-1} \binom{P_{ij}}{k}$$
$$= \gamma \sum_{1\le i<j\le n} x_{ij} \left(1 - \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}}\right)$$
$$= \gamma|E| - \gamma \sum_{1\le i<j\le n} x_{ij} \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}} \tag{2.6}$$

Note, then, that any edge that does not participate in a triangle (so $x_{ij} = 1$ but $P_{ij} = 0$) does not contribute to the the alternating k-triangle statistic. On the other hand, as we add additional shared partners to an edge, the second term in (2.6) falls exponentially so the statistic increases, but by less for higher order k-triangles than for lower-order triangles. Sociologically, this term can be interpreted as the importance of triangle closure in the generation of the graph [GKM09]. In contrast to directly including the number of triangles in the graph, the alternating k-triangles statistic is more stable, preventing the model degeneracies discussed above.

Alternating k-two-path

Using the representation of k-two-paths in terms of shared partners from Equation (2.3) and the binomial theorem, we can rewrite the *alternating k-two-path* statistic as:

$$\begin{aligned} u_\gamma^{(p)}(x) &= \sum_{1\le i<j\le n} \sum_{k=1}^{n-2} \left(\frac{-1}{\gamma}\right)^{k-1} \binom{P_{ij}}{k} \\ &= \gamma \sum_{1\le i<j\le n} \left(1 - \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}}\right) \\ &= \gamma\binom{n}{2} - \gamma \sum_{1\le i<j\le n} \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}} \end{aligned} \tag{2.7}$$

Thus, the alternating k-two-path has an interpretation similar to that of the alternating k-triangle. As shared partners are added for any two nodes, the second term of the statistic increases, but the increase falls exponentially with additional partners. This term is generally only included in conjunction with the k-triangle statistic to try to separate out the effects of two-paths forming between unconnected nodes and mutual connections forming between already connected nodes.

2.2.2 Sufficient Statistics for Labeled Nodes

The alternating statistics over k-stars, k-triangles and k-two-paths capture structural properties of network data. Frequently, however, there labels associated with nodes in the network, which are important to model. ERGMs can take into consideration both the structure of the network and the labels associated with nodes, by including sufficient statistics based off of the labels, allowing researchers to capture properties like homophily, the tendency for similar actors to build relationships with one another within a network. Generally, labels are taken to be fixed and exogenous to the edges, so that attributes of the nodes may affect the formation of the network, whereas relationships in the network are not thought of as impacting attributes. This is generally a reasonable assumption, as labels often represent the identity of an individual, containing characteristics like gender, race, or age. As there are many potential ways to incorporate labeled data into an ERGM, we will focus here on three of the most commonly used statistics for discrete nodal attributes, "homophily", "popularity" and "mixing."[2]

In particular, letting z_i be a discrete attribute of node i (gender, for instance) we introduce the following sufficient statistics to represent different processes of social selection [LKR12]:

[2]These have fairly straightforward analogues for continuous nodal attributes, but we focus on the discrete case, as this is applicable to the dataset analyzed.

Table 2.1: Common sufficient statistics for discrete nodal attributes.

Parameter	*Statistic*
Homophily (Uniform)	$\sum_{i<j} x_{ij}\mathbb{I}(z_i = z_j)$
Homophily (Differential)	$\sum_{i<j} x_{ij}\mathbb{I}(z_i = z_j = a)$
Popularity	$\sum_{i<j} x_{ij}\left(\mathbb{I}(z_i = a) + \mathbb{I}(z_j = a)\right)$
Mixing	$\sum_{i<j} x_{ij}\mathbb{I}(z_i = a)\mathbb{I}(z_j = b)$

Uniform homophily captures the tendency for nodes with the same attribute to share an edge, while differential homophily captures this phenomenon for a specific attribute, which may be useful if, for instance, we thought that men and women have different propensities to become friends with people of the same gender. The popularity parameter is fairly self-explanatory as it measures the number of edges that have nodes with a given attribute as an endpoint and can be thought of as the overall sociability of a group with a specific attribute. Finally, the mixing parameter represents the number of edges between nodes with two different, specific attributes. Including such nodal attribute statistics in conjunction with the alternating sufficient statistics discussed in Section 2.2 allows for specification of ERGMs that separate out social selection effects like homophily from structural effects like triangle closure, making ERGMs a powerful modeling tool.

2.3 Bayesian Inference on ERGMs

Having provided an ERGM specification that captures the characteristics of interest in a network, the goal of inference is to find parameters θ that describe the realized data well. In the framework of maximum likelihood estimation, this means finding a θ that maximizes the probability of drawing observed network x_{obs} from the distribution $p(X|\theta)$. In the Bayesian paradigm, an analyst specifies a prior distribution over θ and then wishes to compute a posterior distribution of θ given the observed network. Bayesian inference is more general than maximum likelihood inference in the sense that if an analyst chooses a flat prior on θ (a uniform prior over the parameter space) and takes the maximum of the posterior as a point estimate, then Bayesian inference reduces to maximum likelihood inference.

In general, exact inference is not feasible for ERGMs due to the presence of the intractable normalizing constant $\psi(\theta)$ in the likelihood (Definition 2.1), which is a sum over the space of possible graphs on n nodes of size $2^{\binom{n}{2}}$. Therefore, a number of approximate MCMC approaches have been proposed to perform inference. In this work, we focus on Bayesian inference over ERGMs, because it constitutes the state-of-the-art in non-private inference methods and has been shown to be more stable than MCMC-MLE approaches

[CF11]. Additionally, the noise from differentially private mechanisms can be incorporated quite naturally into the Bayesian framework. The non-private Bayesian inference method proposed by Caimo and Friel is based on the Exchange Algorithm[MGM12] and is fairly simple to describe:

Algorithm 1 Non-Private Bayesian Inference for ERGMs (Exchange Algorithm) [CF11]

Input: ERGM distribution $\pi(X|\theta)$, prior $p(\theta)$, observed graph x_{obs}, number of burn-in draws r, symmetric proposal distribution $h(\cdot|\theta)$.
Output: sequence of draws $\theta^{(r)}, ...\theta^{(T)}$ from posterior distribution $p(\theta|x_{obs})$.

For $t = 1, ..., T$:

1. Draw parameter vector $\theta^* \sim h(\cdot|\theta^{(t-1)})$
2. Sample graph $x^* \sim \pi(\cdot|\theta^*)$
3. Accept the proposed move with probability $\min\{1, \alpha\}$. If the move is accepted, set $\theta^{(t)} = \theta^*$. Otherwise, set $\theta^{(t)} = \theta^{(t-1)}$

where

$$\alpha = \frac{p(\theta^*)}{p(\theta^{(t-1)})} \exp\left\{\left(\theta^* - \theta^{(t-1)}\right)^T \left(u(x_{obs}) - u(x^*)\right)\right\}$$

The algorithm can be justified by considering sampling from an augmented distribution with two auxiliary variables x^*, θ^*:

$$p(x^*, \theta^*, \theta|x_{obs}) \propto \pi(x_{obs}|\theta)p(\theta)h(\theta^*|\theta)\pi(x^*|\theta^*)$$

where π refers to the ERGM probability distribution. Marginalizing out θ^* and x from the augmented distribution gives the posterior distribution $p(\theta|x_{obs})$ of interest. Steps 1 and 2 are Gibbs updates of θ^* and x^*, while step 3 can be justified as the appropriate Metropolis-Hastings acceptance ratio:

$$\begin{aligned}\alpha &= \frac{\pi(x_{obs}|\theta^*)p(\theta^*)h(\theta^{(t-1)}|\theta^*)\pi(x^*|\theta^{(t-1)})}{\pi(x_{obs}|\theta^{(t-1)})p(\theta)h(\theta^*|\theta^{(t-1)})\pi(x^*|\theta^*)} \\ &= \frac{p(\theta^*)}{p(\theta)} \frac{\pi(x_{obs}|\theta^*)\pi(x^*|\theta^{(t-1)})}{\pi(x_{obs}|\theta^{(t-1)})\pi(x^*|\theta^*)}\end{aligned}$$

where we drop the h transition probabilities by symmetry and the intractable normalizing constants for θ^* and $\theta^{(t-1)}$ cancel, allowing easy computation of α. Thus, by standard MCMC theory, the draws $\theta^{(t)}$ come asymptotically from the desired posterior distribution.

In practice, Caimo and Friel advocate the use of a population-MCMC variant of their basic algorithm, in which multiple Markov chains are run in parallel, with the state space defined over the θ's of these multiple chains, as this population MCMC approach tends to converge faster and lead to less temporal dependence in draws from the Markov chain. We use this method, known as Parallel Adaptive Direction Sampling, in our private inference methods and explain it in detail in Appendix A along with a Metropolis-Hastings sampler to simulate networks from an ERGM with specified parameters.

Chapter 3: Differential Privacy over Networks

We employ the framework of differential privacy to protect individuals' data while analyzing network data. First, we provide the basic definition of differential privacy and mechanisms that meet this definition for general datasets, then explain specific problems that arise in applying these general definitions to network data, and finally detail the machinery of "restricted sensitivity" which we propose to use for differentially private inference over ERGMs.

3.1 Basics of Differential Privacy

Definitions

Let $\mathcal{D}$ denote the space of all possible datasets. Then:

Definition 3.1. Two datasets $x, x' \in \mathcal{D}$ are *adjacent*, written as $x \sim x'$, if they differ in the record of one individual. For tabular data, this means that the datasets differ in a single row.

