# ERGMS FOR PARTIALLY RANKED DATA

Patrick Liang

Supervisor: ~~Professor~~ Dr. Pavel N. Krivitsky

School of Mathematics and Statistics
UNSW Sydney

November 2025

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE DEGREE OF
BACHELOR OF SCIENCE WITH HONOURS

# Plagiarism statement

I declare that this thesis is my own work, except where acknowledged, and has not been submitted for academic credit elsewhere.

I acknowledge that the assessor of this thesis may, for the purpose of assessing it:
- Reproduce it and provide a copy to another member of the University; and/or,
- Communicate a copy of it to a plagiarism checking service (which may then retain a copy of it on its database for the purpose of future plagiarism checking).

I certify that I have read and understood the University Rules in respect of Student Academic Misconduct, and am aware of any potential plagiarism penalties which may apply.

By signing this declaration I am agreeing to the statements and conditions above.

Signed: _____     Date: _____

# Acknowledgements

By far the greatest thanks must go to my supervisor for the guidance, care and support they provided.

Thanks must also go to Emily, Michelle, John and Alex who helped by proof-reading the document in the final stages of preparation.

Although I have not lived with them for a number of years, my family also deserve many thanks for their encouragement.

Thanks go to Robert Taggart for allowing his thesis style to be shamelessly copied.

Patrick Liang, 2 November 2015.

# Abstract

Exponential-family random graph models (ERGMs) are widely used for modeling social networks, but current implementations assume rankings are fully observed, limiting their use with partially ranked data. This project presents a modified `ergm.rank` implementation, capable of fitting maximum likelihood estimates (MLEs) to partially observed rankings, to improve computational efficiency while accurately estimating key network statistics.

The author applies this methodology to publicly available datasets to assess its reliability and compare the results to known historical outcomes. This work shows that ERGMs can be effectively applied to partially observed ranked networks, offering a practical tool for network analysis and a foundation for further methodological research.

# Contents

Suggested reorg for Chapter 1-3 (all in the same chapter):
1) Ordinal networks (in general).
2) Non-ERGM methods.
3) ERGMs; briefly talk about binary, then Krivitsky and Butts 2017 overview.
Make sure to explain network modelling in general.
4) Problem: most ordinal networks are not completely observed, e.g., only top X,
as well as scalability.
5) Missing data methods for ERGMs.
6) What this thesis will do.

# CHAPTER 1

# Exponential-Family Random Graph Models (ERGMs)

Exponential-family random graph models (ERGMs) have become a central framework for modelling the structure of social networks. ERGMs are useful in that they can be specifically designed to capture the effects that variables have on the likelihood of certain patterns within a network. By representing the dependencies between individuals in a network, ERGMs enable the statistical analysis of network processes under a probabilistic framework.

This section introduces ERGMs, and reviews their applicability to rank-order relational data, as well as their uses for inference when modelling random graphs.

## 1.1 ERGMs in Networks

Networks are a representation of relational data, and show the interactions and connections between individuals. There are a variety of different networks which can be studied, from networks of social contacts, financial networks and collaboration networks, among others. As such, many researchers are interested in examining the factors which affect the structure of these networks, such as why some networks are densely interconnected and why others may be sparse [van der Pol, 2018, p. 845].

ERGMs are useful in the study of social networks, as for unknown links between individuals within a network, it allows the probability of a link between two nodes to be dependent on other links inside the network, amongst other variables, whilst also accounting for direct interactions and weighted interactions between nodes.

I think you need to pass "round" to natbib to use ().

Within the context of this project, ERGMs can be used on incomplete networks with missing or censored information, in order to model the possible variations of the missing links in the network, based on the known information and sample data. ERGMs can also be used for hypothesis testing, which opens up the possibility of analysing relational data as well as normal data [van der Pol, 2018, p. 846].

Furthermore, van der Pol [2018] provides sample code to operationalise and interpret results and analyse goodness of fit, which can be used to trial and error ERGMs in RStudio to find which model maximises the likelihood. This can be helpful to ensure that the ERGM fitted is effective for in-depth analysis and to check for statistical significance.

## 1.2 Degeneracy and Projectivity

Using trial and error to test ERGMs is important to ensure that they are non-degenerate. Degeneracy generally refers to when ERGMs do not behave as expected during sampling and estimation [Mukherjee, 2020]. Near-degenerate ERGMs tend to concentrate probability mass on a small set of extreme network configurations, such as almost empty or nearly complete networks. Concerns about the effects near-degeneracy and non-projectivity have on inference have been investigated in other works [Schweinberger et al., 2020b, p. 632].

Consider a subgraph $\mathbf{y}_{\mathcal{N}'}$ of a population graph $\mathbf{y}_{\mathcal{N}}$, induced by a subset of nodes $\mathcal{N}' \subset \mathcal{N}$. That is, $\mathbf{y}_{\mathcal{N}'}$ is the subgraph with the set of nodes $\mathcal{N}'$ and all the edges among nodes in $\mathcal{N}'$ contained in the population graph $\mathbf{y}_{\mathcal{N}}$. An ERGM is considered projective if:

$$\eta(\theta, \mathcal{N}') = \boldsymbol{\theta} \ \text{ for all } \ \boldsymbol{\theta} \in \boldsymbol{\Theta} \ \text{ and all } \ \mathcal{N}' \subseteq \mathcal{N},$$

and

$$\mathbb{P}_{\mathcal{N}', \boldsymbol{\theta}}(\mathbf{Y}_{\mathcal{N}'} = \mathbf{y}_{\mathcal{N}'}) = \mathbb{P}_{\mathcal{N}, \boldsymbol{\theta}}(\mathbf{Y}_{\mathcal{N}'} = \mathbf{y}_{\mathcal{N}'}, \mathbf{Y}_{\mathcal{N} \backslash \mathcal{N}'} \in \mathcal{Y}_{\mathcal{N} \backslash \mathcal{N}'}),$$

where $\mathbf{Y}_{\mathcal{N} \backslash \mathcal{N}'} \in \mathcal{Y}_{\mathcal{N} \backslash \mathcal{N}'}$ denotes the subset of possible edges of the population graph not contained in $\mathbf{y}_{\mathcal{N}}$. That is, the same parameters govern the marginal distributions of all subgraphs of a larger network. Schweinberger et al. [2020b] concludes that a lack of projectivity can affect non-likelihood-based inference, but is not necessary for likelihood-based-inference. As the analyses in this thesis rely primarily on likelihood-based estimation, projectivity is not a concern in this context.

## 1.3 Population Inference Scenarios

For the data generating process, Schweinberger et al. [2020b] defines three broad goals of statistical inference: finite, super and infinite population inference. Finite population inference is concerned with a finite population of nodes and a fixed population graph, super population inference is similar but assumes the population graph was generated by a population property model, and infinite population inference having an infinite population of nodes.

They each have different goals: finite aims to estimate functions of the population graphs, such as the total number of edges in the population graph, super aims to estimate the population probability model which generates the possible population graphs, and the infinite aims to estimate the population probability model based on a subgraph induced by a subset of nodes. In all three of these scenarios, the consistency and asymptotic normality results for the likelihood-estimators confirmed that statistical inference for ERGMs is possible, but requires well-posed questions in order for the ERGMs to be well-behaved.

In this project, we adopt the superpopulation inference framework, treating a given partially ranked network as a single draw from an underlying probabilistic model. Using ERGMs to perform maximum likelihood estimation (MLE) on the partial rankings, we generate parameter estimates of the underlying model.

# Ranked Data and Statistical Methods

Rank-order relational data, or ordinal relational data, is collected by having each actor in a network rank other actors according to some criterion, as defined by Krivitsky and Butts [2017]. Rank-order data is notated by a set of $n$ actors, $N$ of which we index as $N = \{A, B, C, \ldots\}$. Then, each actor $i$ denotes an ordinal relation $\succ_i$ over a set $N$. This relation can represent a multitude of things, such as "preferred to" or "interacted more with", and so on.

## 2.1 Modelling Framework

For two actors (egos) $i$ and $j$, when asking them to rank some other actors (alters) $k$ and $l$, it's not meaningful to compare how either $i$ or $j$ ranks $k$, but it can be to ask whether $(k \succ_i l) = (k \succ_j l)$. That is, ordinal relational data can be used to determine whether $i$ and $j's$ rankings of $k$ and $l$ are concordant, and can tell us something about the network. For example, if both $i$ and $j$ prefer $k$ over $l$, it could suggest that $l$ is less liked among the individuals in the network. The modelling framework utilised from Krivitsky and Butts [2017] is founded on these distinctions.

Do NOT use [h] for figures. Let them float, and use labels and cross-references to refer to them by number.
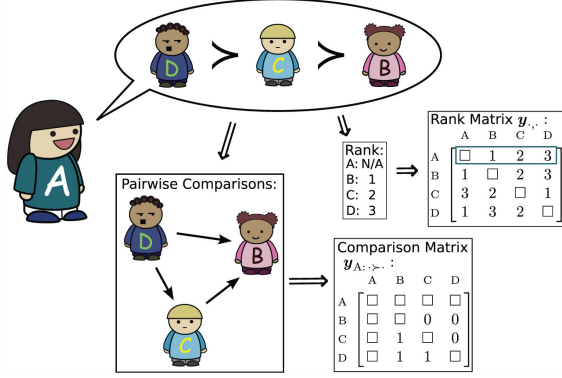
**Figure 2.1:** A simple example of a ranking structure for ordinal relational data from Krivitsky and Butts [2017]. Here, ego **A**'s rankings of alters **B, C** and **D** can be encoded as either a rank matrix by assigning arbitrary ranks of 1, 2 and 3 to **B, C** and **D** respectively, or instead as pairwise comparisons implied by **A**. Note that □ denotes matrix entries that are unobservable and/or meaningless, such as people ranking themselves.

Whilst ERGMs have been used to model internal network structures in a variety of contexts, they tend to pose some issues. When applied to dichotomised rank-order data, that is, the ordinal or ranked preferences have been converted into binary (yes/no, 0/1) ties, the model would inherit the difficulties that come with dichotomising, most notably losing information and introducing biases [Goodreau et al., 2009].

