Carnegie Mellon University
Dietrich College of Humanities and Social Sciences
Dissertation

Submitted in Partial Fulfillment of the Requirements
For the Degree of Doctor of Philosophy

Title: Learning Social Networks from Text Data using Covariate Information

Presented by: Xiaoyi Yang

Accepted by: Department of Statistics & Data Science

Readers:

Nynke M.D. Niezink, Advisor

Rebecca Nugent, Advisor

Brian Junker

Cosma R. Shalizi

Christopher Warren

Approved by the Committee on Graduate Degrees:

Richard Scheines, Dean Date

Carnegie Mellon University

Learning Social Networks from Text Data using Covariate Information

A Dissertation Submitted to the Graduate School

in Partial Fulfillment of the Requirements for the degree

Doctor of Philosophy

in

Statistics

by

Xiaoyi Yang

Department of Statistics & Data Science
Carnegie Mellon University
Pittsburgh, PA 15213

Carnegie Mellon University

August 2021

Dedicate to my dear cat, Rainne

Acknowledgements

I would like to thank my thesis advisors, Dr. Nynke Niezink and Dr. Rebecca Nugent, for their help during my PhD years. They not only show me how to do the research but also show me the rigorous and critical thinking that a scientist should maintain. I want to express my deepest appreciation to their patience and positive attitudes when I make mistakes, which is a strong motivation for my work.

I would also like to thank my committee members, Dr. Cosma Shalizi, Dr. Brian Junker and Dr. Christopher Warren for their valuable suggestions on this thesis. In addition, I would like to thank the professors that I used to work with, Dr. Robin Mejia and Dr. Jared Murray. Working with them allowed me to learn various applications of statistics and know how to work with people from other fields. Also, a great gratitude to Dr. Joel Greenhouse, Dr. Peter Freeman, Prof. Gordon Weinberg, Dr. Zach Branson and Dr. Alex Reinhart for being excellent role models on how to teach statistics and work with students.

I am also thankful to my cohort and all students in the department for the continuous support during the past few years. The warm and friendly environment here makes Pittsburgh another home for me. In addition, I would like to give a special thanks to my friend Dr. Qifan Li and all other friends I know from social media platforms. Getting to know PhD students from different backgrounds not only gives me opportunities to think about the potential of statistics in different fields but also makes me feel connected in all aspects.

Moreover, I want to express my gratitude to my parents, for their support on my academia career. I would also like to thank Dr. Robert and Stacey Wettstein for their mental support.

In the end, I would like to thank my dear cat Rainne, for her loyal company in the past four years.

Abstract

Accurately describing the lives of historical figures can be challenging, but unraveling their social structures perhaps is even more so. Historical social network analysis methods can help in this regard and may even illuminate individuals who have been overlooked by historians, but turn out to be influential social connection points. Text data, such as biographies, are a useful source of information for learning historical social networks but the identification of links based on text data can be challenging. Traditional methods directly use the number of name co-mentions in the text to infer relations. The use of a conditional independence structure reduces the tendency to overstate the relationship between "friends of friends". However, this method does not take into account the abundance of covariate information that is often available in text data.

In this work, we first explore the effect of multiple conditional independence structures on reconstructing social network from the text. Then we extend the Local Poisson Graphical Lasso model with a (multiple) penalty structure that incorporates covariates, opening up the opportunity for similar individuals to have a higher probability of being connected. We propose both greedy and Bayesian approaches to estimate the penalty parameters and present results on data simulated with characteristics of historical networks and show that this type of penalty structure can improve network recovery as measured by precision and recall. We also illustrate the approach on biographical data of individuals who lived in early modern Britain between 1500 to 1575. Finally, we show the model can also incorporate continuous covariates and discuss several applications of how to create continuous covariates from text data.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

A social network is a structure representing the relationships in a social community. Typically, a node in a social network represents a person or an organization while a link represents a social interaction, such as knowing each other, between two nodes. Characterizing and understanding the social network of a community can help us to learn how information flows within the community and how the individuals and sub-communities interact with each other. When reconstructing a modern social network, the process is relatively simple and flexible, since accessing information about or from each individual could be done through multiple sources. For example, to learn someone's personal social network, we could either survey them or use information from social media platforms. In an academic setting, co-authorship information from curriculum vitae, journals, or online archives can provide information about collaborations.

There is also interest in learning and understanding historical social networks where information might not be as readily available. For example, Backhouse (2007) studies the correspondence between Cambridge Economists during early 20th century to understand how economists communicate and develop ideas with each other. With respect to information flow, Johansen (2017) focuses on learning how people diffused ideas about Western science and civilization with cheap secular books and periodicals in 1830s London. Moreover, if we can generate a dynamic (longitudinal) historical network, we can also learn how political, historical, social changes over time. For example, Medjedović (2021) proposed to use dynamic network analysis to conceptualize human life history pathways.