Definition 3.2. The *distance* between two datasets $x, x' \in \mathcal{D}$, denoted $d(x, x')$ is the minimum length of the sequence of datasets beginning with x and ending with x' such that every two consecutive datasets on the path are adjacent. So, two datasets are clearly adjacent, or neighboring, if $d(x, x') = 1$.

Definition 3.3 (ϵ-differential privacy [DMNS06])**.** Let $\mathcal{A}$ be an algorithm over datasets in $\mathcal{D}$. Then $\mathcal{A}$ is *ϵ-differentially private* if for all $S \subseteq \text{Range}(\mathcal{A})$ and for every pair of neighboring datasets $x, x' \in \mathcal{D}$,

$$\Pr[\mathcal{A}(x) \in S] \leq e^{\epsilon} \Pr[\mathcal{A}(x') \in S]$$

Intuitively, differential privacy promises that the participation of any individual in a dataset does not significantly change the outcome of an analysis run on the dataset, limiting the potential harm (or benefit) to a data provider due to the inclusion of her data. Smaller values of ϵ correspond to stronger guarantees of privacy where $\epsilon = 0$ suggests that the algorithm does not learn anything from the data and therefore the algorithm is useless. Additionally, it is clear that no non-trivial deterministic algorithm satisfies ϵ-differential privacy for any value of ϵ. If $\mathcal{A}$ is deterministic and its output differs on at least two datasets, then there must be neighboring datasets such that the probability of a specific output is 0 on one dataset and 1 on the other, preventing the ratio between probabilities on this response from being bounded as required. Therefore, mechanisms that provide differential privacy will have to introduce some randomness, or noise, into their answers.

We can relax the definition of ϵ-differential privacy to allow for a small probability of potentially catastrophic privacy leakage:

Definition 3.4 ((ϵ, δ)-differential privacy [DMNS06])**.** $\mathcal{A}$ is (ϵ, δ)-*differentially private* if for all $S \subseteq \text{Range}(\mathcal{A})$ and for every pair of neighboring datasets $x, x' \in \mathcal{D}$,

$$\Pr[\mathcal{A}(x) \in S] \leq e^{\epsilon} \Pr[\mathcal{A}(x') \in S] + \delta$$

It is immediate from the definition that (ϵ, δ)-differential privacy is equivalent to ϵ-differential privacy when $\delta = 0$. For $\delta > 0$, however, (ϵ, δ) guarantees that the mechanism is ϵ-differentially private with probability $1 - \delta$, but makes no promises about the privacy loss that occurs with probability δ. Therefore, if δ is on the order of $\frac{1}{n}$ where n is the size of the dataset, it is possible to satisfy (ϵ, δ)-DP by releasing a row of the data. Further, a mechanism that sometimes releases the entire dataset still satisfies (ϵ, δ)-DP.

Example 3.1. An algorithm that selects at random one record in the dataset and exactly releases this record is $(\epsilon, \frac{1}{n})$-differentially private for any value of ϵ.

Example 3.2. An algorithm that releases the entire dataset with probability δ and a constant value with probability $1 - \delta$ is (ϵ, δ)-differentially private for any value of ϵ.

As these examples demonstrate, (ϵ, δ)-differential privacy only provides meaningful privacy for values of δ much smaller than $\frac{1}{n}$. In particular, ϵ should be taken to be "cryptographically small" (e.g. take $\delta = \frac{1}{1{,}000{,}000}$ for networks over a few 100 nodes.)

Properties

One of the desirable properties of differential privacy is its immunity to *post-processing* – armed with the output of a differentially private mechanism, an analyst cannot degrade privacy any further without additional information about the private dataset. In the context of inference over ERGMs, this property suggests that after computing sufficient statistics of a model in a differentially private manner, inference using these sufficient statistics can be thought of as a post-processing step that does not further degrade privacy. Formally:

Property 1 (Post-processing [DMNS06])**.** If $\mathcal{A}$ is an (ϵ, δ)-differentially private algorithm, then for an arbitrary mapping f, $f \circ \mathcal{A}$ is also (ϵ, δ)-differentially private.

A second useful property of differential privacy is that multiple differentially private algorithms compose, so applying many differentially private algorithms to the same dataset still provides privacy, albeit with higher privacy loss. This allows for basic DP algorithms to be used as building blocks in more complicated algorithms and in particular to split a privacy budget across multiple private computations on the data. Specifically, basic composition states that the privacy loss incurred by running multiple DP algorithms on a dataset grows linearly:

Property 2 (Basic Composition [DMNS06])**.** Let $\mathcal{A}_i$ be an (ϵ_i, δ_i)-differentially private algorithm for $i \in [k]$. Then, the algorithm releasing the result of running all k algorithms on the dataset $\mathcal{A}_{[k]}(x) = (\mathcal{A}_1(x), ..., A_k(x))$ is $\left(\sum_{i=1}^{k} \epsilon_i, \sum_{i=1}^{k} \delta_i\right)$-DP.

Mechanisms

We now describe two simple mechanisms that satisfy differential privacy. First, we describe the Laplace Mechanism, which answers queries on a dataset in a differentially private manner by adding Laplace noise to queries. Then, we introduce Randomized Response, which provides privacy by randomly perturbing the underlying dataset.

Laplace Mechanism

We define a query to be a function mapping the dataset to a vector of real numbers, $f : \mathcal{D} \to \mathbb{R}^m$. Then the local sensitivity of a query on a dataset x is the maximum ℓ_1-norm of the difference in the query over neighbors of dataset x.

Definition 3.5 (Local sensitivity)**.** The *local sensitivity* of a query f on a dataset x is

$$LS_f(x) = \max_{x' \sim x} ||f(x) - f(x')||_1$$

The global sensitivity is the worst-case local sensitivity over all possible datasets:

Definition 3.6 (Global sensitivity)**.** The *global sensitivity* of a query f is

$$GS_f = \max_{x \in \mathcal{D}} LS_f(x)$$

A basic result in differential privacy is that adding Laplace noise scaled to the global sensitivity provides differential privacy:

Theorem 3.1 (Laplace mechanism [DMNS06])**.** *Let f be a query on dataset x with global sensitivity GS_f and let* Lap *denote the zero-mean Laplace distribution*[1]*. Then, the Laplace mechanism $\mathcal{A}_L$ that outputs*

$$\mathcal{A}_L(x, f, \epsilon) = f(x) + (Y_1, ..., Y_m)$$

where $Y_i \overset{i.i.d.}{\sim}$ Lap $\left(\frac{GS_f}{\epsilon}\right)$ is ϵ-differentially private.

Note that the Laplace mechanism scales noise to the *global sensitivity*. While it is tempting to calibrate noise to local sensitivity, this does not protect privacy, because the noise level may disclose information about the underlying dataset. However, we can add noise scaled to a smooth upper bound on the local sensitivity, namely a function S that is larger than the local sensitivity for all datasets and for which $\ln(S(\cdot))$ is not too sensitive. The smoothness is parameterized by β, where β depends on ϵ and δ:

[1]The Laplace distribution centered at 0 with scale parameter b has probability density function $p(x|b) = \frac{1}{2b}e^{-|x|/b}$ and the variance of the distribution is $\sigma^2 = 2b^2$.

Theorem 3.2 (Calibrating Noise to β-Smooth Upper Bound on Local Sensitivity [NRS07])**.** *A β-smooth upper bound on the local sensitivity of query f is a function $S_{f,\beta}$ that satisfies:*

(i) $S_{f,\beta}(x) \geq LS_f(x) \quad \forall x \in \mathcal{D}$

(ii) $S_{f,\beta}(x) \leq \exp\{-\beta d(x,x')\} S_{f,\beta}(x') \quad \forall x, x' \in \mathcal{D}$

It is possible to satisfy (ϵ, δ)-differential privacy by adding Laplace noise scaled to $\frac{2S_{f,\beta}(x)}{\epsilon}$ where $\beta = -\frac{\epsilon}{2\ln(\delta)}$ and $S_{f,\beta}$ is a β-smooth upper bound on $LS_f(x)$. It is possible to satisfy ϵ-differential privacy by adding Cauchy noise[2] scaled to $\sqrt{2}S_{f,\beta}(x)$ where $\beta = \epsilon/\sqrt{2}$.

Then, global sensitivity trivially satisfies this definition, but it is a very conservative bound on the local sensitivity. The smallest function S to satisfy the definition of a β-smooth upper bound is known as the *smooth sensitivity*:

Definition 3.7 (Smooth Sensitivity [NRS07])**.** For query f and dataset x, define the *local sensitivity at distance of t* to be

$$LS_f^{(t)}(x) = \max_{\substack{x' \in \mathcal{D}: \\ d(x,x') \leq t|}} LS_f(x')$$

Then the *smooth sensitivity* is

$$S_{f,\beta}^*(D) = \max_t e^{-t\beta} LS^{(t)}(D)$$

The smooth sensitivity is the smallest β-smooth upper bound on the local sensitivity in the sense that for any other β-smooth upper bound S, $S_{f,\beta}^*(D) \leq S(D)$ for all datasets D. Thus, if we can compute the smooth sensitivity efficiently, then we can potentially add much less noise by calibrating to smooth rather than global sensitivity.