<span style="color:red">Are these citations?</span>

Building on the work of Robins et al. (1999), who was the first to model categorically network data using ERGMs, and Snijders (1996) who applied the stochastic actor-oriented framework to this data, Krivitsky [2012] formulated a generalised framework for ERGMs on networks whose ties have values, as well as introducing Markov-chain Monte Carlo (MCMC) methods for simulation and maximum likelihood inference for this general framework. The paper also formulated the maximum likelihood inference for this general framework, which can be used to determine whether a fitted ERGM maximises the likelihood.

Krivitsky and Butts [2017] formally defined the sample space complete rankings of every actor in a network by every other actor as:

$$\mathscr{Y} = \left\{ \boldsymbol{y}\prime \in \mathbb{S}^{\mathbb{Y}} : \forall_{i \in \mathbb{N}} \ \forall_{r \in \mathbb{S}} \exists!_{j \in N \setminus \{i\}} \boldsymbol{y}\prime_{i,j} = r \right\}.$$

which is a directed network with no self-loops, whose set of observed relations $\mathbb{Y} = N^{2 \neq}$ maps to dyad values $\mathbb{S} = \{1, ..n-1\}$. This also leads to the constraint that an ego must assign a unique rank to each possible alter.

The probability associated with each network $\boldsymbol{y} \in Y$ in the sample space is given by

$$\Pr_{\boldsymbol{\theta}; g, \boldsymbol{x}}(\mathbf{Y} = \boldsymbol{y}) = \frac{\exp\left\{\boldsymbol{\theta} \cdot g(\boldsymbol{y}; \boldsymbol{x})\right\}}{\kappa_{g, \boldsymbol{x}}(\boldsymbol{\theta})}, \quad \boldsymbol{y} \in \mathscr{Y}$$

with the normalising factor

$$\kappa_{g,\boldsymbol{x}}(\boldsymbol{\theta}) = \sum_{\boldsymbol{y\prime} \in \mathscr{Y}} \exp\left\{\boldsymbol{\theta} \cdot g(\boldsymbol{y\prime}; \boldsymbol{x})\right\},$$

where the sufficient statistic $\boldsymbol{g}(\boldsymbol{y}; \boldsymbol{x})$ is a function of network $\boldsymbol{y} \in Y$ that may be also dependent on fixed and known covariates $\boldsymbol{x} \in \mathbb{X}$ [Krivitsky and Butts, 2017, Section 2.3]. The probability is used to define the probability distribution of networks under a given ERGM, and serve as the basis for model estimation (ERGM fitting) and checking goodness of fit.

## 2.2 Promotion Statistics

Another important concept to discuss is the use of "promotion" statistics, which in this context means to change statistics or scores within the data structure. For complete ordering data, an ego changing the ranking of one alter will change the ranking of at least one other alter, and an atomic change swaps the rankings of two adjacently ranked alters.

An atomic change is represented as the effect of having ego $i$ "promote" a promotable alter $j \in \{k \in N : k \neq i \ \wedge \ y_{i,k} < n - 1\}$, which represents swapping $j$'s rank with that of the alter previously ranked immediately above $j$. Then, the promotion statistic is defined by Krivitsky and Butts [2017] as

$$\Delta_{i,j}^{\nearrow} g(\boldsymbol{y}) \equiv g(\boldsymbol{y}^{i:j \leftrightarrows j^+}) - g(\boldsymbol{y}),$$

where $(\boldsymbol{y}^{i:j \leftrightarrows j^+})$ represents the network $\boldsymbol{y}$ with $i's$ ranking of $j$ and $j^+$, the actor previously ranked immediately above $j$ by $i$ [Krivitsky and Butts, 2017, Section 3.1]. That is, the promotion statistic is defined as the change in $g$ resulting from $i$ "promoting" $j$ by exactly one rank.

## 2.3 Rank-order Network Statistics

We also introduce statistical measures used to analyze ranked network data, as their associated parameters provide insight into the underlying network structure. The primary measures considered are nonconformity, deference, and inconsistency.

### 2.3.1 Nonconformity

Nonconformity is a measure of the departure from shared or expected ranks. An instance of nonconformity, for example, would be if most egos rank $\mathbf{A}$ over $\mathbf{B}$, but one ego ranks $\mathbf{B}$ over $\mathbf{A}$. A natural statistic to summarise the degree of ratings nonconformity is defined by Krivitsky and Butts [2017] as

Number these for referencing later.
$$g_{GNC}(\boldsymbol{y}) = \sum_{(i,j,k,l) \in N^{4\neq}} y_{l:j \succ k}(1 - y_{i:j \succ k}).$$

The promotion statistic for nonconformity is also given by

$$\Delta_{i,j}^{\nearrow} g_{GNC}(\boldsymbol{y}) = 2 \sum_{l \in N \setminus \{i,j,j^+\}} (y_{l:j^+ \succ j} - y_{l:j \succ j^+}),$$

You can give computational complexity for all of these in big-O notation.

which is derived by observing that when $i$ promotes $j$ over $j^+$, the statistic is incremented by 2 every other ego $l$ who has $j^+$ ranked over $j$, and decremented by 2 for every $l$ who has $j$ ranked over $j^+$.

<span style="color:red">\citet{} here. --></span> [Krivitsky and Butts, 2017] also defines the statistics for local nonconformity $g_{LNC}$, which unlike global nonconformity $g_{GNC}$, factors in that $i$ may be more likely to conform to those whom they rank highly over those whom they rank lower. The statistic for $g_{LNC}$ is given by

$$g_{LNC}(\boldsymbol{y}) = \sum_{(i,j,k,l) \in N^{4 \neq}} y_{i:l \succ j} y_{i:l \succ k} y_{l:j \succ k} (1 - y_{i:j \succ k}),$$

with atomic effects for this statistic given by

$$\Delta_{i,j}^{\nearrow} g_{LNC}(\boldsymbol{y}) = \sum_{k \in N \setminus \{i,j,j^+\}} \big( y_{i:k \succ j^+} y_{k:j^+ \succ j} - y_{i:k \succ j^+} y_{k:j \succ j^+}$$
$$+ \, y_{k:i \succ j^+} y_{k:j^+ \succ j} - y_{k:i \succ j} y_{k:j \succ j^+}$$
$$+ \, y_{j:k \succ j^+} y_{i:j^+ \succ k} - y_{j^+:k \succ j} y_{i:j \succ k} \big).$$

### 2.3.2 Deference

Deference aversion is a measure of the tendency for an ego to not rank others highly if they do not reciprocate deference, that is, they rank them below others. An example of deference aversion would be if an ego $i$ does not rank $l$ above $j$, with $l$ ranking $i$ below $j$. Krivitsky and Butts [2017] captures the notion of this statistic with the following:

$$g_D(\boldsymbol{y}) = \sum_{(i,j,l) \in N^{3 \neq}} y_{l:j \succ i} y_{i:l \succ j},$$

which is expected to be lower when deference aversion is present. The promotion statistic is defined as:

$$\Delta_{i,j}^{\nearrow} g_D(\boldsymbol{y}) = 2(y_{j^+:i \succ j} + y_{j:j^+ \succ i} - 1),$$

and is similarly to nonconformity reliant on adjacent actors. This makes calculating the promotion statistics for either nonconformity or deference more computationally intensive, as it needs to know which indicators change in between the two actors being swapped. Ideally for the promotion statistics for global or local nonconformity and deference, the promotion statistic formula is changed so that it can minimise the number indicators that change. This would allow us to easily recalculate whenever an atomic change occurs, which would reduce computational costs for the network's data structure.

### 2.3.3 Inconsistency

Inconsistency in a given network is a measure of the tendency of each ego to rank alters discordantly. That is, whether an ego follows a trend to their alter rankings, as we expect that an ego will tend to exhibit consistency in alter ratings [Krivitsky and Butts, 2017]. For example, if the general trend in the network is to rate the three alters $A, B$ and $C$ as $A \succ B \succ C$, if an ego $X$ ranks them as $A \succ C \succ B$,

ego $X$ exhibits inconsistency. For two ranking structures $\boldsymbol{y}$ and $\boldsymbol{y}\prime$ on vertex sets $N$ and $N'$ such that $N_s$ represents the actors involved in both networks, the statistic measuring inconsistency is given by

$$g_{\mathrm{I}}(\boldsymbol{y}; \boldsymbol{y}\prime) = \sum_{(i,j,k) \in N_s^{3\neq}} [\boldsymbol{y}_{i:j\succ k}(1 - \boldsymbol{y}\prime_{i:j\succ k}) + (1 - \boldsymbol{y}_{i:j\succ k})\boldsymbol{y}\prime_{i:j\succ k}]$$

with promotion statistics

$$\Delta_{i,j}^{\nearrow} g_{\mathrm{I}}(\boldsymbol{y}; \boldsymbol{y}\prime) = 2(\boldsymbol{y}\prime_{i:j^+\succ j} - \boldsymbol{y}\prime_{i:j\succ j^+})$$

Inconsistency tends to be sparse and difficult to capture, and is therefore less useful than deference and nonconformity for understanding a network's underlying structure. For this reason, the analyses in this project will focus on deference and nonconformity as the parameters.

## 2.4 The "G-Method"

The "G-Method" is a generalized approach to assigning ranks in datasets where observations come from multiple groups or populations ("c" samples), include ordinal, non-numeric, or discrete variables and can also account for tied values.

The "G-Method" is a series of equations which form a mathematically formal rank assignment system, which systematically assigns ranks to observations in each of the "c" samples after all the sample observations are pooled and assigned ranks [Oyeka et al., 2014, Section 2]. These "c" samples are defined as $x_{il}$, which are the $i^{th}$ observation or score in a random sample of size $n_l$ drawn from a population $x_l$, for $i = 1, 2, .., n$ and $l = 1, 2, \ldots c$.

This method is helpful dealing with rank-order relational data which may contain non-numeric ordinal entries, as the "G-Method" allows us to convert these into ranks for data analysis. Since ERGMs require valid rankings, the "G-method" ensures that the input is well-formed before modelling for statistical analysis. For example, in Section 3, Oyeka et al. [2014] demonstrates the use of the "G-method" to convert letter grades into ranks for use in data analysis.