However, learning historical social networks comes with additional challenges. First, we can not directly access the individuals or the original network, thus verification requires external sources. Also, the study of social networks began in the late 19th century. As such, it is not common to see records of links or relationships prior to that time period. Therefore, in order to learn a historical social network, researchers usually rely on historical documents associated with individuals to build a network. One of the historical documents we can consider is the biography, written by scholars to include information about all major

phases and life events. Information in a biography can be very comprehensive but often without a fixed format. In addition, a person can have a biography only if his or her life has been well studied. Based on people's social identities, the amount of information in biographies for each person varies. Some people may have independent records for their titles, occupations, educations and family relationships while some others may only have a name and birth/ death year referred once. In this case, what if there exist some people that are ignored by historians for a long time but turn out to key connection points in the social network at that time? In other words, there may exist some less well-known people, who do not have their own biographies, but frequently appear in some other's biographies. Notice that those less well-known people in potential social connection positions of influence usually bring troubles to generate a comprehensive historical social network but reconstructing the social networks at that time is also a potential way to identify them.

Previous work on estimating social networks from text data spans different humanities and social science applications. First of all, some format of historical text data does indicates relations. For example, Marsden (1990) and Üsdiken and Pasadeos (1995) use structured surveys and citations respectively to estimate collaboration networks. If the data is unstructured, some people use pattern match and The China Biographical Database Project* is also a great example of that. This labor-intensive project involved manually listing all the possible expressions of human relations (e.g., "A is friends with B") and then searching the text using pattern matching to extract relational links. A more common way is to use the co-occurrence, which means if two people's name appear in the same part of the document, we assume that there is a link between them. For example, Almquist and Bagozzi (2019) uncover the underlying network structure of radical activist groups with British radical environmentalist texts from 1992 to 2003. Their work primarily concentrates on the application of topic models to analyze the text, and they infer a network using text co-occurrence counts. Such strategy is not only applied to the historical document analysis, but can also be be applied on fiction texts: Bonato et al. (2016) extract and analyze the social network from three best-selling novels, defining a link between two characters if their names co-appear within 15 words.

Even though the co-appearance is considered as the most popular and convenient signal for the social connection, Six Degree of Francis Bacon (SDFB) project † argues that co-appearance does not necessarily suggest a connection, especially when the research period is long enough, and there is a strict definition on the social connection, such as people have to yearly overlapped to know each other physically. It is possible that the co-appearance is caused by a common friends. For example, two people do not know each other, but they can be mentioned together in one of their common friends' biography. In that case, using co-appearance as the solo standard to create links between people may bring mistakes into the reconstructed social network thus lead to more errors in later historical analysis. In order to solve such problem, Warren et al. (2016) proposed to use a conditional independence structure to model the relationships and show their methods

*See `https://projects.iq.harvard.edu/cbdb/home`.
†See `http://www.sixdegreesoffrancisbacon.com/`

tends to increase the accuracy of the reconstructed social network. We will discuss their approach in the following section.

1.1 Six Degrees of Francis Bacon

SDFB estimated a social network in Britain in 1500–1700 (Warren et al., 2016). The project makes use of natural language processing (NLP) tools and statistical graph learning techniques to extract names and infer relations from biography data. Figure 1.1 is a snapshot from the SDFB website, which shows the social network between Francis Bacon and Robert Cecil, including two targeted people (large blue circles), common connections between the two target people (large red circles), common connections between one of the targeted people and one of the common connections between two targeted people (small red circles) and connections only connect to one of the targeted people (small blue circles). The grey lines are the social relations inferred from the statistical models, and the black lines are the ones later justified by the experts. The website has over 15000 people as the nodes and over 170000 relations as the links. The network is constructed based on text from the *Oxford Dictionary of National Biography* (ONDB). ODNB[‡] is the national document of people who are known in British history and culture, worldwide, from the Romans to the 21st century. The dictionary, constructed in 12 sections, is a set of biographies written by experts in each field. It was first published in 2004 but has been updated ever since. So far, the dictionary contains over 60,000 individuals biography and 536 Theme articles, which represent critical social groups and events across British history.The website also allows experts to manipulate the current nodes and links, including adding, deleting, and annotating any nodes and links. The website also includes 130 groups listed on the website right now, and most of them are extracted from the ODNB's list of Theme articles in the targeted period (1500-1700), which cover the groups, clubs, factions, and movements that are important during the period. Also, experts have manually added some groups about the occupations, like judges and booksellers.

For each biography owner in ODNB, we can extract their information about name, title, birth/death year, and historical identity from the website. The following is a short example of the biography text of Francis Bacon.

> **Bacon, Francis,** Viscount St Alban (1561–1626), lord chancellor, politician, and philosopher, was born on 22 January 1561 at York House in the Strand, London, the second of the two sons of **Sir Nicholas Bacon** (1510–1579), lord keeper, and his second wife, **Anne** (c.1528–1610) [see **Bacon, Anne**], daughter of **Sir Anthony Cooke**, tutor to **Edward VI**, and his wife, **Anne, née Fitzwilliam**. He was baptized in the local church of St Martin-in-the-Fields, but spent most of his childhood, together with his elder brother, **Anthony Bacon** (1558–1601), at Gorhambury, near St Albans, Hertfordshire, which their father had purchased in 1557.