Randomized Response

In contrast to the Laplace Mechanism, which perturbs the output of a query on a dataset, randomized response perturbs the underlying dataset by randomly introducing spurious data. A typical version of randomized response over binary data proceeds as follows:

For each bit in a dataset consisting of $\{0, 1\}$ values:

1. Flip a biased coin with probability p_1 of heads.
2. If tails, then record the bit truthfully.
3. If heads, then flip a second biased coin with probability p_2 of heads and record 1 if heads, 0 if tails.

[2]The Cauchy distribution with median 0 and scale paramter b has probability density function $p(x|b) = 1/(b\pi(1 + (x/b)^2))$. Roughly, the Cauchy distribution can provide ϵ-DP because it has fatter tails than the Laplace distribution.

A benefit of randomized response is that it can be employed while collecting data, by using the coin-flipping procedure to collect responses in a study. The method provides plausible deniability for respondents, so it may incentivize participation in surveys for sensitive information. It is easy to verify that taking $p_1 = 2p$ and $p_2 = \frac{1}{2}$ yields the following simpler description:

Theorem 3.3 (Binary Randomized Response [War65],[KKS17])**.** *Let $\mathcal{D} = \{0,1\}^n$ so $x \in \mathcal{D}$ consists of binary data. Then, randomized response flips each bit of x with probability $p \in (0, \frac{1}{2})$ and releases the resulting noisy bits. This process provides ϵ-differential privacy taking $p \geq \frac{1}{e^\epsilon + 1}$.*

We may also define randomized response for a dataset where each row is drawn from a general data universe $\mathcal{U}$:

Theorem 3.4 (General Randomized Response)**.** *Consider dataset $x \in \mathcal{D} = \mathcal{U}^n$ and an algorithm which with probability p for each row, replaces the row with an entry drawn uniformly at random from $\mathcal{U}$. Then, this algorithm is ϵ-differentially private, taking $p \geq \frac{|\mathcal{U}|-1}{e^\epsilon + |\mathcal{U}|-1}$.*

3.2 Edge-Level vs. Node-Level Adjacency

We now turn to the question of how to define "adjacency" for graphs, as opposed to tabular data. We will define graphs abstractly in terms of vertex sets and edge sets, rather than as adjacency matrices in this section, as it makes the definitions easier to specify and more intuitive. There are two reasonable and widely used definitions of adjacency, which provide privacy at very different granularities and thus may be appropriate in different circumstances:

Definition 3.8 (Edge-level adjacency)**.** We define two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ to be *edge-adjacent* if they have the same vertex set ($V_1 = V_2$) and they differ in only one edge ($|E_1 \triangle E_2| = 1$).

Differential privacy with respect to edge-adjacency protects the privacy of individual relationships between nodes. Thus, edge-level privacy could protect a Facebook friendship with a controversial political leader. However, privacy at the edge-level could not promise to prevent an adversary from discerning whether an individual has mostly Republican or Democratic friends on Facebook. Such concerns motivate a stronger definition of neighboring graphs:

Definition 3.9 (Node-level adjacency)**.** We define two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ to be *node-adjacent* if $G_1 - v_i = G_2 - v_i$ for some vertex v_i, where $G - v_i$ means deleting edges adjacent to node v_i.

An additional consideration in defining adjacent graphs is how to account for labeled nodes. In the node-level case, labels are protected since removing a vertex and replacing it with a different vertex suggests changing the labeling on that vertex. For edge-level

privacy, labels could be taken to be either public or private information. There may be cases where the only sensitive information is the edges in the graph, not the identities of nodes (for instance, in a public social network, where people's identities may be readily searchable online, while their friendships are kept private.) However, in many settings, it seems preferable to protect the labels in addition to the relationships. Thus, letting there be some labeling function associated with a network that specifies a vector of nodal attributes for each node $\ell : V \to \mathbb{R}^m$, we define edge-level adjacency for labeled networks as follows:

Definition 3.10 (Edge-level adjacency with private labels)**.** We define two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ with labeling functions ℓ_1 and ℓ_2 to be *edge-adjacent with private labels* if they have the same vertex set ($V_1 = V_2$) and either they differ in only one edge ($|E_1 \triangle E_2| = 1$) or differ in one label ($\ell_1(v) \neq \ell_2(v)$ for exactly one vertex v.)

3.3 Restricted Sensitivity

Node-level privacy constitutes a strictly stronger guarantee than edge-level privacy, but it is often much more difficult to perform accurate analysis under node-level privacy. For instance, consider computing the degree distribution on an n-node graph. The global sensitivity under edge-level adjacency is only 2, since the degree of two nodes will change by 1 due to the addition or removal of an edge. However, under node-level adjacency, removing or adding all edges to a node of degree $n-1$ would affect n entries of the degree distribution, so the global sensitivity is n and naive application of the Laplace mechanism would completely destroy the counts of the degree distribution. Furthermore, even under edge-level adjacency many statistics computed on networks have high global sensitivity. For instance, the count of triangles in a graph (which is used in the alternating k-triangle sufficient statistic in ERGMs) has global sensitivity $O(n)$ in the edge-level case, since a single edge could be the base of a triangle with each other node in the graph.

The high global sensitivity of many graph statistics is particularly problematic for sparse graphs, where the noise completely overwhelms the true statistics. This is especially troubling, because sparsity is a characteristic of many real world networks. For instance, Facebook has billions of users, but users tend to have on the order of 1000 friends or fewer. We can formalize the hypothesis that a graph is sparse by considering the *degree* of the graph, the maximum degree of any of its nodes. If we hypothesize that all the graphs under consideration have limited degree, then the global sensitivity might be much lower over these limited-degree graphs than over all graphs on n nodes. For example, considering the space of graphs with degree of at most $k << n$, the triangle count would have a much global sensitivity of $O(k)$ rather than $O(n)$ over the space of all graphs on n nodes.

If we were certain that the graphs under consideration always had limited degree, we could scale noise to the sensitivity over limited degree graphs. However, our hypothesis might be false, so adding noise assuming that the graph has limited degree would not protect privacy for an arbitrary graph. Therefore, it is necessary to first project the graph into the space of limited degree graphs. If the limited degree hypothesis is true then the projection will not alter the graph at all, so the analysis is accurate up to the distortion

of the noise-adding procedure. We formally define the *limited degree hypothesis* as the space of graphs on n nodes with degree k:

Definition 3.11 (Limited Degree Hypothesis)**.** Let $\mathcal{G}_n$ be the space of graphs on n nodes. Then, the limited degree hypothesis $\mathcal{H}_k$ is the set of graphs:

$$\mathcal{H}_k = \{G = (V, E) \in \mathcal{G}_n : \deg(v) \leq k, \forall v \in V\}$$

Then, the restricted sensitivity is the global sensitivity of the query restricted to limited degree graphs:

Definition 3.12 (Restricted sensitivity [BBDS13])**.** For a given notion of adjacency (either edge or node), we define the *restricted sensitivity* of query f over hypothesis $\mathcal{H}_k \in \mathcal{G}_n$ as

$$RS_f(\mathcal{H}) = \max_{\substack{G,G' \in \mathcal{H}_k: \\ G \sim G'}} ||f(G - f(G')||_1$$

To protect privacy over arbitrary graphs, while calibrating noise to the restricted sensitivity rather than the global sensitivity, we require a projection $\mu : \mathcal{G} \to \mathcal{H}_k$. We can define the sensitivity of the projection in terms of how much it changes the distance by a multiplicative factor between any two adjacent graphs. In particular:

Definition 3.13 (Local sensitivity of projection μ [KNRS13])**.** Define the local sensitivity of projection $\mu : \mathcal{G}_n \to \mathcal{H}_k$ on graph $G \in \mathcal{G}_n$ to be:

$$LS_\mu(G) = \max_{G' \sim G} d(\mu(G), \mu(G'))$$

Then, the global sensitivity and smooth sensitivity can be defined as before. Now, if we can find a projection μ, where it is possible to bound the global sensitivity by a small constant, so $\forall G \in \mathcal{G}_n : LS_\mu(G) \leq c$, then for any two neighboring graphs the effect of first projecting a graph to $\mathcal{H}_k$ before answering a query only increases global sensitivity by a multiplicative factor of c:

Lemma 3.1 (Global Sensitivity on Composed Functions)**.** *For projection $\mu : \mathcal{G}_n \to \mathcal{H}_k$ and query $f : \mathcal{G}_n \to \mathbb{R}^m$, define $f_{\mathcal{H}_k} = f \circ \mu$ to be the query applied to the projection. Then $GS_{f_{\mathcal{H}_k}} \leq GS_\mu \cdot RS_f(\mathcal{H}_k)$.*

In particular, this suggests that if we can find a projection $\mathcal{H}_k$ with low global sensitivity c, then using ϵ-differentially private mechanisms like the Laplace mechanism that calibrate noise to $c \cdot RS_f(\mathcal{H})$ can give significant accuracy gains over global sensitivity. Blocki et al. give such a projection for the edge-adjacency model with $GS_\mu = 3$ that is also efficient (linear in the number of edges in the graph.) We give the details of this projection in Appendix B.