## 2.5 Analysis Methods for Ranked Data

Ranked response data, compared to other forms of statistical data, it is unique in that it is deterministic in nature, that is, the rank of the other items in the sample determine the rank of a certain item.

Finch [2022] provides applicable methodology to the use of libraries in R, as well as R code for the purposes of analysing ranking data, mainly through the statistical modelling of relative ranks, and inference regarding the difference in ranking patterns which may be noticeable between groups or samples.

An example of the latter is the analysis of fraternity data from Newcomb [1961b],

which modelled the difference in ranking patterns in a group of fraternity students, as it changed over the course of 15 weeks of collected data.

# Sampling, Privacy, and Representativeness

When dealing social networks, certain factors can affect the inference and consequently the conclusions drawn from a network. This section focuses on discussing how different methods of sampling and missing data can affect inference.

## 3.1 Likelihood Inference

Survey methods for gathering social network data tend to result in censored, missing or otherwise incomplete data. For example, the fixed rank nomination (FRN) scheme asks participants to nominate and rank a small number of contacts or friends, leaving other relations and their existence uncertain [Hoff et al., 2013]. Problematically, statistical models tend to be formulated in terms of complete observed binary networks, meaning statistical analysis of FRN data with these kinds of models ignores the censored and ranked nature of the data, which can result in misleading statistical inference.

Interestingly, when comparing Bayesian parameter estimates obtained from a likelihood for complete binary networks compared to likelihoods derived from the FRN scheme, they can draw different conclusions from the same data. Binary likelihood can provide misleading inference, especially for model parameters that relate network ties to characteristics of individuals and pairs of individuals [Hoff et al., 2013, p. 255].

It is important to not treat missing connections as non-existent, but instead censored, when simulating on partially ranked or incomplete data. We aim for an approach similar to the likelihoods derived from the FRN scheme, as we could treat partially ranked ordinal data as a form of data obtained from a FRN scheme, so that we do not make any assumptions about any of the missing data, and treat non-nominations instead as ranked and censored data instead, rather than absent. For example, we could assume a maximum nomination limit of $m$, and ties not nominated are treated as censored when the limit was reached, and as absent otherwise.

Another important concept for modelling network data using ERGMs is the general convex hull problem. The convex hull of a finite set of points $S = \{p_1, p_2, \ldots, p_n\}$ in $\mathbb{R}^d$, denoted $\text{conv}(S)$, is the smallest convex set that contains all points in $S$. The general convex hull problem is useful when determining whether the estimator of the log-likelihood-ratio has a maximiser [Krivitsky et al., 2023b].

Krivitsky [2012] also discusses implementations in the widely used `ergm` package

for network modeling, which improve the algorithm for the general convex hull problem algorithm. The general convex hull problem is helpful for calculating the log-likelihood ratio to assess the relative goodness of fit between two ERGMs applied to partially ranked data.

It is important to consider the approach we should take when modelling social networks from sampled data. Most notably, inference for social networks assumes the presence or absence of all possible links is observed and the information is reliable with no measurement error. Naturally, this is not true in practice when it comes to network data collected through sampled surveys, and even if a census of a population is attempted, individuals and their links do not appear in the data.

Handcock and Gile [2010] suggests that likelihood-based inference from an adaptive network sample (a selection of additional nodes from a partially sampled network) can be conducted using a complete network model. However, it should be based on the available observed data, including the sampling design information.

That is, for a partially observed network $Y$, we can consider using the face-value likelihood based solely on $Y_{\text{obs}}$, which ignores additional information about $\eta$ available in $D$, where $\eta$ is the natural parameter and $D$ is the $n \times n$ random binary matrix indicating if the corresponding element of $Y$ was sampled or not [Handcock and Gile, 2010, Section 3.2]. Then, the inference for $\eta$ and $\psi$ should be based on all the available observed data only, and the likelihood is any function of $\eta$ and $\psi$ proportional to $P(D, Y_{\text{obs}}|\eta, \psi)$, where $\psi$ is the probability an individual gets selected at random from sampling.

Such an approach gives some insight on a general likelihood framework for network inference for when the full network is not observed, and can be applied to partial rankings, as well as unobserved or unranked connections. Since partially ranked data isn't random, it is crucial that we consider the sampling mechanism, so that we can base our likelihood inference on it.

## 3.2 Protecting the Privacy of Sampled Individuals

When analysing synthetic networks, networks that mimic the statistical properties of a real social network, it is necessary to protect the privacy of individual relationships captured by the social network. A proposed approach to satisfy this would be a randomized response scheme for perturbing the edges and non-edges of the network, to generate a collection of synthetic edges whilst satisfying DP to control the risk [Karwa et al., 2017, Section 3.1]. The Differential Privacy (DP) framework is designed to capture the worst-case risk of releasing sensitive data [Dwork et al., 2006].

Edge differential privacy (EDP) is a measure of the worst-case disclosure risk of identifying any relationship, which is represented by edges, between entities, represented by nodes [Karwa et al., 2017, Section 3.2]. To generate synthetic networks under $\epsilon$-edge differential privacy, Karwa et al. [2017] proposes a simple randomised

I'd reframe this as motivating partially observed inference as allowing more privacy. E.g., we don't release full rankings.

response mechanism. For any two neighbouring networks $x$ and $x'$, the EDP bounds the worst-case ratio of the likelihoods of $Y$ conditionally on $x$ and $x'$. That is, $P_\gamma(Y = y \mid X = x)$ is $\epsilon$-edge differentially private if and only if

$$\max_y \max_{x,x':\Delta(x,x')=1} \log \left\{ \frac{P_\gamma(Y = y \mid X = x)}{P_\gamma(Y = y \mid X = x')} \right\} \le \epsilon,$$

where $P_\gamma(Y = y \mid X = x)$ denotes a family of conditional probability distributions, $x$ is the network that requires privacy protection, $Y$ is the random synthetic network obtained by sampling, and $\gamma$ is a vector parameter of the privacy mechanism controlling the generation of $Y$ from $x$. To find the maximum likelihood estimate (MLE) of $\boldsymbol{\theta}$, we apply the Monte Carlo maximum likelihood estimator [Karwa et al., 2017, Section 5]. The application and usefulness of these techniques are demonstrated in a case study using a version of the Enron e-mail corpus data set [Krivitsky et al., 2023a].

These privacy-preserving methods are essential when dealing with sensitive social network data. However, this project focuses on methodological evaluation using publicly available datasets, and therefore privacy concerns are not a limiting factor.

## 3.3 Generalisability of Network Sampling

It would be helpful for our project to have a way to test whether a model is generalisable, that is, whether it can be applied to different datasets to simulate incomplete data. Generalisability is important as an indicator for whether a model for simulating incomplete data is overfitted.

Krivitsky et al. [2023a] focuses on a case study between two samples of small networks of within-household contacts in Belgium, which were collected using two different but complementary sampling designs. The first sample was smaller, but with all contacts in each household observed, and the other sample is larger and more representative but recording contacts of only one person per household.

The paper aims to combine their strengths to learn the social forces that shape household contact formation. By doing so, it enables simulation for prediction of disease spread, while generalising to the population of households in the region. To accomplish this, they describe a flexible framework for specifying multi-network models in the ERGM class, and identify the requirements for inference and prediction under this framework to be consistent, identifiable and generalisable, for both complete or incomplete data. For each fitted model, they simulate multiple networks using estimated parameters and compare simulated network statistics to the observed ones. We are most interested in the framework's goal of evaluating model stability across structurally similar, but contextually different networks.

Krivitsky et al. [2023a] aims to find a model which is generalisable, that is, a model can be fitted to another dataset and can reproduce its structure, it is more generalisable. If a model fitted to dataset A poorly reproduces the data structure

of dataset B, it signals that the model has limited generalisability. Their framework, whose implementation is published in the R package `ergm.multi`, is useful to determine whether a given ERGM can be applied to multiple datasets to simulate partial data.

# CHAPTER 4

# Optimising the Estimation of Ranked ERGMs

Our main goal is, given an observed ranking, we want to be able to estimate the parameters of the underlying model which generated the network. The current version of `ergm.rank` supports Metropolis–Hastings proposals only for ERGMs with complete orderings. Our goal is to extend this functionality to accommodate partial orderings as well. Benchmarking figures and scripts used for testing `ergm.rank` can be found at ~~Liang [2025].~~ Just give the URL here.

## 4.1 Dealing with Computational Time Issues

The reader doesn't need to know about function names. They can be in the appendix.

Ideally, we aim to model networks with up to 100 individuals. However, this proves to be a problem in time complexity, as the functions which consider how proposed changes to the network affect deference and local or global nonconformity, namely `c_deference`, `c_nonconformity` and `c_localAND_nonconformity` respectively, have significant computational run times. There are also some other functions which were changed, but these functions were the main issues, each with a time complexity of up to $\mathcal{O}(n^3)$.

In order to account for these issues, a new auxiliary structure was devised and created for each network structure. In the original functionality of `ergm.rank`, the only auxiliary structure was `sm`, which was a matrix of values which represented the rank each ego gave to an alter. For example, `sm[i][j] = 4` means that the ego $i$ gave the alter $j$ a ranking of 4. As a result, when recalculating statistics such as deference and nonconformity, it would be necessary for `ergm.rank` to iterate over every individual in a network.

| sm[i][0] | sm[i][1] | sm[i][2] | ⋯ | sm[i][N_NODES-1] |

Too small:  Iterates over for each ego i
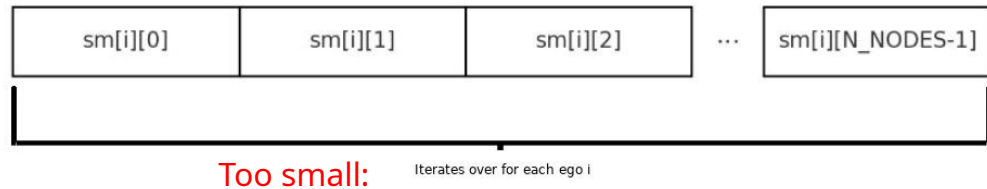
**Figure 4.1:** When recalculating statistics such as deference and local or global nonconformity, `ergm.rank` currently iterates over every alter $j$ for each ego $i$ in the network.