Identifying links from such historical text data is challenging, since (1) people who are mentioned in the same part of a text may not necessarily know each other, it may only be due to the fact they both know

[‡]See `https://www.oxforddnb.com/`

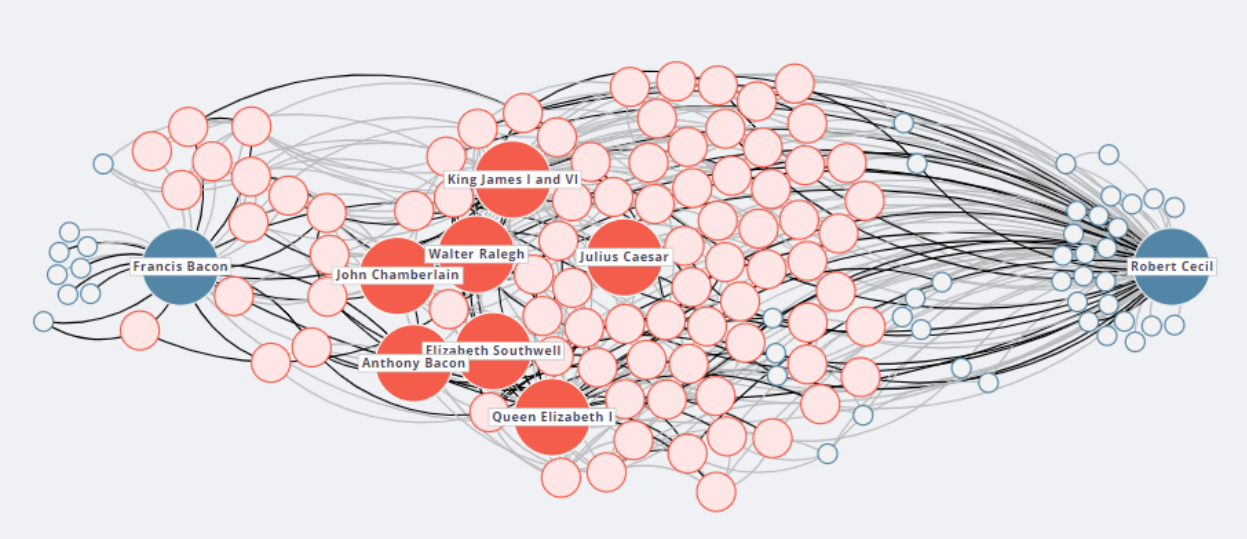

Figure 1.1: A snapshot of a part of social network of SDFB. Large blue circles: targeted people Francis Bacon and Robert Cecil; large red circles: first degree common connections; small red circles: second degree common connections; small blue circles: non-common connections; grey lines: connections inferred by statistical model; black lines: connections later justified by experts.

another person. For example, Edward VI is mentioned in this paragraph only because he is a student of Bacon's grandfather, but he has no direct connection with Francis Bacon; (2) the cases of people share the same name (duplicated) or people referred by only first or last name (partial) are ubiquitous in historical texts, and it is unclear how to assign their mentions to individuals. For example, there are two "Anne" in the text, so it is hard to decide which lady is referred to without having a correctly associated last name; and (3) there is a lot of information of various types in the text, like time, location, title, etc. The information usually does not have a specific format, thus is hard to extract and organize.

The first step of SDFB is to create a person-by-document matrix based on the text. The process is described in Figure 1.2. The people's names are extracted from the biographies during the targeted period (1500-1700) with a Name-Entity-Recognition (NER) tool (Finkel et al., 2005). The biographies are chopped into short documents that have no more than 500 words each, and if a document does not refer to any names from the target period (1500-1700), it will be removed. Among 12149 biographies in the research, 73.6% of them result in only one document, and 15.3% of them are chopped into two documents. The longest biography is chopped into 42 documents, which belongs to a Scottish politician and judge, Sir James Murray. Then, we can create a doc-by-person matrix such that an entry Y_{ij} in the matrix represents how many times a person j is mentioned in a document i.

There is also a lot of labor work to roughly clean and handle the duplicated and partial names afterward since the NER tool determines whether a word is a name or not, but fails to distinguish people with the same name. In all the SDFB data, approximate 13.3% of the names are duplicated. Also, the tool can not

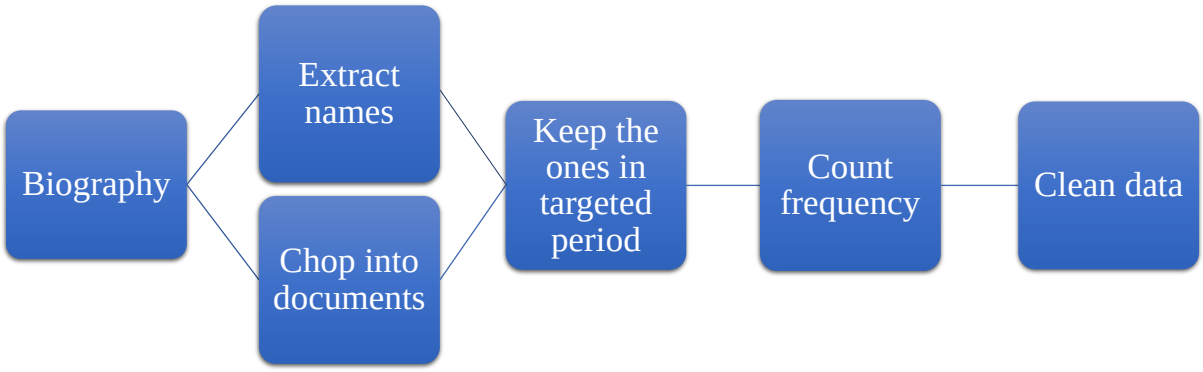

Figure 1.2: The data processing of SDFB. SDFB takes the biographies from the ODNB, then extracted names with NLP tools and chop each biographies into shorter documents to generate document-by-person metrics.