In the node level-adjacency model an efficient projection with low global sensitivity is not known [KNRS13]. However, it can be shown that if we use the smooth sensitivity of μ, then multiplying this β-smooth upper bound by the restricted sensitivity of f gives a β-smooth bound on the local sensitivity of the composition $f_{\mathcal{H}_k}$ as above:

Lemma 3.2 (β-Smooth Bound on Composed Functions)**.** *Let $S_\mu(G)$ be a β-smooth upper bound on the local sensitivity of μ on graph $G \in \mathcal{G}_n$. Then $S_{f_{\mathcal{H}_k}} = S_\mu(G) \cdot RS_f(\mathcal{H}_k)$ is a β-smooth bound on the local sensitivity of $f_{\mathcal{H}_k} = f \circ \mu$.*

We detail two possible projections for the node-adjacency model in in Appendix B ([KNRS13], [BBDS13]). The first, which we refer to as μ_{trunc} simply removes nodes of high degree and the other, μ_{LP} solves a linear program. It is possible to give β-smooth upper bounds on the local sensitivity for each of these projections. Roughly speaking, the benefits of node truncation are that it is more efficient than the LP and has low smooth sensitivity when there are few nodes with degree close to the cutoff k, which we may assume since degree distributions often follow a power law. However, the smooth sensitivity of μ_{trunc} could be high for graphs in $\mathcal{H}_k$ if the graph does in fact have many nodes with degree close to the cutoff k. On the other hand, the smooth sensitivity of μ_{LP} is always relatively low when the hypothesis $\mathcal{H}_k$ is true, but the LP is not strictly a projection in that it is guaranteed to project graphs in $\mathcal{H}_k$ to themselves, but its image is $\mathcal{H}_{2k}$ not $\mathcal{H}_k$. Therefore, we must calibrate noise to the restricted sensitivity over $\mathcal{H}_{2k}$ when using the LP. Since, we expect the conditions under which μ_{trunc} has low smooth sensitivity to be met for network data in practice, it requires lower noise addition from restricted sensitivity and is more efficient than the linear program-based projection we propose using node truncation as the projection for node-adjacency model.

Then, taking advantage of restricted sensitivity over $\mathcal{H}_k$ and the appropriate projections, we can perform inference over ERGMs while adding relatively low noise to sufficient statistics that have high global sensitivity. Our primary focus in proving privacy will be bounding the restricted sensitivity of the queries of interest over $\mathcal{H}_k$ in order to take advantage of this machinery of restricted sensitivity.

Chapter 4: Private Inference on ERGMs

In this chapter, we propose methods for differentially private inference over ERGMs with the alternating and nodal attribute sufficient statistics defined in Section 2.2. We propose perturbing the sufficient statistics, taking advantage of restricted sensitivity to limit the amount of noise needed to protect privacy. Then, it is possible to perform Bayesian inference taking into account the level of noise added. Since the restricted sensitivity is public, we are able to do principled inference over the posterior incorporating the randomness of the privacy mechanism, which has been shown to lead to more reliable inference in many cases (see [FGWC16], [KS16], [KKS17], [LM14] for instance).

Our primary contribution is the proposal to take use the machinery of restricted sensitivity in adding noise to sufficient statistics. The advantages of employing restricted sensitivity for inference over ERGMs are threefold:

- Calibrating noise to restricted sensitivity enables *lower noise in the edge-adjacency model* than current methods, permitting accurate inference at lower privacy budgets.
- Restricted sensitivity permits private release of sufficient statistics under edge level privacy with *private labels*, whereas prior work has treated labels as public.
- By using restricted sensitivity, we suggest the first method to our knowledge that performs private inference under the *node-adjacency model*, a strictly stronger notion of privacy than in the edge-adjacency model.

Restricted sensitivity relies on the hypothesis that the graph we are analyzing is sparse, namely that its max degree node has degree k. There are a number of reasons to believe that the limited degree hypothesis $\mathcal{H}_k$ is a reasonable assumption when modeling real social network data with ERGMs. First, previous empirical analyses of ERGMs have demonstrated that for reasonable parameter values, the distribution tends to put low probability mass on high-degree graphs [SPRH06]. Thus, given that we assume that an observed network is roughly drawn from the probability distribution specified by an ERGM, we believe with high probability that the graph has relatively low degree. Second, many real-world social networks are fairly sparse and have bounded degree. Thus, an analyst is likely to believe that their network data represents a sparse graph and could reasonably choose a degree cutoff based on similar public datasets or domain knowledge.

4.1 Releasing Private Sufficient Statistics

In this section, we bound the restricted sensitivity under $\mathcal{H}_k$ of a number of the most commonly used sufficient statistics in ERGMs. As the following summary shows, in

the edge-adjacency case, restricted sensitivity is much lower than global sensitivity for alt-k-triangle and alt-k-two path, assuming $k << n$. In the node-level case, adding noise scaled to the global sensitivity overwhelms the computed statistics in most cases, motivating the need for restricted sensitivity. For labeled networks, the global sensitivity is very low if labels are considered public and only edges are taken to be private. However, if labels are private, then the restricted sensitivity is much lower than the global sensitivity for sparse graphs.

Table 4.1: Restricted Sensitivity on $\mathcal{H}_k$ for Common Structural Statistics

	Edge-Level		Node-Level	
	$RS_f(\mathcal{H}_k)$	GS_f	$RS_f(\mathcal{H}_k)$	GS_f
Edges	1	1	k	$n-1$
Alt k-star ($u_\lambda^{(s)}$)	2λ	2λ	$3\lambda k$	$O(n)$
Alt k-triangle ($u_\gamma^{(t)}$)	$2(k-1)+\gamma$	$O(n)$	$k^2+(\gamma-1)k$	$O(n^2)$
Alt k-two-path ($u_\gamma^{(p)}$)	$2(k-1)$	$O(n)$	k^2	$O(n^2)$

Table 4.2: Restricted Sensitivity on $\mathcal{H}_k$ for Common Statistics of Labeled Networks

	Public Labels	Private Labels	
	GS_f	$RS_f(\mathcal{H}_k)$	GS_f
Homophily	1	k	$n-1$
Popularity	2	$2k$	$2n$
Mixing	1	k	$n-1$

Below, we formally derive the restricted sensitivity of the alternating sufficient statistics of an ERGM under edge level privacy and node level privacy respectively. The "weighting parameters" of the alternating statistics γ and λ are generally set to be small constants between roughly 1 and 5 (most empirical work seems to find that values between 1 and 2 suffice) so the choice of this parameter has a fairly minor effect on the level of noise.

4.1.1 Edge Level Privacy

For the alternating k-star statistic under edge-level privacy, restricted sensitivity does not give any advantage over using global sensitivity, as the global sensitivity of this statistic is quite low:

Claim 4.1.1 (Global sensitivity of alternating k-star under edge-level privacy)**.** *The global sensitivity of the alternating k-star statistic is less than 2λ.*

Proof. We use the alternative formulation of the statistic given in Equation (2.5):

$$u_\lambda^{(s)}(x) = \lambda^2 \sum_{i=0}^{n-1} \left(\frac{\lambda-1}{\lambda}\right)^i D_i + 2\lambda|E| - n\lambda^2$$

Then, consider adjacent graphs x, x' differing in one edge where x has the additional edge. Then, the first term of the alternating k-statistic is larger for x' than for x and by at most 2λ and at least 0, while the second term is larger for x than for x' by 2λ. Hence, the difference between the alternating k-star statistic computed on x and x' is at most $|2\lambda - 0| = 2\lambda$. □

Claim 4.1.2 (Restricted sensitivity of alternating k-triangle under edge-level privacy)**.** *The restricted sensitivity of the alternating k-triangle statistic under $\mathcal{H}_k$ is less than $2(k-1)+\gamma$.*

Proof. Consider two adjacent graphs $x, x' \in \mathcal{H}_k$ differing in exactly one edge, so that $x_{ij} = 1$ and $x'_{ij} = 0$. Now, note that for nodes i and j, the number of shared partners is the same in x and x' since all edges are the same except for the edge between i and j. Then, let $P_{ij} = P'_{ij} = m \leq k-1$ by the limited degree hypothesis. Note that there are $2m$ edges for which $P'_e = P_e - 1$, since there are two other edges in each triangle. Then, recalling the definition of the alternating k-triangle statistic in terms of the shared partners of i and j given in Equation (2.6):

$$u_\gamma^{(t)}(x) = \gamma|E| - \gamma \sum_{1\leq i<j\leq n} x_{ij} \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}}$$

we have that

$$\begin{aligned}
|u_\gamma^{(t)}(x) - u_\gamma^{(t)}(x')| &= \left|\gamma - \gamma\left(\frac{\gamma-1}{\gamma}\right)^m + \gamma\sum_{e=1}^{2m}\left[\left(\frac{\gamma-1}{\gamma}\right)^{P_e-1} - \left(\frac{\gamma-1}{\gamma}\right)^{P_e}\right]\right| \\
&= \left|\gamma - \gamma\left(\frac{\gamma-1}{\gamma}\right)^m + \sum_{e=1}^{2m}\left(\frac{\gamma-1}{\gamma}\right)^{P_e-1}\right| \\
&\leq 2m + \gamma \\
&\leq 2(k-1) + \gamma
\end{aligned}$$

□

Note the usefulness of restricted sensitivity here, in contrast to global sensitivity. The global sensitivity of this statistic is $O(n)$, since in the worst case there could be a graph with an $(n-1)$-triangle where removing the base of the triangle leads to the removal of $O(n)$ triangles. However, if we restrict degrees, we add much less noise.