The new auxiliary structure for implemented for `ergm.rank`, called `udsm`, is instead a matrix of structures which keep track of which alter is immediately above it and which alter is immediately below it.

13

**Figure 4.2:** A visual representation of the struct created for `udsm`.

For example, for an ego $i$ and an alter $j$, `udsm[i][j].up = k` means that the alter $k$ is the individual ranked immediately above $j$ by $i$. Then, it becomes more cost efficient to iterate over the network, since it would only be necessary to iterate between two rankings when recalculating statistics, as evident in the formulas from Chapter 2.

Suppose we arbitrarily choose 2 points on a line uniformly. On average, the distance between these points would be a third of the length of the line. As such, it's expected that when iterating only between the old value which $i$ ranked $j$ and the new value that $i$ ranked $j$, the time complexity would be cut by two-thirds.



This is very good, but I'd make it more "mathematical" if you can.

Add the algorithm for updating the udsm: that's a contribution.
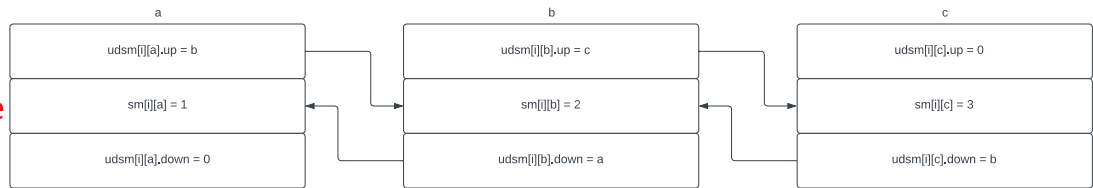
**Figure 4.3:** Consider an ego $i$ and their rankings of alters $a, b$ and $c$. Then, if `sm[i][a]` were changed, using `udsm` we can start at the original value for `sm[i][a]` and iterate either up or down to the new value of `sm[i][a]`, allowing us to easily recalculate statistical measures without iterating over individuals who were not affected by the change.

By tracking adjacently ranked alters, this auxiliary structure allows us to make use of promotion statistics (discussed in Chapter 2) when recalculating metrics such as deference and nonconformity. In turn, this enables the network's proposals to be re-designed as more efficient small-step proposal moves, so that `ergm.rank` can be reasonably applied to larger networks without egregious computational times.

## 4.2   Proposals

Might be worth moving most of this to lit. review.

ERGMs use Markov chain Monte Carlo (MCMC) in order to simulate networks from the model, as well as estimate the parameters by comparing simulated and observed statistics. This MCMC relies on proposals, which specify a mechanism for generating a small random change to the current network to explore the sample space.

Krivitsky and Butts [2017] uses a Metropolis-Hastings sampling algorithm (Algorithm A1) which chooses and ego at random, and makes a proposal to switch the ranking of two of its alters. The algorithm generates a series of networks through these small changes, accepting or rejecting each proposed network based on the current parameter estimates. These sampled networks are then used to estimate expected network statistics, which are compared to the observed statistics to update the parameter estimates. This process repeats iteratively until the parameter estimates converge, yielding the MLEs for the underlying probability model that generated the observed rankings.

Algorithm A1, however, is designed for sampling from a complete rank ERGM, which makes the assumption that each ego provides a complete, strict ordering of all alters in the network. As such, it is important for us to design an algorithm to propose changes to both a complete and a partially ranked network, as well as these changes being as small as possible for efficiency.

A complete rank ERGM assumes that each ego provides a strict ordering of all alters with no ties and no omissions. Consequently, in the case of a complete ordering, the distribution of the orderings is that of a permutation of alters by the egos. As such, the smallest move we can make in a complete ordering is have an ego swap an adjacent pair of alters. At present, the current `ergm.rank` infrastructure implementation only has the `AlterSwap` proposal, which is applicable solely to ERGMs with complete orderings.

Comparatively, a partially ranked ERGM is encoded to account for ties and missing rankings when generating a proposal. This is accomplished by giving each alter a rating from 1 to $n-1$, where $n$ is the number of actors in the network, so that the distribution instead is the egos assigning repeatable values in that range to alters. The two proposals designed for partially ranked ERGMs in this project are `AdjacentAlterSwap` and `PartialDisc`. For this thesis, we have elected to use `AdjacentAlterSwap` for the purposes of testing of the proposed infrastructure and fitting partially ranked networks, but it is still worthwhile to distinguish between the two. As discussed in Section 3.1, the proposals will treat a partially ranked dataset as if it were completely ranked, effectively treating the missing data as censored rather than absent.

### 4.2.1 PartialDisc

You might want to clarify that this is for when the underlying model is about partial orders.

For `PartialDisc`, the smallest move we can make in a partial ordering is either incrementing or decrementing the rating of an alter by an ego. However, this implementation runs into a problem: what if we are currently looking at an alter rated $n-1$ by an ego, or rated 1? In that case, it would be impossible to increment up or down respectively. Consider a stationary distribution $p(y) \propto \exp(\eta \cdot g(y)) = \pi(y)$ for some parameter $\eta$ and $g(y)$ is a function of network $y \in Y$. Then, suppose we

propose a change to the state of $y$, $y^* \, q(\cdot | y^t)$. We denote the acceptance probability from the Metropolis-Hastings algorithm as:

$$\alpha = \min \left( 1, \frac{\pi(y^*)}{\pi(y)} \times \frac{q(y | y^*)}{q(y^* | y)} \right)$$

$$\log(\alpha) = \min \left( \log 1, \log \frac{\pi(y^*)}{\pi(y)} + \log \frac{q(y | y^*)}{q(y^* | y)} \right)$$

$$= \min \left( 0, \log \frac{\pi(y^*)}{\pi(y)} + \log \frac{q(y | y^*)}{q(y^* | y)} \right)$$

$$= \min \left( 0, \log \frac{\exp(\eta \cdot g(y^*))}{\exp(\eta \cdot g(y))} + \log \frac{q(y | y^*)}{q(y^* | y)} \right)$$

$$= \min \left( 0, \log \exp(\eta \cdot g(y^*) - \eta \cdot g(y)) + \log \frac{q(y | y^*)}{q(y^* | y)} \right)$$

$$= \min \left( 0, \eta \cdot (g(y^*) - g(y)) + \log \frac{q(y | y^*)}{q(y^* | y)} \right)$$

We accept the proposal if for $U \sim \text{Unif}(0, 1)$ we have $U \leq \alpha \Leftrightarrow \log U \leq \log \alpha$.

$$\log \frac{q(y | y^*)}{q(y^* | y)}$$

is commonly referred to as the Hastings Ratio [Holder et al., 2005]. Introducing this term explicitly helps separate the effects of the network change, represented by the function

$$\eta \cdot (g(y^*) - g(y)),$$

and the proposal asymmetry. Note that $q(y^* | y)$ represents the probability of going from state $y$ to state $y^*$, and vice-versa for $q(y | y^*)$.

Now, consider the edge-state for an altar rated the maximum or minimum by an ego ($n$ and 1 respectively). Because we can't increment/decrement beyond the edges, the proposal distribution becomes asymmetric. So, the probability for $q(y | y^*)$ and $q(y^* | y)$ would not be the same, as for a proposed state $y^* = \max - 1$, the probability of $q(y^* | y) = 1$, whereas the probability of $q(y | y^*) = 0.5$ since at $y^* = \max - 1$ the edge-state can increment up or down.

This is similarly the case for the minimum, $y = 1$, and so in both cases we have that

$$\log \frac{q(y | y^*)}{q(y^* | y)} = \log \frac{0.5}{1} = \log \frac{1}{2}.$$

Similarly, for the edge-states where an altar is rated one above the minimum or one below the maximum, the probability would be flipped, and we'd instead have

$$\log \frac{q(y|y^*)}{q(y^*|y)} = \log \frac{1}{0.5} = -\log \frac{1}{2}.$$

For any other edge-state, the move probabilities are symmetric, so the proposal log-ratio is

$$\log \frac{q(y|y^*)}{q(y^*|y)} = \log \frac{0.5}{0.5} = 0.$$

That is, the log acceptance probability changes when an alter is ranked either on or next to the maximum or minimum by an ego, and has been accounted for in the proposal to swap adjacently ranked alters.

### 4.2.2 AdjacentAlterSwap

Comparatively, the `AdjacentAlterSwap` proposal is similar to the `PartialDisc` proposal, with the main difference being that the former always proposes "up" using the new auxiliary structure, and simply abandons the proposal if it happens to pick a node already at the top. This case occurs with probability $1/(n-1)$ of the time, meaning it does not need adjustment as it only slightly hurts the efficiency of the proposal. As such, there is no edge cases which `AdjacentAlterSwap` needs to account for when considering the log acceptance probability, making it much simpler for the purposes of testing the infrastructure and fitting partial datasets for this project.

For future work, the `PartialDisc` proposal could be fully integrated into the R side of the package. Since the C implementation is already in place, this would allow us to evaluate its efficiency in practice and assess whether it provides advantages over `AdjacentAlterSwap`.

CHAPTER 5

# Implementation and Evaluation using `ergm.rank`

Using the new auxiliary structure, we apply the framework to larger networks to benchmark computational time. We also evaluate whether the redesigned proposal mechanism produces accurate MLE estimates on partially observed networks, compared to MLE estimates obtained from the full network.

## 5.1 Dataset and Network Structure

The Newcomb Fraternity Data [Newcomb, 1961a] is a classic dataset used in social network and psychology research. Conducted by Theodore M. Newcomb at the University of Michigan, it was focused on exploring how individuals' preferences and social interactions evolve in a new group setting over time.

The dataset consists of fifteen, $17 \times 17$ matrices which recorded the sociometric preference rankings from 17 men across 15 weeks. Each week, participants would rank the other 16 men from best friend, down to least friend. Due to its small and fully observed nature, the Newcomb dataset has been widely studied, and in this project serves as a benchmark for testing the time complexity and accuracy of fitted missing data MLEs using the newly implemented infrastructure for `ergm.rank`. Note that week 9 is missing, as participants were not present for that week.