automatically distribute credit for partial matches either. For example, "Bacon, Francis" will be considered as two individual names "Bacon" and "Francis". In those cases, SDFB applied the following rules to distributed counts:

- If two people have exactly the same name and their life span (based on birth/death year and allow for one-year margin) overlapped and if both of them have a biography in ODNB, then the total counts of their name will be distributed based on the length of their biography with the percentages capped at a max/min of 75% and 25%.
- If there are more than two people who have precisely the same name and their life span (based on birth/death year and allow for one-year margin) overlapped, then the total counts of their name will be distributed evenly.
- For the partial names (only first name or only last name is mentioned), the count will be randomly allocated to the people with the same first or last name.

These assignments were done a 100 times, with the last point in the rules mentioned above leading to 100 doc-by-person matrices with different random assignments. An example of one doc-by-person matrix Y is the following Figure 1.3, where each row represents a document, each column represents a person, and each entry represents how many times this person is being mentioned in the corresponding documents. For example, in Figure 1.3 the circled entry represents in document n, King Charles I has been mentioned three times.

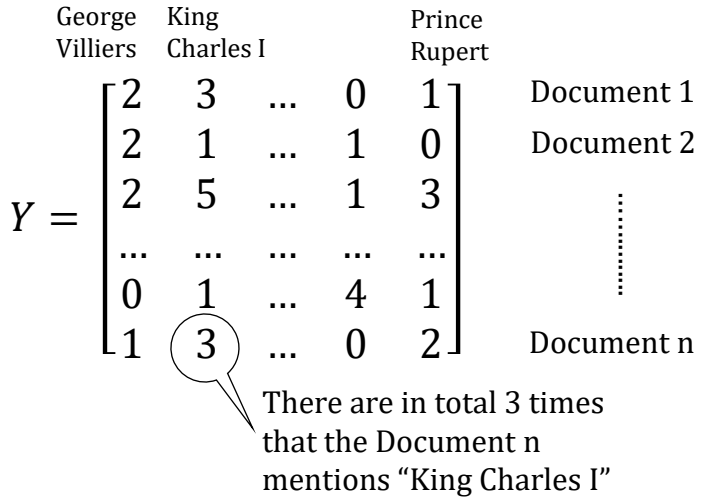

Figure 1.3: Example of SDFB data structure

When inferring the relations, all 100 doc-by-person matrices were considered. Together, these help to quantify the certainty of the relationship, defined as the number of times a link has been identified among the 100 matrices. Notice that certainty here does not represent the strength of a link. A high certainty link may be a weak link. On the SDFB website, only the links with certainty over 50 are depicted.

The relation inference of such a doc-by-person matrix is through statistical graphical models, using a conditional independent structure. The assumption for this choice is that if two people know each other, it is more likely that they show up in the same biography or the same part of the biography. However, even if two people are co-mentioned in the same paragraph, it does not necessarily mean that they know each other since they may be co-mentioned. They may be in the same paragraph only because they both know another (third) person. This implies that a conditional independence structure may be useful for representing social relations, and SDFB uses the Local Poisson Graphical Lasso Model (Allen and Liu, 2012). SDFB validated the Local Poisson Graphical Lasso Model in several ways and showed that it performs better than simply using co-appearance.

There is no doubt that the methodology of SDFB creates a pioneer pipeline of generating large social networks from the text data. However, there is also much room for improvement. In the validation of precision and recall among 12 non-random people, even though Warren et al. (2016) have shown that the methodology leads to high precision, we have observed that the links picked up by the SDFB model tend to relatively low recall. There are several reasons for the missing links. First, the inference only makes use of the co-mention counts but ignores all the semantic information. According to homophily theory, similar individuals are more likely to connect to each other than the dissimilar ones (McPherson et al., 2001). In the validation with topics models, Warren et al. (2016) have shown that the people within the same topics

are more likely to be linked. Besides of that, several studies also have shown that people who share similar age, education level (Kossinets and Watts, 2009), occupation (Calvo-Armengol and Jackson, 2004), gender and economic status (McPherson and Smith-Lovin, 1982) are more likely to be connected. Given these results and from the first validation, we can see that the people who know each other do share some common characteristics, so including the information like common last name and collective group identities should be able to help us pick more links than what SDFB has done. In a subsequent study, Mohamed (2020) used a logistic model to predict whether two people know each other using features of the estimated SDFB network (e.g., common links) and pairwise covariates (e.g., same gender or social group), and finds that at least one third of the false positive links (i.e., the links that the model predicts with a high probability but do not exist in the estimated SDFB network) have supporting historical sources. This indicates that using covariate information within the model may help us to improve estimation of historical networks.

Second, the randomization of partial names and duplicated names have weakened the signals. Even though the current approach with repeated evaluation through 100 doc-by-person matrix helps to correct the effect of randomness, it is much more time consuming. We are wondering whether Record linkage or other NLP techniques should be included in the name assignment process or later in the modeling process, like through adding a covariate to address the existence of duplicated names. This may help identify links more accurately. Last but not the least, if we know there exists some covariates that largely affect the probability whether two people know each other or not, for example, a covariate to indicate family relationships, it is reasonable to include them into the modelling process, since even though a friend's friend is not necessarily a friend, a family's family is still family.