Claim 4.1.3 (Restricted sensitivity of alternating k-two-path under edge-level privacy)**.** *The restricted sensitivity of the alternating k-two-path statistic under $\mathcal{H}_k$ is less than $2(k-1)$.*

Proof. The proof will proceed in roughly the same way as for k-triangles. Define x and x' in the same way and recall the definition of the alternating k-two-path statistic in terms of shared partners as given in Equation (2.7):

$$u_\gamma^{(p)}(x) = \gamma\binom{n}{2} - \gamma \sum_{1\leq i<j\leq n} \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}}$$

Then, the change between the statistic on x and x' is equal to

$$|u_\gamma^{(p)}(x) - u_\gamma^{(p)}(x')| = \sum_{e=1}^{2m} \left(\frac{\gamma-1}{\gamma}\right)^{P_e - 1} \leq 2m \leq 2(k-1)$$

□

4.1.2 Node Level Privacy

Claim 4.1.4 (Restricted sensitivity of alternating k-star under node-level privacy)**.** *The restricted sensitivity with hypothesis $\mathcal{H}_k$ of alternating k-star under node-level differential privacy is less than $3\lambda k$.*

Proof. We will again use the formulation of the alternating k-star statistic in terms of degree distribution from Equation (2.5). Now, consider two graphs $x, x' \in \mathcal{H}_k$ differing in one node i of degree $m \leq k$, with all of its incident edges removed in x'. Then, the degree of node i is m in x and 0 in x', while the degrees of m other nodes are 1 lower in x' than in x, so:

$$\begin{aligned} |u_\lambda^{(s)}(x) - u_\lambda^{(s)}(x')| &= \left| 2\lambda m + \lambda^2 \left(\left(\frac{\lambda-1}{\lambda}\right)^m - 1 \right) + \sum_{j:x_{ij}=1} \lambda \left(\frac{\lambda-1}{\lambda}\right)^{d_j - 1} \right| \\ &\leq \left| 3\lambda m + \lambda^2 \left(\left(\frac{\lambda-1}{\lambda}\right)^m - 1 \right) \right| \end{aligned}$$

and note that $0 \leq \left(\frac{\lambda-1}{\lambda}\right)^m \leq 1$ and that $|\lambda^2| \leq 3\lambda m$ for reasonable choices of k and λ (since generally we choose $1 < \lambda < 5$, so in order to have the λ^2 term dominate the $3\lambda k$ term we would have to restrict k to 1, which would not be interesting or realistic, so the sensitivity is bounded by $3\lambda k$. □

Claim 4.1.5 (Restricted sensitivity of alternating k-triangle under node-level privacy)**.** *The restricted sensitivity with hypothesis $\mathcal{H}_k$ of the alternating k-triangle statistic under node-level differential privacy is less than $k^2 + (\gamma - 1)k$.*

Proof. Consider two adjacent graphs $x, x' \in \mathcal{H}_k$ differing in one node i of degree m. Now, since each of the m edges incident to node i is removed this changes m edges $x_{ij} = 1$ to $x'_{ij} = 0$, so $E(x) - E(x') = m$ and for each of these m edges

$$x_{ij} \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}} - x'_{ij} \left(\frac{\gamma-1}{\gamma}\right)^{P'_{ij}} = \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}}$$

so the direct effect of removing the x_{ij} is that $u_\gamma^{(t)}(x') - u_\gamma^{(t)}(x') \leq m\gamma - 0$ (ignoring the effect on the shared partners of edges not adjacent to i.)

Now, we consider edges e such that the endpoints of e have i as a shared partner. Note that there are $\binom{m}{2} = m^2 - m$ such edges, because we can choose any 2 edges of i and the

endpoints of these edges have i as a shared partner. Now, each of these edges still exists in x'_{ij} but has its number of shared partners decrease by 1. Then, we have

$$\begin{aligned}|u_\gamma^{(t)}(x) - u_\gamma^{(t)}(x')| &= \left|\gamma m - \gamma \sum_{j:x_{ij}=1} \left(\frac{\gamma-1}{\gamma}\right)^{P_{ij}} + \sum_{e=1}^{m^2-m} \left(\frac{\gamma-1}{\gamma}\right)^{P_e - 1}\right| \\ &\leq |\gamma m + (m^2 - m)| \\ &\leq k^2 + (\gamma - 1)k\end{aligned}$$

□

Claim 4.1.6 (Restricted sensitivity of alternating k-two-path under node-level privacy)**.** *The restricted sensitivity with hypothesis $\mathcal{H}_k$ of the alternating k-two-path statistic under node-level differential privacy is less than k^2.*

Proof. As for k-triangles, consider two adjacent graphs $x, x' \in \mathcal{H}_k$ differing in node i of degree m. Then, the removal of these m edges impacts the shared partners of m^2 edges, the m incident to i and the $\binom{m}{2} = m^2 - m$ that have i as a shared partner and the decrease in shared partners for each of these edges can change the statistic by at most 1 so the overall change is at most $m^2 \leq k^2$. □

4.1.3 Private Labels

If labels are considered public, then the global sensitivity of the sufficient statistics using discrete attributes of nodes given in Table 2.1 have low global sensitivity in the edge-adjacency model, since they are effectively counts of edges for nodes with certain attributes, and adjacent graphs have only a single edge changed and all labels kept the same. However, if labels are considered private then the change in a single label could change the count for all edges incident to that node, leading to very high global sensitivity of $O(n)$. By using restricted sensitivity, we can bound the sensitivity to be $O(k)$. Additionally, note that differential homphily and popularity are vectors of queries, with dimension the size of the number of attributes under consideration. However, these queries are structurally disjoint as a change in one attribute can only change the counts of two entries of the vector, making it easy to bound the ℓ_1-sensitivity of the entire vector. We give the proof for homophily, while the proofs for popularity and mixing follow from the same argument:

Claim 4.1.7 (Restricted sensitivity of homophily with private labels.)**.** *Both differential and uniform homophily have ℓ_1-restricted sensitivity k.*

Proof. Recall that for attributes $a_1, ..., a_m$ differential homophily is given by

$$\left(\sum_{i<j} x_{ij}\mathbb{I}(z_i = z_j = a_1), ..., \sum_{i<j} x_{ij}\mathbb{I}(z_i = z_j = a_m)\right)$$

Then, changing nodal attribute z_i from a to b changes the endpoint of at most k edges. If both endpoints of an edge had endpoint a, then this reduces the count of entry a in the

vector by 1, while if the endpoints of the edge were a and b to start with, this increases the count in entry b by 1. These cases are disjoint so the largest ℓ_1 difference in the vector is k.

For uniform homophily, it is easy to see that changing one label could change at most k edges and each edge is counted only once in uniform homophily, so the global sensitivity is 1. □

4.2 Inference Using Noisy Sufficient Statistics

Now, by projecting a network into $\mathcal{H}_k$ using the projections specified in Appendix B and then applying the Laplace mechanism (3.1), we can release the sufficient statistics of the ERGM in a differentially private manner by calibrating the noise of the Laplace mechanism to the restricted sensitivity. We could now release these sufficient statistics to analysts who wish to study the network, since the likelihood of the ERGM depends on the data only through the sufficient statistics. Using noisy statistics directly for standard inference techniques has been shown to lead to biased estimates, however, as the sufficient statistics may not even be graphical. Therefore, in the framework of Bayesian inference we want to compute the posterior over both the observed network and the privacy mechanism. In particular, letting $\tilde{y}$ be the "noisy network" defined by the application of our privacy mechanism to the true network we wish to compute the posterior:

$$p(\theta|\tilde{y}) \propto p(\tilde{y}|\theta)p(\theta) = \sum_x p(\tilde{y}|x)p(x|\theta)p(\theta) \tag{4.1}$$

where $p(\tilde{y}|x)$ is the privacy distribution defined by our mechanism, $p(x|\theta)$ is the ERGM distribution and $p(\theta)$ is the prior on θ which is specified by the analyst.

Then, along the lines of [LM14], it is simple to modify the Exchange Algorithm for non-private inference to draw from the posterior that incorporates the privacy distribution:

As in the non-private case, this method draws samples from the true posterior of interest as $T \to \infty$ by MCMC theory. Steps 1 and 2 can be justified as Gibbs updating steps as in the non-private case, while steps 3 and 4 are component-wise Metropolis-Hastings updates , where we update the variables separately rather than in a block, because this tends to lead to higher acceptance ratios and thus faster convergence [GL06], [LM14]. Intuitively, x^* can be thought of as our best guess of the true underlying network. Then, θ^* is replaced in step 3 if it explains this best guess of the network well, while x^* is updated if the new network is likely to be the true network over the noise of the privacy mechanism. Additionally, we propose using the population MCMC version of the exchange algorithm, as this leads to better convergence in practice and still converges to the correct posterior.