## 5.2 Implementation

For this project, a script was written to perform the benchmarking experiments for evaluating the proposed auxiliary structure in the `ergm.rank` framework. The benchmarking scripts were implemented in R, and to assess the efficiency and accuracy of model fitting, for both complete and missing datasets.

### 5.2.1 Model Fitting

The benchmarking process takes the Newcomb dataset and generates a modified version containing missing data. In the first benchmark script, all rankings outside of the top five were replaced with NA to imitate a missing dataset. Two models are then fitted for each of the fifteen weeks in the dataset to estimate the parameters:

1. **Complete data MLE:** The maximum likelihood estimate (MLE) using the fully observed rankings
2. **Missing data MLE:** The MLE fitted using the partially observed dataset.

For the missing data model, it is assumed that there exists a complete ordering beneath the observed partial ordering, so that the `AdjacentAlterSwap` proposal

can be used. The fitted MLEs for each week are stored in an indexed list.

### 5.2.2   Evaluation Metrics

A set of utility functions are used to extract and compare key model statistics between the complete data and missing data fits. For each week, the script computes and stores:

**Get rid of R commands here.**

- Coefficients: (`coef()`)
- Standard errors: (`sqrt(diag(vcov()))`)
- Log-likelihood at the MLE: (`logLik()`)

### 5.2.3   Timing and Output

The total runtime is computed and written to a file alongside model results. The scripts also generate visual comparisons of the evaluation metrics computed across the fifteen weeks, allowing for direct comparison between the results from the complete data MLE and missing data MLE.

## 5.3   Computational Results and Analysis of Newcomb [1961a]

Having established the benchmarking framework and described the model-fitting procedure, we run the benchmarking scripts and obtain the evaluation metrics from the missing data and complete data MLEs, visually represented through the presented figures.

### 5.3.1   Deference

For the [Newcomb, 1961a] dataset, we see that parameter estimates of deference for both the constrained data MLE and the complete data MLE are negative throughout the entire fifteen weeks. This suggests that deference is disfavoured by the model, meaning the model would generate networks where deference is less prevalent compared to a network generated by ranking randomly. That is, there may not be a deferential structure, where people reciprocate when others rank them highly, and instead a structure that results in a competitive tendency between individuals in the network. We also generally see that the parameter estimates for deference generally trends downwards as the weeks progress, implying the network strayed further from a deferential structure as time passed.

The parameter estimates for the constrained model are within one standard error of those from the true model, indicating that little information was lost by restricting the data to the top five rankings. Interestingly, the constrained model yields more negative parameter estimates, suggesting that even less deference is captured when by the top 5 rankings. That is, an individual's top choices appear to be driven less by deference and more by other factors such as personal preference or affinity.
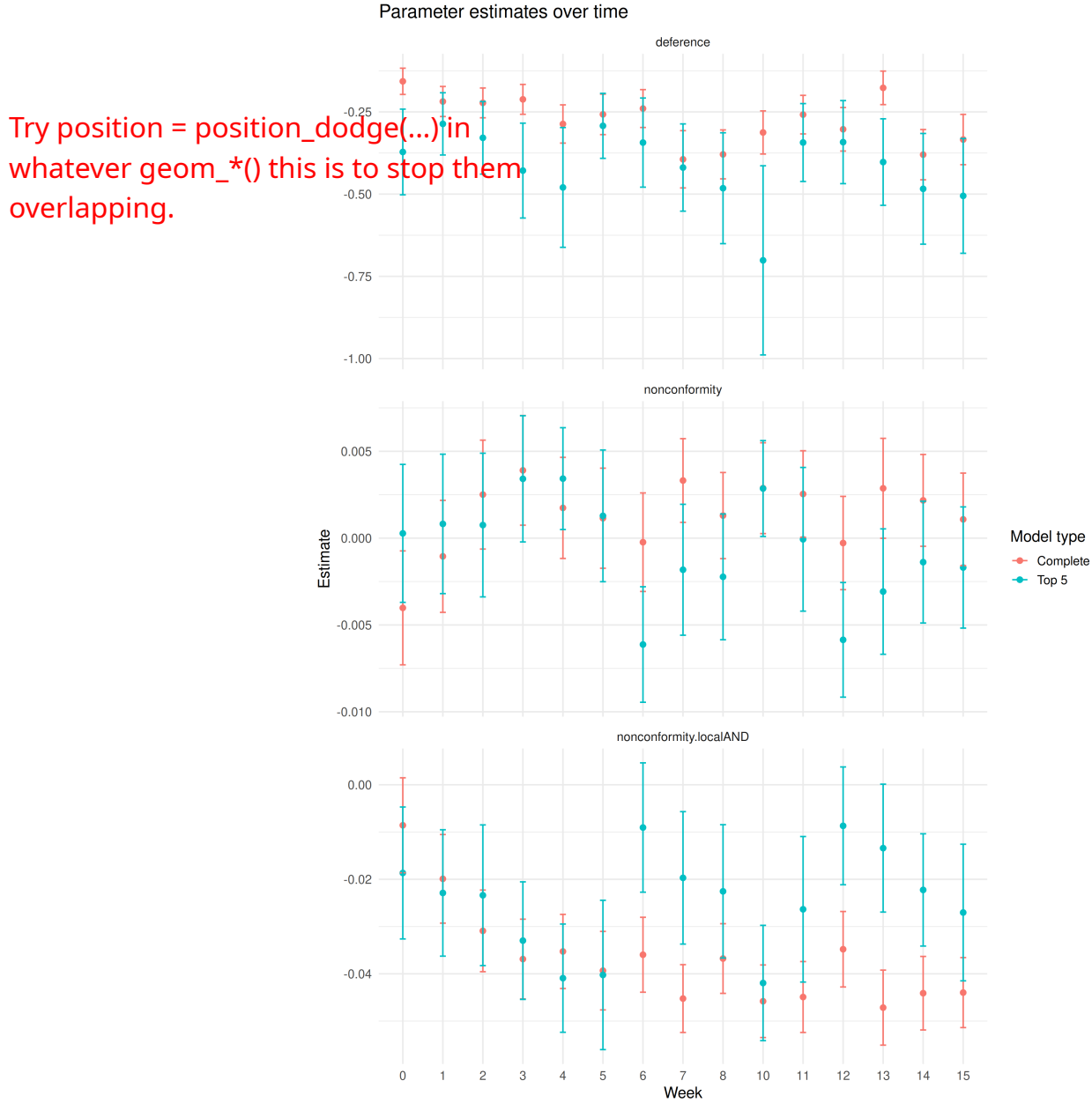
**Figure 5.1:** Parameter estimates over time for deference, global nonconformity and local+global nonconformity, where error bars indicate ±1 standard error. Note that Week 9 is missing as no rankings were recorded.

### 5.3.2 Global Nonconformity

The parameter estimates for global nonconformity under the complete-data MLE generally stay close to 0 throughout the time period, indicating that the level of global nonconformity in networks generated from the fitted model is not noticeably different from that expected under a random baseline. Note that the most negative parameter estimate occurs in the first week, likely reflecting the influence of initial impressions formed before participants had established any friendships. In the context of the [Newcomb, 1961a] dataset, the results imply that participants' rankings of others evolved largely independently, without contradicting any general

20

consensus that may have formed within the group.

Comparatively, for the fitted MLE using only the top-five rankings, the parameter estimates for global nonconformity show similar values to the true model, but trend slightly below it as the weeks progress. This suggests that in the early weeks, there was limited agreement among individuals regarding whom they ranked in their top five, which follows as social circles and friendships had not yet been formed. However, as time went on, individuals appeared to reach more of an agreement about whom they considered their top five friends compared to others in the group. Very little information was lost by restricting the data to the top-five rankings, even when compared to the deference parameter, as the estimates between the two models lie within half a standard error of each other for a majority of the weeks.
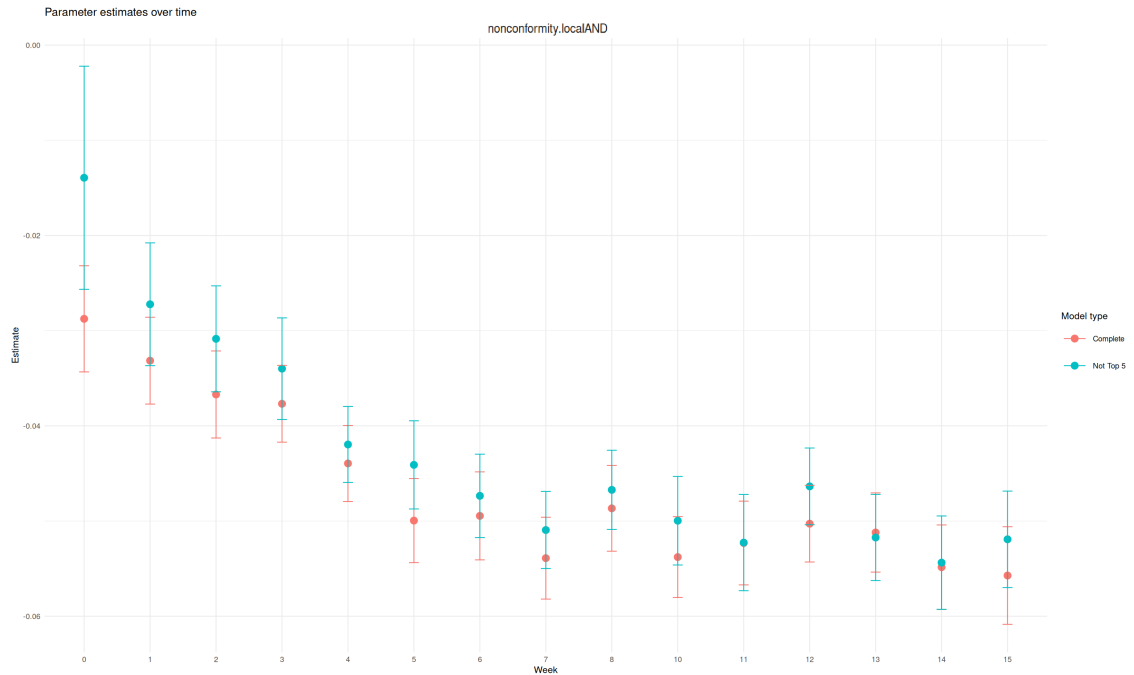
### 5.3.3   Global and Local Nonconformity

Interestingly, we see the biggest discrepancy between the parameter estimates for the complete and top 5 model when we consider both global and local nonconformity. Most notably, from week 6 onwards, we see that the parameter estimates for the top 5 model are noticeably higher than compared to the parameter estimates using the complete data. We recall that local nonconformity factors in that egos may be more likely to conform to those whom they rank highly, over those whom they rank lower.