In the Chapter 2 of the thesis, we first try to explore and understand SDFB's approach to infer relations through the local Poisson graphical lasso model. We compare the performance of local Poisson graphical Lasso model with traditional Gaussian Graphical model as well as the Poisson log-normal model. We will focus on both metric performance and evaluation efficiency.

In Chapter 3, we extend the local Poisson graphical lasso model in the context of the SDFB project to include covariates information. We will show how to implement the node covariates into the network model's penalty factors and describe two potential methods for estimating penalty factors for potentially a large number of covariates. The performance of the extension is represented by how the inclusion of additional information into penalty estimation can significantly improve precision and recall.

In Chapter 4, we extend the model to include continuous covariates. We will discuss several applications to include continuous covariates with record linkage, natural language processing (NLP) tools as well as other text information in the biographies, so that more text information can be included into the model to create a more comprehensive reconstructed early social network.

Chapter 2

Using Graphical Models to identify social links

SDFB used the Local Poisson Graphical Lasso Model to estimate the social network, but this is not the only model using a conditional independence structure. In this section, besides the Local Poisson Graphical Lasso Model, we also consider two other graphical models: the Gaussian Graphical Model and the Poisson Log-Normal Model. The Gaussian Graphical Model is the most well-studied model that interpreting the conditional independence structure, while the Poisson Log-Normal Model is designed for the conditional independence structure of discrete count data. Notice that we only consider undirected graphs, since we assume the acquaintance relation between two people to be mutual. Both of the models can be estimated globally, in contrast with the local estimation of the Local Poisson Graphical Lasso Model in SDFB. There also exist some other models designed to estimate a network based on count data, but they usually can be considered as the variations of the Poisson Graphical Model. In the following, we will discuss the definitions of the models, their estimation as well as whether each model fits our context. The final section compares the performance of three model using simulations, measured by area under the curves (AUC).

2.1 Local Poisson Graphical Lasso Model

2.1.1 Model definitions

The local Poisson graphical lasso model is the model that generates the current SDFB network, and is originated from the Poisson Graphical Model(Allen and Liu, 2012). The Poisson Graphical Model assumes that conditional on all the other variables, each variable is a Poisson random variable. Suppose now we only have one document, and Y_j is the count for the person j mentioned in this document and $Y_{\setminus j}$ is the counts

for all the other person excluding j, then the model can be expressed as:

$$P(Y_j|Y_{\setminus j} = y_{\setminus j}, \theta, \Theta) \sim \text{Poisson}(e^{\theta_j + \sum_{k \neq j} \Theta_{jk} y_k}). \tag{2.1}$$

What we need to estimate is the edge parameter matrix Θ. We assume $\Theta_{jk} = 0$ if and only if there is no connection between person j and person k.

2.1.2 Estimation

Based on the previous definition, the model is stated in a pairwise and local relation. If we want the equations to hold jointly and form a Poisson Markov Random Field, we will have the probability density as

$$P(Y|\theta, \Theta) = \exp\left\{ \sum_j (\theta_j Y_j - \log(Y_j!)) + \sum_{j,k} \Theta_{jk} Y_j Y_k - A(\theta, \Theta) \right\}. \tag{2.2}$$

Now if we try to estimate Θ globally, we will realize that since the value of Y_{ij} will be 0 to infinity, to make sure $A(\theta, \Theta)$ is finite and $P(Y|\theta, \Theta)$ is a probability between 0 to 1, all $\hat{\Theta}_{jk}$ need to be non-positive. However, in our context, the negative coefficient is not meaningful since a negative $\hat{\Theta}_{jk}$ suggests that the number of mentions for person j will decreases if the number of mentions for the person k increases, which does not show a social relation between j and k.

In order to allow $\hat{\Theta}$ to be positive, the SDFB project therefore chose to estimate the model locally, which leads to the Local Poisson Graphical Lasso Model. Thus, for each j, we fit a generalized linear model with the package `glmnet` and include an L1 penalty to enforce the sparsity (Friedman et al., 2010).

2.2 Gaussian Graphical Model

2.2.1 Model definitions

The Gaussian Graphical Model is the most popular model for which a graph represents the conditional independence structure between random variables (Lauritzen, 1996). Consider a multivariate Gaussian $X \sim N_d(\mu, \Sigma)$, with the density

$$f(x|\mu, \Sigma) = (2\pi)^{-d/2} (\det(\Omega))^{1/2} e^{-(x-\mu)^T \Omega (x-\mu)/2}, \tag{2.3}$$

where $\Omega = \Sigma^{-1}$ is called the precision matrix. We interpret $\Omega_{jk} = 0$ as person j and person k not being connected. Notice that the conditional distributions for each variable of the multivariate Gaussian will be univariate Gaussian and the quadratic form $x^T \Omega x$ can be written as $\sum_{ij} x_i, x_j \Omega_{ij}$, thus the density is proportional to a product of potential functions $\prod_{ij} \phi(x_i, x_j)$, which each of these potential functions depends

on only two coordinates. Therefore, it can satisfy the Markov conditional property, and thus the multivariate Gaussian distribution can be represented by a pairwise Gaussian Markov Random Field (GMRF). Unlike the Poisson Graphical Model, parameter estimates can range from $-\infty$ to ∞. Thus, the model can be estimated globally.