Note that α_1 does not depend on the choice of privacy mechanism, while α_2 is simple to compute under the addition of Laplace noise. In particular, if we add Laplace noise scaled to L (for instance, $L = 3 \cdot RS_f(\mathcal{H}_k)$ in the edge-adjacency case) to the sufficient statistics of the network then:

Algorithm 2 Bayesian Inference for ERGMs with Differentially Private Network Data

Input: ERGM distribution $\pi(X|\theta)$, prior $p(\theta)$, noisy network $\tilde{y}$, privacy distribution $\pi_p(\tilde{y}|y)$, number of burn-in draws r, symmetric proposal distribution $h(\cdot|\theta)$.
Output: sequence of draws $\theta^{(r)}, ...\theta^{(T)}$ from posterior distribution $p(\theta|\tilde{y})$.

For $t = 1, ..., T$:

1. Draw parameter vector $\theta^* \sim h(\cdot|\theta^{(t-1)})$
2. Sample graph $x^* \sim \pi(\cdot|\theta^*)$
3. Replace $\theta^{(t-1)}$ with θ^* with probability $\min\{1, \alpha_1\}$.
4. Replace $x^{(t-1)}$ with x^* with probability $\min\{1, \alpha_2\}$.

where

$$\alpha_1 = \frac{p(\theta^*)}{p(\theta^{(t-1)})} \exp\left\{\left(\theta^* - \theta^{(t-1)}\right)^T \left(u(x^{(t-1)}) - u(x^*)\right)\right\}$$
$$\alpha_2 = \frac{\pi_p(\tilde{y}|x^*)}{\pi_p(\tilde{y}|x^{(t-1)})} \exp\left\{\left(\theta^* - \theta^{(t-1)}\right)^T \left(u(x^{(t-1)}) - u(x^*)\right)\right\}$$

$$\log \frac{\pi_p(\tilde{y}|x^*)}{\pi_p(\tilde{y}|x^{(t-1)})} = \log\left(Lap\left(\tilde{y} - x^*|L\right)\right) - \log\left(Lap\left(\tilde{y} - x^{(t-1)}|L\right)\right) \tag{4.2}$$
$$= \frac{|\tilde{y} - x^{(t-1)}|}{L} - \frac{|\tilde{y} - x^*|}{L}$$

In the edge-adjacency model, the restricted sensitivity is public,so we can easily compute the ratio in *eq.* (4.2) using $L = 3RS_f(\mathcal{H}_k)$ where the factor of 3 comes from the global sensitivity of the projection. In the node-level case, we compute smooth sensitivity of the projection which cannot be publicly released, but for μ_{LP} we know that the $L = g(\frac{\beta}{4})RS_f(\mathcal{H}_{2k})$ assuming that the hypothesis is true, so we can use this L as the noise level and the inference is still valid if the network does have degree less than k. For the node truncation projection, μ_{trunc}, we have that $L = (1 + \ell + \frac{1}{\beta})RS_f\mathcal{H}_{\hat{k}}$ assuming that there are only ℓ nodes with degree in range $\hat{k}$, where the analyst can set $\hat{k}$ to appropriately trade off $\hat{k}$ and ℓ and then use public L based off of these assumptions.

In general, we assume that the number of nodes in a graph is known and public. For the case where labels are public, sampling a graph in step 2 is straightforward as we sample a graph from the space of all possible graphs with the n labeled nodes of the original graph. However, if labels are private, then these labels must be privatized as well. This is straightforward to by releasing a noisy histogram of the labels, which has global sensitivity of 1 and therefore is high accuracy assuming that there are a limited number of types of labels [DMNS06]. Then, this noisy histogram can be used as the node-set over which graphs are sampled in step 2 of the algorithm.

The Full Workflow

Putting together our bounds on restricted sensitivity and inferential procedure the workflow for differentially private inference looks roughly as follows, given hypothesis $\mathcal{H}_k$, privacy budget ϵ, δ (where the δ is necessary only for the node-level case using Laplace noise) and network data y:

1. Split privacy budget between sufficient statistics under consideration (and potentially using some of the privacy budget for a histogram of private labels).
2. Project y to $\mathcal{H}_k$ using smooth projections μ specified for edge and node level privacy respectively in Appendix B.
3. Compute and release restricted sensitivity of sufficient statistics.
4. If labels are considered private, then release noisy histogram of node labels.
5. Draw noise according to restricted sensitivity (Laplace noise scaled to $3RS_f(\mathcal{H}_k)/\epsilon$ in the edge-level case and $S_{\mu,-\epsilon/2\ln\delta}RS_f(\mathcal{H}_k)$ in the node-level case where μ is the node-truncating projection. Add this noise to sufficient statistics and release these noisy sufficient statistics.
6. Using the noisy sufficient statistics from step 4 and the restricted sensitivity levels from step 3, perform inference using Algorithm 2.

Then, privacy follows by applying composition in step 1, restricted sensitivity with the Laplace Mechanism in steps 2-5 and post-processing in step 6. Post-processing is particularly useful here, because MCMC methods frequently require tuning of the inference, whereby we run the inferential procedure multiple times and run diagnostics to make sure it converges (for instance, by checking that every 100 samples from the posterior are not highly correlated.) By post-processing, we can run step 6 an arbitrary number of times to tune the inferential procedure because differential privacy is provided by steps 1 to 5.

4.3 Related Work

Our work builds on two proposed methods, both of which only consider the edge-adjacency privacy model with labels taken to be public. The method most closely related to our work is that of Lu and Miklau [LM14] who also suggest adding noise to sufficient statistics and then performing Bayesian inference. In order to avoid adding noise scaled to the high global sensitivity of these statistics, they calibrate noise to a private bound on the local sensitivity of the network. In particular, they take advantage of the approach suggested in [KRSY14] of computing an upper bound on the local sensitivity $\hat{L} = LS_f(G) + \text{Lap}(1/\epsilon) + \ln(1/\delta)/\epsilon$ and then adding Laplace noise calibrated to $\hat{L}/\epsilon$, which provides $(2\epsilon, \delta)$-differential privacy. This suggests that in their method, the expected noise level (over the addition of Laplace noise to the noise level) is

$$\frac{2LS_f(G)}{\epsilon} + \frac{4\ln(1/\delta)}{\epsilon^2}$$

whereas we add noise scaled to $3RS_f(\mathcal{H}_k)/\epsilon$. For reasonable choice of k, the restricted sensitivity is close to the local sensitivity on graph G, while the second term can be quite large for small privacy budgets in ϵ and δ. Miklau and Lu test their approach with $\delta = 0.1$, which as discussed is an unreasonable choice of this parameter in practice, since a method that released the entire dataset one tenth of the time would satisfy (ϵ, δ)-DP with this parameter. We find through a battery of tests that our proposed method (which takes $\delta = 0$) adds much less noise than the bounding of local sensitivity approach, especially for small privacy budgets. This difference in the magnitude of noise makes a significant difference in the accuracy of inference, as our method can perform accurate inference for realistic privacy budgets whereas using bounding of local sensitivity for small privacy budgets, accuracy is severely degraded. While our approach does require a reasonable estimate of the the maximum degree of the network, if the graph is not relatively sparse, bounding local sensitivity may still add significant noise as well. Further, Miklau and Lu's approach does not extend well to the node-adjacency model, because the local sensitivity of a graph could be much higher than the restricted sensitivity, since any graph is adjacent to a graph with a node of degree $n - 1$. Lastly, it is worth noting that the Bayesian inferential method using the exchange algorithm that we employ is based on Lu and Miklau's method, although we extend it to take advantage of population MCMC described in Appendix A which leads to more stable inference in practice.

Another approach from Karwa et. al. [KKS17], suggests using randomized response on edges of the underlying network and then performing maximum likelihood estimation on this network taking into account the perturbation on edges. The main benefit of this method is that it potentially permits greater flexibility as the perturbed network can be released for public use and researchers can use any sufficient statistics they like. Our method requires a commitment to use a specific set of sufficient statistics, although we bound restricted sensitivity for a fairly broad range of the most commonly used sufficient statistics in ERGMs. The primary drawback of the randomized response approach is that for low privacy budgets, it leads to extensive distortion of the underlying network. For instance, taking $\epsilon = 1$ suggests a probability of flipping each edge of around 25%, which for a sparse network in which only 5% of edges are present may completely overwhelm network structure. We test randomized response, using Bayesian inference, as it is straightforward to account for this noise in the Bayesian inferential framework and find, as expected, that their method only works for relatively large privacy budgets, whereas our proposal works for lower, more realistic privacy budgets. It is unclear what the benefits and drawbacks of their maximum likelihood inferential approach compared to the Bayesian approach over the distribution incorporating the noise of the privacy mechanism. However, noisy sufficient statistics released using restricted sensitivity could be used in their proposed MLE inference method.

Chapter 5: Empirical Evaluation of Private Inference

Because the convergence properties and accuracy of non-private inference methods for inference over ERGMs are primarily understood from an experimental standpoint, we propose experimental evaluation of methods for inference over private sufficient statistics.

Chapter 6: Conclusion

References

[AGZF09] E. M. Airoldi, A. Goldenberg, A. Zheng, and S. Fienberg, "A survey of statistical network models", eng, *Machine Learning -Boston-*, vol. 2, no. 2, 2009, ISSN: 0885-6125.

[BBDS13] J. Blocki, A. Blum, A. Datta, and O. Sheffet, "Differentially private data analysis of social networks via restricted sensitivity", eng, in *Proceedings of the 4th conference on innovations in theoretical computer science*, ser. ITCS '13, ACM, Jan. 2013, pp. 87–96, ISBN: 9781450318594.