Then, this difference could be explained by the fact that local nonconformity cannot be captured with the absence of the middle and lower portions of the rankings. In the complete data, egos and their highly ranked alters tend to agree more broadly across the full ranking, which reduces local nonconformity. When only the top five rankings are observed, this additional agreement is no longer visible to the model, leading to consistently higher estimates of local nonconformity, once individuals have established who they consider to be their close friends as social circles form. While this is important to keep in mind when interpreting local nonconformity, the discrepancies in parameter estimates are small enough to indicate that the proposal method remains robust to missingness, even for this statistic.

Interestingly, when we fit the MLE on a modified dataset that excludes the top-five rankings, the trend is reversed. In this case, the parameter estimates for the partial data MLE are higher than those of the complete data MLE during the first six weeks, before converging to similar values in later weeks. This supports the idea that local nonconformity is not captured by the absence of the middle and lower portions of the rankings.

**Figure 5.2:** Parameter estimates over time for local+global nonconformity, fitted on data where the top 5 rankings for each ego is missing.

### 5.3.4 Time Efficiency of `AdjacentAlterSwap` compared to `AlterSwap`

It is also important to compare the computational performance of the two proposal mechanisms on complete orderings. Fitting an ERGM to the full Newcomb [1961a] dataset for the key statistics of deference, global nonconformity, and combined local–global nonconformity using the current `ergm.rank` implementation with the `AlterSwap` proposal requires approximately 2229 seconds. In contrast, using the revised implementation, incorporating the new auxiliary structure together with the `AdjacentAlterSwap` proposal, reduces the total computation time to roughly 475 seconds. Interestingly, using `AlterSwap` but using the revised implementation for the calculations of the key statistics takes approximately 257 seconds. This indicates that `AlterSwap` is inherently more efficient than `AdjacentAlterSwap`, although it is limited to complete rankings.

This is possibly due to the fact that `AdjacentAlterSwap` performs additional lookups in the auxiliary structure and enforces constraint checks for each proposed toggle, whereas `AlterSwap` relies on simple random selection and minimal computation, making it inherently faster for complete orderings. Importantly, `AlterSwap` cannot be applied to partial orderings, as it assumes all rankings are well-defined, with no missing or tied entries. Nevertheless, these results demonstrate that our proposed implementation of the up-down auxiliary structure substantially improves computational efficiency. For future work, we could explore different model parameters or benchmark performance for a given sample size, and extract statistics to estimate the effective sample size per second of runtime.

A few notes:
We could very easily enable AlterSwap to fit to partially observed networks, so that's not really a disadvantage.

I would suggest redoing this benchmark with snctrl(init.method = "zeros") for all of them, since, as I mentioned, their starting values might differ.

This is because the AlterSwap makes bigger proposals, which may make things better or worse for MCMLE, but almost certainly better in for Contrastive Divergence initial value (Krivitsky, 2017, Using Contrastive...).

22

### 5.3.5   Robustness of the Proposal Method to Partial Rankings

Overall, the results from fitting MLEs to the Newcomb [1961a] dataset show that little information is lost when restricting the data to partial rankings, as the parameter estimates remain close to those obtained from the complete rankings. These findings indicate that the proposed auxiliary structure, along with the modifications to the MCMC sampling algorithm, exhibits satisfactory robustness to missingness. These findings give confidence that the modified `ergm.rank` infrastructure can be applied to real-world datasets with genuine missing entries, while still yielding reliable and interpretable estimates.

Where you able to run the simulation with effectiveSize per second?

# CHAPTER 6

# Applying `ergm.rank` to Incomplete Real-World Data

Having demonstrated that the proposed implementation of `ergm.rank` both improves computational efficiency and remains robust to missing data when applied to partially ranked datasets, we now turn to its application on real-world, incomplete data.

## 6.1 The Sampson Dataset

The Sampson Monastery dataset [Sampson, 1968] is included in the `ergm` package. It describes the social relations among a group of men who are preparing to join a monastic order. The dataset records four distinct types of relations: esteem, liking, influence, and praise, and has separate matrices capturing both positive and negative ties for each relation. In this study, we focus on the liking relation. The Sampson dataset for the liking relation consists of six networks: `samplk1`, `samplk2`, `samplk3`, `sampdlk1`, `sampdlk2` and `sampdlk3`, where they represent three time points for liking and three time points for disliking. The liking relation datasets allow each ego to assign rankings to individuals from 1 to 3, where 3 represents the highest rank, with ties allowed, whereas the disliking relation dataset reverses this, with -3 representing their most disliked alter.

These three time points were in the period during which a new cohort had entered the monastery near the end of the study but before the major conflict began, which resulted in the expulsion of four monks. Sampson [1968] divided the novices into four groups: Young Turks, Loyal Opposition, Outcasts, and an interstitial group. The Loyal Opposition consists of the novices who entered the monastery first, and the Young Turks arrived later. The latter questioned practices in the monastery, which the former defended. The who novices did not take sides in this debate were labeled 'interstitial'. The Outcasts are novices who were not accepted in the group.

Each member of the monastery was asked to rank only his top three choices for both relations, with the rest of the rankings being unknown. Most of the data were collected retrospectively, after the conflict had occurred, which may reduce their reliability. In contrast, the general consensus is that the liking data was collected contemporaneously, making them the primary focus of this study. This is precisely the type of dataset that the previous implementation of `ergm.rank` was unable to accommodate for. With the updated infrastructure, we can now fit an MLE to the missing rankings, allowing us to learn about the underlying preference structure, which we previously could not. The main motivation of fitting on this dataset is to

see whether the observed rankings from these relations could have provided early indications of this conflict.

## 6.2 Implementation

Similarly to the Newcomb [1961a] dataset, a script was written to use the auxiliary structure in the `ergm.rank` to extract parameter estimates from key model statistics.

### 6.2.1 Model Fitting

The script takes each of the six networks and fills in the missing rankings based on the observed values, producing an initial network with a complete ordering. It then estimates the model parameters (MLE) for each network, while constraining the fit to the rankings that were actually observed, using `AdjacentAlterSwap`, storing the fitted MLEs for each network in a list. Currently, the initial parameter estimates are set to zero. A potential extension would be to first fit a model using zero starting values, and then use the resulting parameter estimates as starting points for a second fit. However, this dramatically increases computational time, as random or uninformed starting values generally leads to slower convergence and longer MCMC burn-in periods. As a result, this approach was not feasible within the constraints of this project.

### 6.2.2 Evaluation Metrics

The same utility functions from the benchmarking script were used to extract the key model statistics, computing and storing:

Don't include the R commands here.

- Coefficients: (`coef()`)
- Standard errors: (`sqrt(diag(vcov()))`)
- Log-likelihood at the MLE: (`logLik()`)

### 6.2.3 Output

The script generates visual comparisons of the evaluation metrics across the three time points for both the liking and disliking relations, allowing us to analyze how deference, global nonconformity, and local/global nonconformity changed over the study period.

## 6.3 Computational Results and Analysis of Sampson [1968]

### 6.3.1 Liking Relation Analysis

*Deference*

We observe that across the three time points, the estimated deference parameter for the Sampson liking relation stays negative at approximately -0.5. Such values indicate a low degree of hierarchical or deferential behaviour within the monastery, as initiates did not systematically rank one another in any consistent way. Even accounting for standard error, the estimated parameter for deference suggests there was variation in how individuals ranked one another and a general lack of reciprocal respect. This pattern is consistent with the tension reported during the study period and may have contributed to the conflict that ultimately unfolded.
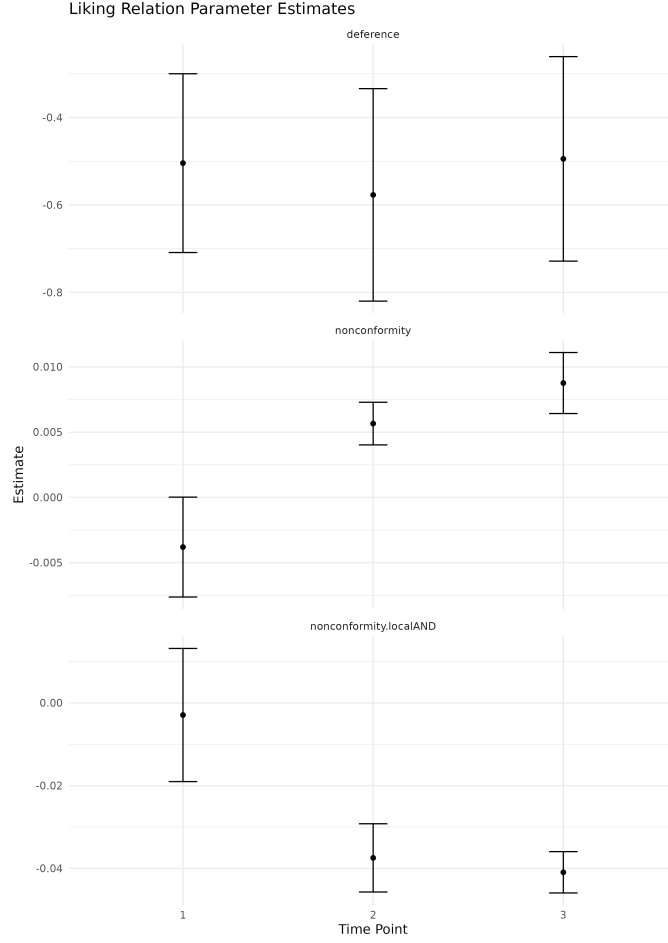
**Figure 6.1:** Parameter estimates over time for deference, global nonconformity and local+global nonconformity using the likeness relation, where error bars indicate ±1 standard error.