2.2.2 Estimation

We can estimate Ω using the `gLASSO` package in R, and the regularization parameter can be selected via cross-validation (Friedman et al., 2008). The estimation takes the empirical covariance matrix as the input and then estimates one row and one column synchronously at each time until convergence, so that it guarantees a global estimate and $\hat{\Omega}_{jk} = \hat{\Omega}_{kj}$ in the final $\hat{\Omega}$.

2.3 Poisson Log Normal Model

2.3.1 Model definitions

The Poisson Log-normal Model was first introduced in 1989 in a regression-based structure (Aitchison and Ho, 1989). Instead of estimating the Poisson Graphical Model directly, it maps the Poisson data to latent Gaussian data and then estimates the Gaussian Graphical Model. Let $Y = (Y_1, ..., Y_n)^T$ be n independent and identically distributed p-dimensional count observations, then a Gaussian $Z = (Z_1, ..., Z_n)^T$ is drawn and the coordinates of Y_i are sampled independently from a Poisson distribution conditionally on Z_i. The model is described as

$$\begin{cases} Y_{ij}|Z_{ij} \sim \text{Poisson}(\exp(Z_{ij})) \\ Z_i \sim N_p(\beta, \Sigma) \end{cases} \tag{2.4}$$

where β is a p-dimensional vector and Σ is $p \times p$. The goal is to estimate the edge parameter matrix $\Omega = \Sigma^{-1}$.

2.3.2 Estimation

The main obstacle for estimating the Poisson Log-normal Model is that the latent variable Z is unobserved. Therefore, it is hard to evaluate the log-likelihood of the observed data $\log p_\Omega(Y) = \log \int p_\Omega(Y, Z) dZ$. There are several ways to bypass this problem. Here we will describe three methods proposed in the last few years that we have considered. Choi et al. (2017) choose to use Laplace's method to approximate the likelihood and its gradients. The parameters are obtained by minimizing the objective function with Newton's method and ADMM (alternating direction method of multipliers) algorithm. The method is implemented in the R package `PLNet` but unfortunately, this package does not consider the case with sparse data, so it will have non-converge errors with our data. Also, it is much slower than the following two EM algorithm methods.

Another method is the EM algorithm, which requires to evaluate the conditional expectation of the complete log-likelihood, which involves the p-dimensional integral. Since, in our case, p is much larger than 4, there is no closed-form solution. Sinclair and Hooker (2019) propose to assume the initial Σ to be diagonal so that the computational burden is much smaller. Then a one-step EM algorithm is applied to the transformed data Z_i from Y_i and performed `gLASSO` model on Z_i to infer the network. This method originally also did not consider the sparsity of the data but with some minor adjustment by setting some infinite integral results to zero it can accommodate sparse data.

Chiquet et al. (2018)'s method adopts a similar but more comprehensive approach, which use variational inference . In this case, each conditional distribution $p_\Theta(Z_i|Y_i)$ is approximated by a set of multivariate Gaussian distribution Q with mean vector m_i and diagonal covariance matrix $\text{diag}(S_i^2)$. Let $\psi = (M, S)$ and $M = (m_1, ...m_p)$. Then the objective function $J_{Y;\psi,\theta}$ is defined as

$$J_{Y;\psi,\Theta} = \sum_{i=1}^{n} E[\log p_\Theta(Z_i|Y_i)] + E[\log p_\Theta(Z_i)] - E[\log q_\psi(Z_i)] \tag{2.5}$$

which is the sum of conditional expectation and the error of approximation, measured by the Kullback-Leibler divergence. The objective function is optimized through repeatedly solving a gradient ascent with box-constraint and Gaussian maximum likelihood until it converges. This method is implemented in the R package `PLNmodels` (Chiquet et al., 2018).

In practice, the variational inference beats the other two on computational complexity and can directly work with sparse data. Even though it is still a question of whether it can be applied to extensive data such as ODNB, it is enough for the simulation study to evaluate the model. Therefore, in the following section, we adopt the variational inference method to estimate the Poisson Log Normal Model.

2.4 Summary of the models

After examining all the choices, unfortunately we realize there is no perfect solution in our case. Each model has its own pros and cons which are summarized in the Table 2.1.

	Fit assumption	Fast estimate	Global estimate
LP	Yes	Yes	No
GGM	No	Yes	Yes
PLN	Yes	No	Yes

Table 2.1: Summary of the conditional independence models

The local estimation is relatively fast compared to other methods of modification on the Poisson Graphical Model. However, the estimation is local, so $\hat{\Theta}_{jk}$ is not necessarily equal to $\hat{\Theta}_{kj}$. Thus, it requires an "AND/OR" rule to determine whether j and k are connected. To be more specific, the "AND" rule suggests

that $\hat{\Theta}_{jk}$ and $\hat{\Theta}_{kj}$ both have to be positive for a link between person j and a person k to exists, while the "OR" rule suggests that if at least one of $\hat{\Theta}_{jk}$ and $\hat{\Theta}_{kj}$ is positive then we can say the link exists. SDFB adopts the "OR" rule. If we want to implement a global penalty, it needs to re-run the model multiple times until it converges, which is not practical with such a large dataset. Another option is to add an additional penalty to control the difference between $\hat{\Theta}_{jk}$ and $\hat{\Theta}_{kj}$. In that case, we need to train all people's penalties at the same time, and a reasonable initialization is required so that the computation complexity can be smaller. Therefore, one possible combination is to train the model locally first and then use the result from the local model as the initialization of global training. A small simulation shows that this idea works, and the result is similar to the OR rule SDFB has used.