[CF11] A. Caimo and N. Friel, "Bayesian inference for exponential random graph models", eng, *Social Networks*, vol. 33, no. 1, pp. 41–55, 2011, ISSN: 0378-8733.

[DMNS06] C. Dwork, F. Mcsherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis", in *Proceedings of the 3rd Theroy of Cryptography Conference*, Springer, 2006, pp. 265–284.

[FGWC16] J. Foulds, J. Geumlek, M. Welling, and K. Chaudhuri, "On the theory and practice of privacy-preserving bayesian data analysis", Mar. 2016.

[FS86] O. Frank and D. Strauss, "Markov graphs", *Journal of the American Statistical Association*, vol. 81, Sep. 1986, ISSN: 0162-1459.

[GL06] D. Gamerman and H. Lopes, *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*, eng. Jan. 2006, pp. xvii–xvii, ISBN: 9781584885870. [Online]. Available: `http://search.proquest.com/docview/1935799470/`.

[GKM09] S. M. Goodreau, J. A. Kitts, and M. Morris, "Birds of a feather, or friend of a friend?: Using exponential random graph models to investigate adolescent social networks", *Demography*, vol. 46, no. 1, pp. 103–125, 2009.

[Han03] M. S. Handcock, "Assessing degeneracy in statistical models of social networks", Tech. Rep., Dec. 2003, Working Paper no. 39, Center for Statistics and the Social Sciences, Univeristy of Washington, Seattle.

[HLMJ09] M. Hay, C. Li, G. Miklau, and D. Jensen, "Accurate estimation of the degree distribution of private networks", eng, IEEE Publishing, Dec. 2009, pp. 169–178, ISBN: 978-1-4244-5242-2.

[HH06] D. R. Hunter and M. S. Handcock, "Inference in curved exponential family models for networks", eng, *Journal of Computational and Graphical Statistics*, vol. 15, no. 3, pp. 565–583, Sep. 2006.

[Jon99] J. Jonasson, "The random triangle model", eng, *Journal of Applied Probability*, vol. 36, no. 3, pp. 852–867, Sep. 1999.

[KKS17] V. Karwa, P. N. Krivitsky, and A. B. Slavkovic, "Sharing social network data: Differentially private estimation of exponential family random graph models", *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 66, no. 3, pp. 481–500, Apr. 2017.

[KRSY14] V. Karwa, S. Raskhodnikova, A. Smith, and G. Yaroslavtsev, "Private analysis of graph structure", eng, *ACM Transactions on Database Systems (TODS)*, vol. 39, no. 3, pp. 1–33, Oct. 2014.

[KS16] V. Karwa and A. Slavkovic, "Inference using noisy degrees: Differentially private β-model and synthetic graphs", eng, *Annals of Statistics*, vol. 44, no. 1, Feb. 2016.

[KNRS13] S. P. Kasiviswanathan, K. Nissim, S. Raskhodnikova, and A. Smith, "Analyzing graphs with node differential privacy", in *Theory of Cryptography*, A. Sahai, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 457–476, ISBN: 978-3-642-36594-2.

[LM14] W. Lu and G. Miklau, "Exponential random graph estimation under differential privacy", eng, in *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*, ACM, Aug. 2014, pp. 921–930.

[LKR12] D. Lusher, J. Koskinen, and G. Robins, *Exponential Random Graph Models for Social Networks : Theory, Methods, and Applications*. Cambridge: Cambridge University Press, 2012, ISBN: 9780511894701.

[MT07] F. Mcsherry and K. Talwar, "Mechanism design via differential privacy", eng, IEEE, Oct. 2007, pp. 94–103.

[MW12] D. Mir and R. Wright, "A differentially private estimator for the stochastic kronecker graph model", eng, in *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, ser. EDBT-ICDT '12, ACM, Mar. 2012, pp. 167–176.

[MGM12] I. Murray, Z. Ghahramani, and D. MacKay, "Mcmc for doubly-intractable distributions", Jun. 2012.

[NRS07] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis", eng, in *Proceedings of the thirty-ninth annual ACM symposium on theory of computing*, ser. STOC '07, ACM, Jun. 2007, pp. 75–84, ISBN: 9781595936318.

[RAS10] E. Roland, B. Alla, and B. Svetlana, "Bayesian statistical modelling of human protein interaction network incorporating protein disorder information", eng, *BMC Bioinformatics*, vol. 11, no. 1, Jan. 2010.

[SDC+16] M. R. Sinke, R. M. Dijkhuizen, A. Caimo, C. J. Stam, and W. M. Otte, "Bayesian exponential random graph modeling of whole-brain structural networks across lifespan", eng, *NeuroImage*, vol. 135, pp. 79–91, Jul. 2016.

[SPRH06] T. Snijders, P. Pattison, G. Robins, and M. Handcock, "New specifications for exponential random graph models", eng, *Sociological Methodology*, vol. 36, pp. 99–153, Jan. 2006.

[UHH13] S. Uddin, J. Hamra, and L. Hossain, "Exploring communication networks to understand organizational crisis using exponential random graph models", eng, *Computational and Mathematical Organization Theory*, vol. 19, no. 1, pp. 25–41, Mar. 2013.

[War65] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias", *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, Mar. 1965.

[WP96] S. Wasserman and P. Pattison, “Logit models and logistic regressions for social networks”, eng, *Psychometrika*, vol. 61, no. 3, pp. 401–425, Sep. 1996.

Appendix A: MCMC Methods Used in Bayesian Inference over ERGMs

Simulating Networks from an ERGM

First, we describe a simple MCMC method for simulating networks from an ERGM given parameters of the model. This method is used both to generate synthetic graphs in our experiments and to draw samples needed for inference.

Algorithm 3 Metropolis-Hastings Sampler for ERGMs

Input: parameter vector θ, initial graph $x^{(0)}$, number of iterations T
Output: sequence of graphs $x^{(1)}, ..., x^{(T)}$ such that $x^{(T)} \sim p(X|\theta)$ as $T \to \infty$

For $t = 1, ..., T$:

1. Select nodes i and j at random
2. Propose graph x^* which is the same as $x^{(t-1)}$ except that we "toggle" the edge between i and j so $x^*_{ij} = 1 - x^{(t-1)}_{ij}$
3. Accept the proposed move with probability $\min\left\{1, \frac{p(x^*|\theta)}{p(x^{(t-1)}|\theta)}\right\}$. If the move is accepted set $x^{(t)} = x^*$. Otherwise, set $x^{(t)} = x^{(t-1)}$

The acceptance ratio (assuming all pairs of nodes are chosen with equal probability) is just $\exp\{\theta^T(u(x^*) - u(x^{(t-1)}))$. As the difference in sufficient statistics between two graphs differing in an edge (known as the "change statistic") is typically a simple function of the nodes participating in that edge, this ratio is easy to compute (for instance, for the edges sufficient statistic, it is always just 1 if adding an edge and -1 if removing).

If an ERGM specification puts most of its probability mass on relatively sparse graphs, the sampler that proposes all pairs of nodes with equal probability in step 1 will reject the addition of an edge in most steps, leading to slow convergence. Therefore, Tie-No-Tie (TNT) sampling is generally used in step 1, where we first select either the set of edges or the set of non-edges with equal probability and then pairs of nodes are selected uniformly at random from within the chosen set, biasing step 1 to consider removing edges more frequently than adding (and accounting for the non-uniform proposal distribution in the acceptance ratio). Therefore, throughout this thesis we use TNT sampling to efficiently draw samples from ERGMs.

Population MCMC Version of the Exchange Algorithm

The basic exchange algorithm for Bayesian inference over ERGMs can be easily modified to take advantage of population MCMC methods, which tend to converge faster, since using various chains reduces temporal dependency between time-steps in the Markov Chain. In particular, Caimo and Friel propose using parallel ADS, which maintains a collection of H chains that interact with one another.

Algorithm 4 Non-Private Bayesian Inference for ERGMs (Parallel ADS) [CF11]

Input: ERGM distribution $\pi(X|\theta)$, prior $p(\theta)$, observed graph x_{obs}, number of chains to use H, tuning parameter γ.
Output: sequence of draws $(\theta_1^{(r)}, ...\theta_1^{(T)}), ..., (\theta_H^{(r)}, ...\theta_H^{(T)})$ from posterior distributions $p(\theta_h|x_{obs})$.

For $t = 1, ..., T$:

For each chain $h = 1, ..., H$:

1. Select at random two different chains h_1 and h_2 from $\{1, ..., H\}\backslash h$
2. Propose $\theta_h^* = \theta_h^{(t-1)} + \gamma\left(\theta_{h_1}^{(t-1)} - \theta_{h_2}^{(t-1)}\right) + \epsilon$
   where ϵ is random noise drawn from a symmetric distribution, such as a Normal.
3. Sample graph $x_h^* \sim \pi(\cdot|\theta_h^*)$
4. Accept the proposed move with probability $\min\{1, \alpha\}$. If the move is accepted, set $\theta_h^{(t)} = \theta_h^*$. Otherwise, set $\theta_h^{(t)} = \theta_h^{(t-1)}$

where

$$\alpha = \frac{p(\theta_h^*)}{p(\theta_h^{(t-1)})} \exp\left\{\left(\theta_h^* - \theta_h^{(t-1)}\right)^T (u(x_{obs}) - u(x_h^*))\right\}$$

The MH acceptance ratio reamins the same as in the single-site update, because the proposal distribution is still symmetric – making the reverse jump from θ_h^* to $\theta_h^{(t-1)}$ simply requires reversing ϵ and the order of h_1 and h_2. The tuning parameter γ controls the amount of interaction between chains and is generally taken to be in the range 0.5 and 1 (in this case we take $\gamma = 0.5$ throughout.) Additionally, the number of chains to use can be tuned in inference, but we choose to use 3 chains throughout as this seems to lead to convergence.