*Nonconformity*

Global nonconformity rises over the three time points, indicating that individuals increasingly diverged from the dominant ranking patterns in the community. This is likely because early on, before the arrival of the Young Turks, the Loyal Opposition formed a relatively cohesive core. As new groups emerged when the new cohort arrived at around Time Point 2, ranking behaviour became more fragmented, which is reflected in the growing nonconformity parameter. However, the parameter estimates for global nonconformity are not statistically significant, and are not indicative of the dynamics leading to the monastery's breakup. It is highly possible that global nonconformity was 0 over the entire time period, where the small fluctuations across time could possibly be due to information loss from observing only the top three rankings for each monk. It is therefore more meaningful to focus on nonconformity when the local component is included.

Local nonconformity reflects how individuals diverge specifically from those they rank highly, which offers nuance that global measures cannot capture, especially during periods of social change. Figure 6.1 shows that local/global nonconformity starts near 0 and dramatically drops to around -0.04 across both of the next two

time points. Similarly to global nonconformity, local nonconformity was likely near 0 initially, as the Loyal Opposition comprised most members, leading to mutual respect among novices. After the arrival of the Young Turks, we see that discrepancies emerge, as individuals increasingly align their rankings with those they rank highly. This likely reflects the formation of these groups, as initiates conform more closely to the preferences of their in-group members, who be ranked highly compared to members who are outside of their group, who they would tend to rank lower. These parameter estimates for global and local nonconformity highlight the emerging divide between groups at the monastery, tensions that likely contributed to its eventual breakup.

*Insights from the Liking Network*

Evidently, The parameter estimates of the liking relation reflect the divide which split the novices into groups as the Young Turks arrived in Time Point 2. Most notably for deference and local/global nonconformity, the parameter estimates exhibit early signs of the lack of respect between novices outside of their respective groups, the tension of which likely came to a head at a later time. Overall, the implemented `ergm.rank` infrastructure successfully fit an MLE to the partially observed Sampson [1968] rankings, yielding results that are consistent with the known context and narrative of events at the monastery.

*6.3.2  Disliking Relation Analysis*

Unfortunately, attempts to fit ERGMs to the three disliking networks (`sampdlk1`, `sampdlk2`, `sampdlk3`) were less successful. While `sampdlk1` converged to a reasonable MLE, `sampdlk2` and `sampdlk3` had extremely long run times and did not reach convergence within feasible computational limits. This issue underscores the challenges faced when fitting MLEs to sparse partial rankings. When so little information is available, parameter estimation is unstable and is prohibitively slow to converge.

You might want to provide some references for this. Since the disliking relations were collected retrospectively, after the conflict had already unfolded, it is likely that these rankings were less reliable compared to the liking data. Retrospective recall, especially for negative relations, tends to introduce noise, resulting in less coherent structural patterns for the ERGMs to capture. As a result, the ERGMs for disliking relations lacked consistent structural patterns for parameter estimation, yielding a poorly defined likelihood and preventing the algorithm from converging to an MLE. Future works could look into improving the infrastructure to successfully fit these datasets, or simply more powerful hardware to reduce run times. We can, however, examine the results from fitting `sampdlk1` and compare them to the corresponding liking network at the same time point, `samplk1`.

*Deference*

Interestingly, we see that for Time Point 1, the parameter estimates for deference fitted to `sampdlk1` was higher than the parameter estimate from `samplk1`. At first glance, this seems surprising for deference: we would expect deference aversion
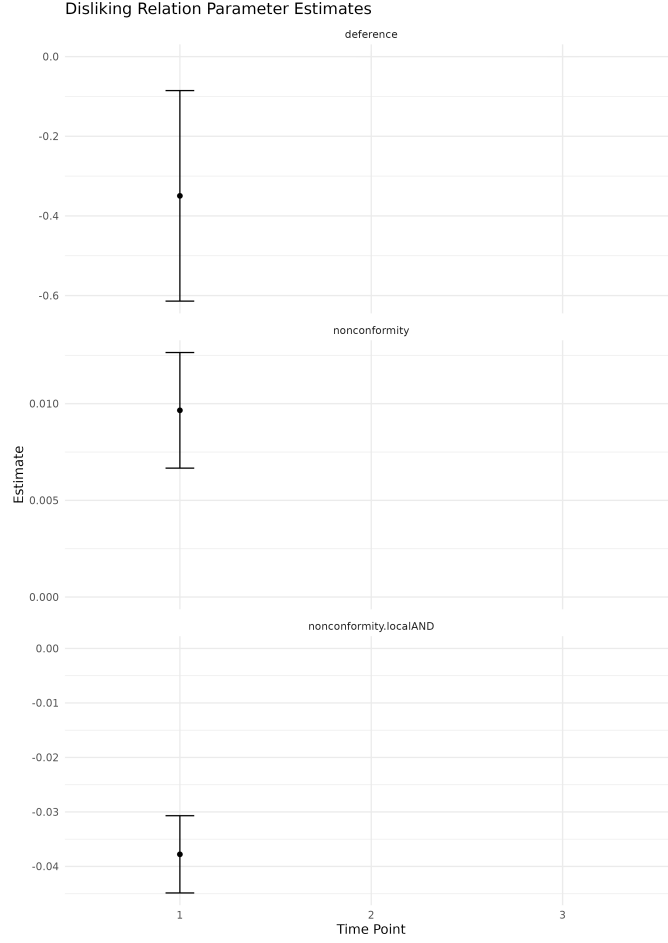
**Figure 6.2:** Parameter estimates over time for deference, global nonconformity and local+global nonconformity using the dislikeness relation, only for Time Point 1 due to issues when fitting the MLE.

to be easier to capture when the observed rankings are the lowest rankings. A plausible explanation is that disliking relations tend to be more weakly structured than liking relations. Negative rankings are often less consistent and transitive, that is, individuals may agree on who they like, but disagree much more about who they dislike [McMillan et al., 2022]. Then, we would expect that models trained on disliking relations likely capture weaker and noisier patterns, making parameter estimation less reliable. Ideally, we could use additional bottom-ranked information, such as the bottom five or six rankings, as it would let us investigate whether this effect persists. Unfortunately, the Sampson [1968] dataset does not include such information.

*Nonconformity*

The difference between the parameter estimates global nonconformity between `samplkl1` and `sampdlk1` are noticeably different, with the former being positive for Time Point 1. This is likely influenced by the noisy and weaker patterns of dislking relations. As previously mentioned, negative rankings are less consistent and less transitive when compared to positive rankings [McMillan et al., 2022], so

people would be less likely to come to a consensus on who they dislike, thus increasing global nonconformity.

The biggest difference between the two fitted networks is the parameter estimates for local/global nonconformity. For Time Point 1 of `samplk1` and `sampdlk1`, the parameter estimate for local/global nonconformity is close to 0 for the former, and approximately -0.037 for the latter. We recall that local nonconformity increases if an ego conforms to those whom they rank highly, over those whom they rank lower. Interestingly, this would imply that the novices in Time Point 1,primarily the Loyal Opposition, exhibited strong conformity toward those they personally ranked highly. This contrasts what was suggested by the corresponding liking relation parameter estimates. However, as previously mentioned, disliking relations generally contain weaker and less stable patterns, so these results should be taken skeptically.

*Insights from the Disliking Network*
Overall, we see that the parameter estimates fitted on Time Point 1 of the observed disliking relations do not align consistently with the results from the parameter estimates from the liking relation. However, these differences could simply be due to the unreliable and less transitive nature of negative rankings [McMillan et al., 2022]. By combining these two rankings together as one singular ranking for each time point, we likely would get more reliable results.

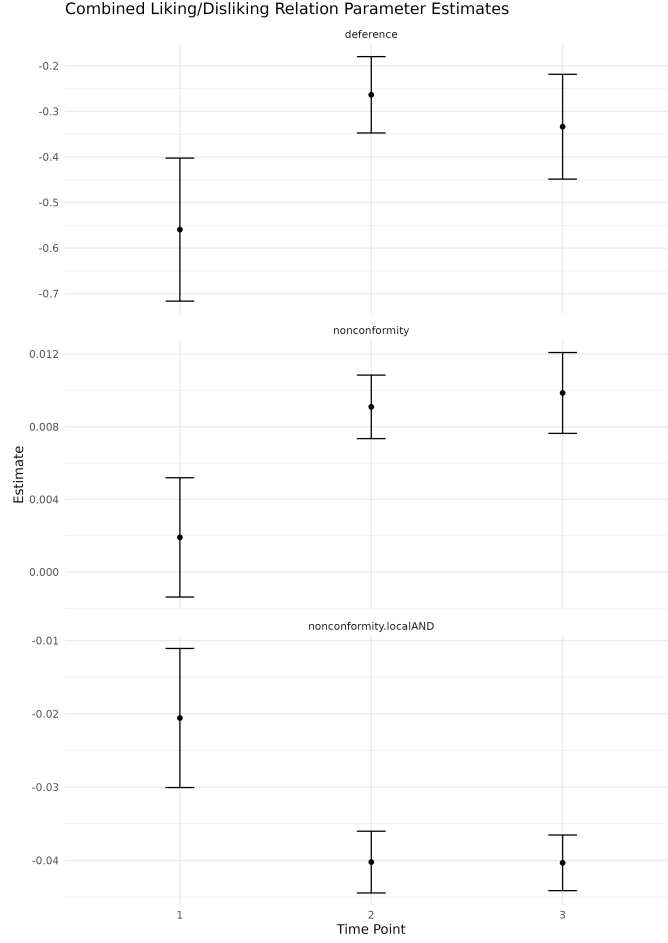*6.3.3   Combined Liking and Disliking Relation Analysis*
A more informative approach would be to combine the partial rankings from both the liking and disliking relations and use them jointly as constraints when fitting the MLE. Recall that the MCMC algorithm iteratively generates proposed networks under the current parameter estimates, recalculating the sufficient statistics, and updating the parameters until convergence. Then logically, providing the algorithm with more information about the true constraints would reduce the space of networks that are consistent with current parameter estimates. With fewer possible networks, the sampler explores a narrower, more informative region of the likelihood surface, allowing the algorithm to converge more quickly and to more stable and accurate parameter estimates. Then, combining the networks for the liking relation with their corresponding disliking relation, such as `samplk1` and `sampdlk1`, we use it as a constraint to fit `samplk1`, `samplk2` and `samplk3` to obtain the following figure.