The Gaussian Graphical Model is well defined. There are a lot of studies and applications about it, and `gLASSO` is a relatively fast way to perform the global estimation. The problem for Gaussian Graphical Model is that under our setting, the data are the number of mentions, which are non-negative integers. Thus the data distribution is probably not Gaussian. It could be Poisson or an other count-valued distribution, like the negative Binomial. Therefore, using Gaussian Graphical Model may violate the assumption of the model.

The Poisson Log-Normal is designed for count data. The estimation is global, which solves the major disadvantage of the Local Poisson Graphical Model. However, as a mixture model, it is hard to estimate the model's parameters. Even though the new algorithm improves the computational efficiency of Poisson Log-Normal, when the sample size is large, it may still not handle the high-dimensional data, like in the case of SDFB project with over 15000 people. Also, the transformation of the data in the variational inference method may introduce some errors when interpreting the dependence structure, since we have to balance the density approximation and optimizing the objective function at the same time.

2.5 Model comparison by simulation

2.5.1 Simulation methods

In order to compare the models quantitatively, we compare the models' performance on different networks, and try to understand how the network structures may affect the linking. First, we create a small community (network) as the ground truth for the simulation. The community is similar by design to the SDFB social network, and we assume to have similar available demographic information. When creating the network, we consider three covariates: last name, group membership and birth/death year overlap. Note that we assume that if the lifespan of two people does not overlap (i.e., it is impossible that they physically met each other), then they should not be linked, regardless of all other factors. Below is a description of the general network design:

1. We generate 50 families in the community with 30 different last names (i.e., people with the same last name can be from different families).

2. For each family, we randomly generate 5 to 12 people, each with a birth and death year between 1500 and 1600 and a life length varying from 5 to 70.

3. Within a family, among those people whose lifespan overlaps, 50% know each other.

4. There are three social groups, A, B and C. Each person is randomly assigned to one of the groups with probability 0.5, 0.25 and 0.25, respectively.

5. Among those people whose lifespan overlaps, we additionally create 100, 100 and 50 links within groups A, B and C, respectively.

6. At the end, we add 300 random links to the whole community.

This design yields 464 people and 1164 links. We will refer this network as the "social network". Figure 2.1 illustrates a subset of the community with ten families, 100 people, and 158 links.

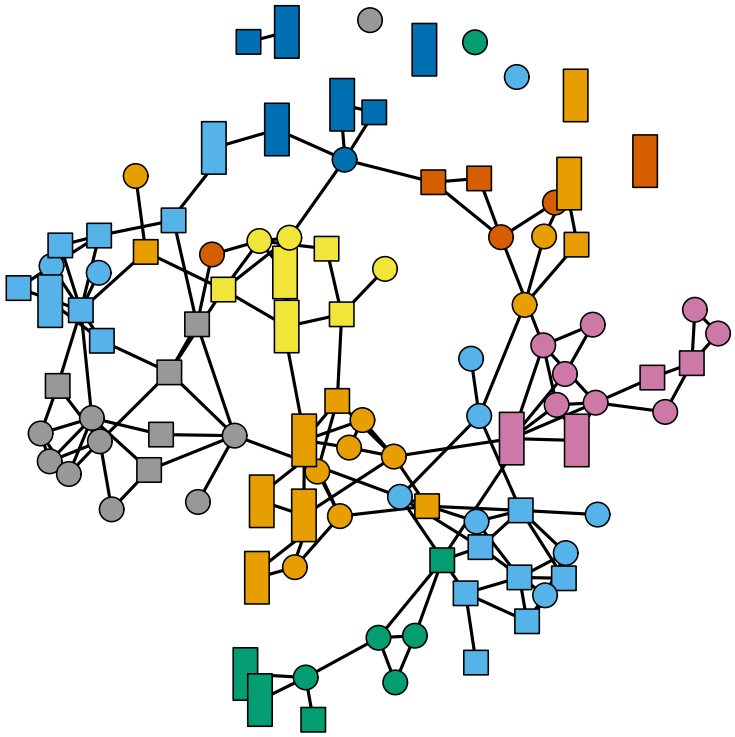

Figure 2.1: 10-family sub-community (100 people, 158 links), Last names are represented by colors and social groups by shapes. The network shows clear family structure with some social group structure and additional random links.

To contrast with the first network, we also generate a random network with the Erdös-Renyi model.This model is a uniform random graph model on the set of all graphs with a specified number of nodes and edges, and the number of nodes is set to 464 and the number of edges is set to 1164 to match what we have in the social network. We will refer this network as the "random network".