Appendix B: Smooth Projections to $\mathcal{H}_k$

B.1 Edge-Adjacency Model

Blocki et. al give an efficient projection to $\mathcal{H}_k$ in the edge-adjacency model with $GS_\mu = 3$ [BBDS13]:

Algorithm 5 3-smooth Projection to $\mathcal{H}_k$ for Edge-Adjacency Model

Input: graph G, cutoff k
Output: graph $\mu(G)$ with max degree k

1. Fix a canonical ordering over all possible edges in a graph on n vertices. Let $e_1^v ... e_t^v$ denote the edges incident to vertex v in this canonical ordering.
2. Delete edge $e = (u, v)$ if and only if:
   (i) $e = e_j^v$ for $j > k$, or
   (ii) $e = e_j^u$ for $j > k$

Intuitively, we keep only the first k edges in the canonical ordering for any node with degree above k. It is clear, then, that this algorithm results in a graph of max degree k and that any graph where all nodes have degree less than k are unchanged. The global sensitivity follows fairly straightforwardly. Consider two graphs G_1 and G_2 that are neighbors differing on a single edge $e = (x, y)$ where wlog G_1 contains e. Then, for every vertex that is not x or y, exactly the same set of edges is deleted, since e does not appear in any other nodes' canonical ordering. If e is deleted, then $\mu(G_1) = \mu(G_2)$. However, if e is not deleted than there may be at most one edge incident to x and one edge incident to y that were deleted from $\mu(G_1)$ but not $\mu(G_2)$, so the neighboring graphs differ in 3 edges. In practice, since this algorithm deletes edges from high degree nodes, it may not bias results too extensively to aggressively estimate k for a graph, as this will only mark edges for deletion on a few nodes that are above the cutoff. However, choosing a cutoff that is too low may remove many edges from many high degree nodes, which will bias not only the number of edges, but also many other sub-graph counts like triangles k-stars, which we explore in our experimental results.

B.2 Node-Adjacency Model

Naive Truncation

The naive truncation projection $\mu_{trunc} : \mathcal{G}_n \to \mathcal{H}_k$ proposed by Kasiviswanathan et. al. simply removes all nodes from the graph with degree above the cutoff k [KNRS13]. It is clear, then, that μ_{trunc} maps any graph in $\mathcal{H}_k$ to itself and that its image is $\mathcal{H}_k$. Moreover, μ_{trunc} is quite efficient, requiring $O(n + \binom{n}{2})$ time. It is also fairly simple to characterize the smooth sensitivity of μ_{trunc}. First, note that the local sensitivity of μ_{trunc} on graph G is $1 + D_k(G) + D_{k+1}(G)$ where D_i is the number of nodes of degree i in graph G since rewiring one node in the graph may affect whether all nodes of degree k or $k+1$ are truncated by μ_{trunc}. We an characterize the smooth sensitivity as follows:

Proposition B.1 (Smooth Sensitivity of μ_{trunc} [KNRS13])**.** Given graph G and hypothesis $\mathcal{H}_k$, let $N_t(G)$ denote the number of nodes with degrees in the range $[k-t, k+t+1]$ and let $C_t = 1 + t + N_t(G)$. Then:

1. The local sensitivity of μ_{trunc} is $C_0(G)$.
2. The local sensitivity at distance t of μ_{trunc} is $C_{t-1}(G)$.
3. The β-smooth sensitivity of μ_{trunc} is $\max_{t \geq 0} e^{-\beta t} C_t(G)$.
4. If $N_{\ln n/\beta}(G) \leq \ell$, so there are at most ℓ nodes in G with degree in range $k \pm (\ln n/\beta)$, then
$$S^*_{\mu,\beta}(G) \leq 1 + \ell + \frac{1}{\beta}$$

Thus, we can compute β-smooth sensitivity efficiently using either part 3 or 4 of the above proposition. Notice that even if a graph is in $\mathcal{H}_k$, it may have high smooth sensitivity if it has many nodes with degree close to the cutoff k. However, part 4 gives a guideline for choosing a conservative cutoff k. In particular, by choosing k to be $\ln n/\beta$ above what is thought to be the max degree of the graph, then the smooth sensitivity would simply be 1. This is not an unreasonable quantity to add to the cutoff, if the cutoff is itself $O(\log n)$, which is often the case. Further, degree distributions are often thought to fall exponentially, so that it is unlikely that there are very many nodes with degree near the cutoff, especially if a conservative cutoff is chosen, suggesting that ℓ might be quite low, even for cutoffs close to the believed cutoff k.

LP-Based Projection

Blocki et. al. propose a projection using linear programming. Their method satisfies a relaxed definition of a projection, where $\mu_{LP} : \mathcal{G}_n \to \mathcal{H}_{2k}$ and $\forall G \in \mathcal{H}_k$, $\mu(G) = G$, (but graphs in $\mathcal{H}_k$ are not necessarily mapped to themselves). Because the image is $\mathcal{H}_{2k}$, their method requires calibrating the restricted sensitivity to $\mathcal{H}_{2k}$. However, in contrast to

naive truncation, their approach guarantees that graphs in $\mathcal{H}_k$ always have low smooth sensitivity, because their algorithm outputs an estimator of the distance between a graph and its projection, used to compute a β-smooth upper bound, where the distance estimator is always 0 for graphs in $\mathcal{H}_k$.

The algorithm is also less efficient than naive truncation as it requires solving a linear program with $n + \binom{n}{2}$ decision variables: a variable x_u per node u representing whether node u should be removed from the projected graph or not and a variable $w_{u,v}$ per edge (u, v) representing whether the edge from u to v remains in the projected graph:

Algorithm 6 Projection and 4-Smooth Distance Estimator to $\mathcal{H}_{2k}$ for Node-Adjacency Model [BBDS13]

Input: graph G, cutoff k
Output: graph $\mu_{LP}(G)$ with max degree $2k$, 4-smooth estimate of distance from graph to its projection $\hat{d}(G)$

1. Solve the following LP to get fractional solution $(\bar{x}^*, \bar{w}^*)$. Let there be n decision variables x_u, one for each vertex, and $\binom{n}{2}$ decision variables $w_{u,v}$ one for each potential edge. Additionally, let $a_{uv} = 1$ if the edge $\{u, v\}$ is in G and 0 otherwise. Then, solve the following LP:

$$\min \sum_{v \in V} x_v \quad s.t.$$

$$(1) \quad \forall v, x_v \geq 0$$
$$(2) \quad \forall u, v, w_{u,v} \geq 0$$
$$(3) \quad \forall u, v, a_{uv} \geq w_{uv} \geq a_{uv} - x_u - x_v$$
$$(4) \quad \forall u, \sum_{v \neq u} w_{u,v} \leq k$$

2. Let $\mu_{LP}(G)$ be the graph resulting from removing every edge in G for which either endpoint has weight greater than $\frac{1}{4}$, so either $x_u^* > \frac{1}{4}$ or $x_v^* > \frac{1}{4}$ for edge (u, v).

3. Define distance estimator to be $\hat{d}(G) = 4 \sum_u x_u^*$.

It is clear that if $G \in \mathcal{H}_k$, then the algorithm will return a distance estimator of 0 and $\mu_{LP}(G) = G$, since we can take all x_v to be equal to 0, $w_{uv} = a_{uv}$ so that conditions 1 to 3 of the LP are met and condition 4 is met because all vertices have degree less than k. Using the distance estimator gives a β-smooth upper bound on the local sensitivity of μ_{LP}:

Proposition B.2 (Smooth Sensitivity of μ_{LP} [BBDS13])**.** The smooth sensitivity of μ_{LP} can be bounded by

$$S_{\mu,\beta}(G) \leq \exp\left\{\tfrac{\beta}{4}\hat{d}(G)\right\} \cdot g\left(\tfrac{\beta}{4}\right)$$

where

$$g(x) = \begin{cases} \frac{2}{x} e^{-1+\frac{5}{2}x}, & 0 \leq x \leq \frac{2}{5} \\ 5, & x > \frac{2}{5} \end{cases}$$

so $S_{\mu,\beta}(G) RS_f(\mathcal{H}_{2k})$ is a β-smooth upper bound on the local sensitivity of $f \circ \mu_{LP}$ on graph G.

Comparing the two proposed methods, it is preferable to use naive truncation in cases where we believe $k \geq \ln n/\beta$, because then setting the cutoff to be $\hat{k} = k + \ln n/\beta$, we expect smooth sensitivity of μ_{trunc} to be below $1 + \frac{1}{\beta}$ and the restricted sensitivity will be lower than $RS_f(\mathcal{H}_{2k})$. In general, since we believe the graphs under consideration to have very few high degree nodes close to the cutoff, we expect naive truncation to perform quite well, since the smooth sensitivity should be relatively low for the graphs considered, while considering restricted sensitivity on $\mathcal{H}_{2k}$ may introduce more noise.