The sampler explores the sample space, not the likelihood surface.

*Comparison to the Liking Relation Parameter Estimates*
We see that both global nonconformity and local/global nonconformity follow the same pattern throughout the three time points between Figures 6.1 and 6.3. Although the combined constraint estimates differ somewhat from those obtained using only the liking rankings, this discrepancy is expected. The additional information using the disliking rankings imposes stronger constraints on the space of feasible networks, allowing the algorithm to converge more accurately. As a result, the parameter estimates under the combined constraints are likely closer to the true values of the parameters. Despite their minor differences, both figures reflect the tensions which heightened between the groups at the monastery after the Young

**Figure 6.3:** Parameter estimates over time for deference, global nonconformity and local+global nonconformity for the constrained to the combined liking and disliking rankings.

Turks arrived in Time Point 2.

Perhaps the more interesting parameter estimate to compare between the two are for deference. We see that the parameter estimates for deference using the combined constraints. For Figure 6.1, we suggested that the consistently negative deference values reflected tension and a general lack of reciprocal respect among the novices. That interpretation remains appropriate for Time Point 1. However, unlike the liking relation, we now see that deference increases substantially at Time Point 2 and stays elevated at Time Point 3. In the context of the Sampson [1968] dataset, this aligns with the arrival of the new cohort, the Young Turks, most likely during Time Point 2, as suggested by Sampson's survey instructions to skip Question 1 if one joined after "[MONTH X]". As the novices divided into these four distinct factions, it may have introduced more structured group-based hierarchies. This group loyalty could have lead to stronger reciprocal respect among group members, thereby increasing the estimated deference parameter.

*6.3.4   Summary and Implications*

Overall, we can say with confidence that the parameter estimates obtained by fitting an MLE to the liking networks, namely `samplk1`, `samplk2`, and `samplk3`, while

constraining the model with both the liking and disliking partial rankings, were no-
ticeably more reliable. These estimates aligned more closely with the historical
context of the Sampson [1968] study.

While the models fitted using only the observed top rankings produced reason-
able and interpretable estimates, incorporating both the top and bottom rankings
as constraints provided substantially more information for the MCMC algorithm,
resulting in parameter estimates that more clearly reflected the underlying commu-
nity structure at the monastery.

Although most of the disliking networks alone did not converge, this is not surpris-
ing given the sparsity and unreliability of retrospective negative rankings. How-
ever, they were still useful in that their combination with the liking data yielded
constraints that were far more informative, ultimately producing better parameter
estimates.

# CHAPTER 7

# Discussion and Future Work

Overall, the implementation of the modified `ergm.rank` in this thesis successfully demonstrated the ability to fix maximum likelihood estimates (MLEs) to partially ranked network data.

## 7.1 Newcomb [1961a] Fraternity Data Insights

The application of the proposed implementation of `ergm.rank` to the Newcomb Fraternity Data [Newcomb, 1961a] confirmed that partially observed rankings can provide meaningful information about the underyling social structure. It was observed that, by fitting an MLE to both the partially ranked dataset and the complete dataset, we produced similar parameter estimates that gave insights to how the underlying network model evolved over time. For both fitted models, the estimates for deference, global nonconformity, and local/global nonconformity were consistent with those reported in Krivitsky and Butts [2017], demonstrating that the proposed `ergm.rank` is robust to missingness and can yield reliable results with real-world, partially observed data.

Most successfully, we found that the new auxiliary structure, together with the `AdjacentAlterSwap` proposal was much more efficient comparatively to the current `ergm.rank` implementation along with the `AlterSwap` proposal, when fitted to the full Newcomb dataset. We also interestingly found that `AlterSwap` with the revised implementation was more efficient than `AdjacentAlterSwap` when fitted to the Newcomb dataset. Future work could include further exploring and testing in what cases is one proposal better than the other, as well as fully implementing `PartialDisc` to compare to the other two.

## 7.2 Sampson [1968] Monastery Data Insights

The Sampson [1968] datasets illustrated both the capabilities and limitations of the proposed implementation of `ergm.rank`. Fitting MLEs to the liking networks (`samplk1`, `samplk2`, `samplk3`) produced parameter estimates that aligned well with historical context, highlighting the emergence of the social groups defined by Sampson, most notably the Young Opposition, novices who entered the monastery first, and the Young Turks, novices who arrived later. These patterns were more clearly defined when the MLEs were constrained not only by the observed liking networks but by a combination of the liking and disliking rankings.

In contrast, fitting the disliking networks (`sampdlk1`, `sampdlk2`, `sampdlk3`) revealed

the challenges associated with sparse or negatively signed data. Of the three networks, only `sampdlk1` successfully converged to an MLE, while the others required unfeasibly long runtimes and failed to converge, likely due to negative rankings being less consistent [McMillan et al., 2022]. Future work could explore optimisations to the `ergm.rank` infrastructure or more powerful hardware to fit these networks within reasonable computational limits.

# CHAPTER 8

# Conclusion

In this thesis, we proposed and implemented a modified version of `ergm.rank` capable of fitting maximum likelihood estimates (MLEs) to partially ranked network data. Using the newly implemented `AdjacentAlterSwap` proposal and its auxiliary structure, we demonstrated substantial improvements in computational efficiency, addressing optimisation issues when recalculating network statistics such as deference, global nonconformity, and local/global nonconformity.

Applications to the Newcomb [1961a] Fraternity dataset and Sampson [1968] monastery data demonstrated robustness to missingness, allowing partially observed rankings to yield meaningful insights into underlying network structures. Fitting MLEs to the latter dataset produced parameter estimates consistent with historical context, and captured patterns which hinted at the events which occurred after the timeframe of the dataset.

Future work could address limitations in the proposed `ergm.rank` infrastructure, particularly its difficulty in converging for sparse or negatively signed data. Nonetheless, these modifications provide a solid foundation for further optimization, and offer a valuable tool for statisticians seeking to fit ERGMs to partially observed ranked networks.

# References

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference (TCC 2006)*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284. Springer, 2006. doi:10.1007/11681878_14.

Holmes Finch. An introduction to the analysis of ranked response data. *Practical Assessment, Research, and Evaluation*, 27(1):7, 2022. doi:10.7275/tgkh-qk47.

Steven M. Goodreau, James A. Kitts, and Martina Morris. Birds of a feather, or friend of a friend? using exponential random graph models to investigate adolescent social networks. *Demography*, 46(1):103–125, 2009. doi:10.1353/dem.0.0045.

Mark S. Handcock and Krista J. Gile. Modeling social networks from sampled data. *The Annals of Applied Statistics*, 4(1):5–25, 2010. doi:10.1214/08-AOAS221.

Peter Hoff, Bailey Fosdick, Alex Volfovsky, and Katherine Stovel. Likelihoods for fixed rank nomination networks. *Network Science*, 1(3):253–277, 2013. doi:10.1017/nws.2013.17.

Mark T Holder, Paul O Lewis, David L Swofford, and Bret Larget. Hastings ratio of the LOCAL proposal used in bayesian phylogenetics. *Syst. Biol.*, 54(6):961–965, December 2005.

Vishesh Karwa, Pavel N. Krivitsky, Aleksandra B. Slavković, and Xiaoran Yu. Sharing social network data: differentially private estimation of exponential family random-graph models. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 66(3):481–500, 2017. doi:10.1111/rssc.12185.

Pavel N. Krivitsky. Exponential-family random graph models for valued networks. *Electronic Journal of Statistics*, 6:1100–1128, 2012. doi:10.1214/12-EJS696.

Pavel N. Krivitsky and Carter T. Butts. Exponential-family random graph models for rank-order relational data. *Sociological Methodology*, 47(1):68–112, 2017. doi:10.1177/0081175017692623.

Pavel N. Krivitsky, Pietro Coletti, and Niel Hens. A tale of two datasets: Representativeness and generalisability of inference for samples of networks. *Journal of the American Statistical Association*, 118(544):2213–2224, 2023a. doi:10.1080/01621459.2023.2242627.

Pavel N. Krivitsky, Alina R. Kuvelkar, and David R. Hunter. Likelihood-based inference for exponential-family random graph models via linear programming. *Electronic Journal of Statistics*, 17(2):3337–3356, 2023b. doi:10.1214/23-EJS2176.

Patrick Liang. Honours. https://github.com/kinguinpenguin/honours_unsw, 2025.

Cassie McMillan, Diane Felmlee, and James R Ashford. Reciprocity, transitivity, and skew: Comparing local structure in 40 positive and negative social networks. *PLoS One*, 17(5):e0267886, May 2022.

Sumit Mukherjee. Degeneracy in sparse ergms with functions of degrees as sufficient statistics. *Bernoulli*, 26(2):1016–1043, 2020. doi:10.3150/19-BEJ1135.

Theodore M Newcomb. *The acquaintance process.* Holt, Rinehart & Winston, New York, 1961a.

Theodore M. Newcomb. The acquaintance process, 1961b.

Ikewelugo Cyprian Anaene Oyeka, Chike H. Nwankwo, and Awopeju K. Abidemi. The g-method for rank determination in rank order statistics for 'c' samples. *American Journal of Theoretical and Applied Statistics*, 3(1):18–24, 2014. doi:10.11648/j.ajtas.20140301.13.

Samuel Frank Sampson. *A Novitiate in a Period of Change: An Experimental and Case Study of Relationships.* Ph.d. dissertation, Cornell University, 1968. Department of Sociology.

Michael Schweinberger, Pavel N. Krivitsky, Carter T. Butts, and Jonathan R. Stewart. Exponential-family models of random graphs: Inference in finite, super and infinite population scenarios. *Statistical Science*, 35(4):627–662, 2020b. doi:10.1214/19-STS743.

Johannes van der Pol. Introduction to network modeling using exponential random graph models (ergm): Theory and an application using r-project. *Computational Economics*, 54(3):845–875, 2018. ISSN 0927-7099. doi:10.1007/s10614-018-9853-2. © 2018, Springer Science+Business Media, LLC, part of Springer Nature.