With each simulated network, we generate a document-by-count matrix. The simulation method is described in Allen and Liu (2012), which is based on Karlis (2003). The $n \times p$ simulated data X, with n independent observations with p nodes, is simulated from the model

$$X = YB + E. \tag{2.6}$$

E is the $n \times p$ matrix of noise and $E_{ij} \sim Poisson(\epsilon)$ for all i and j, where ϵ is the noise signal; Y is the $n \times (p + \frac{p(p-1)}{2})$ matrix of true signal $Y_{ij} \sim Poisson(\mu)$ for all i and j where μ is the true signal. B matrix is defined as $B = [I_{p*p}P \odot (\mathbf{1}_p tri(A)^T)]^T$ where P is the $p \times \frac{p(p-1)}{2}$ pair-wise permutation matrix, $\odot$ represents the Hadamard or element-wise product and $tri(A)$ denotes the vectorized upper triangular portion of the adjacency matrix A. Here is an example for the method. Suppose we need to generate n documents for 3 people j, k and h. Their network and corresponding adjacency matrix are in Figure 2.2:

Figure 2.2: Example of the network and corresponding adjacency matrix with three people. Suppose we have three people j, k and h, and person j is the common friend of the other two while person k and person h do not know each other.

For one document, for each pair j and k, we generate a count $Poiss_{jk}(\mu)$, and for each person j, we generate a noise $Poiss_j(\epsilon)$. Then the total count for each person in the document will be:

$$\begin{aligned}
\text{Person j} \quad & Poiss_{jk}(\mu) + Poiss_{jh}(\mu) + Poiss_j(\epsilon) \\
\text{Person k} \quad & Poiss_{jk}(\mu) + Poiss_k(\epsilon) \\
\text{Person h} \quad & Poiss_{jh}(\mu) + Poiss_h(\epsilon)
\end{aligned}$$

We will repeat this step until we have n documents. In the first simulation, we let $n = 2000$, which is a similar number of documents for 400 people as in the SDFB study. We then generate 10 different document-by-person matrices with this simulation framework and compare the precision and recall for each model. In

the second simulation, we also try to compare the computing time on local Poisson graphical model and Gaussian Graphical model when the n and p are increasing.

2.5.2 Simulation results

We first compare the results in the first two network when there is no extreme cases with the "popular" group. We compare through the AUC, which represents the overall ROC performance, and the precision and recall of the best model. The AUC calculation is self-implemented by calculating the 20 rectangles under the curves. The best model is selected as the one with highest sum of precision and recall, where precision and recall are defined as

$$\text{precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \qquad \text{recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}. \tag{2.7}$$

The result for the random network is given in Figure 2.3 and the result for social network is reported in Figure 2.4. It seems that the Local Poisson graphical model tends to perform better with metrics on precision and recall for both cases. If we value equally on precision and recall, local Possion tends to give us the better reconstruction of the network. However, notice that the difference between the Gaussian Graphical model and the local Poisson model is not large. In fact, in the social network, Gaussian graphical model tends to perform slightly better on the AUC.

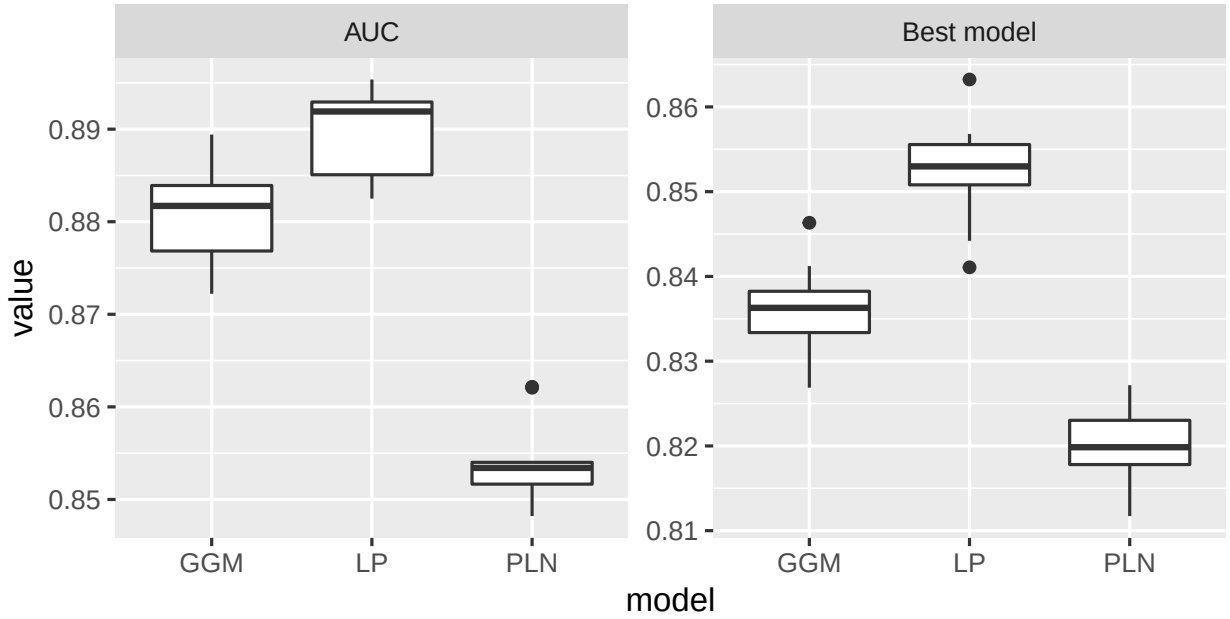

Figure 2.3: Left: The AUC for the GGM, the LP and the PLN on random network with high degree nodes over 10 runs. Right: The best average of precision and recall for the GGM, the LP and the PLN on the random network.

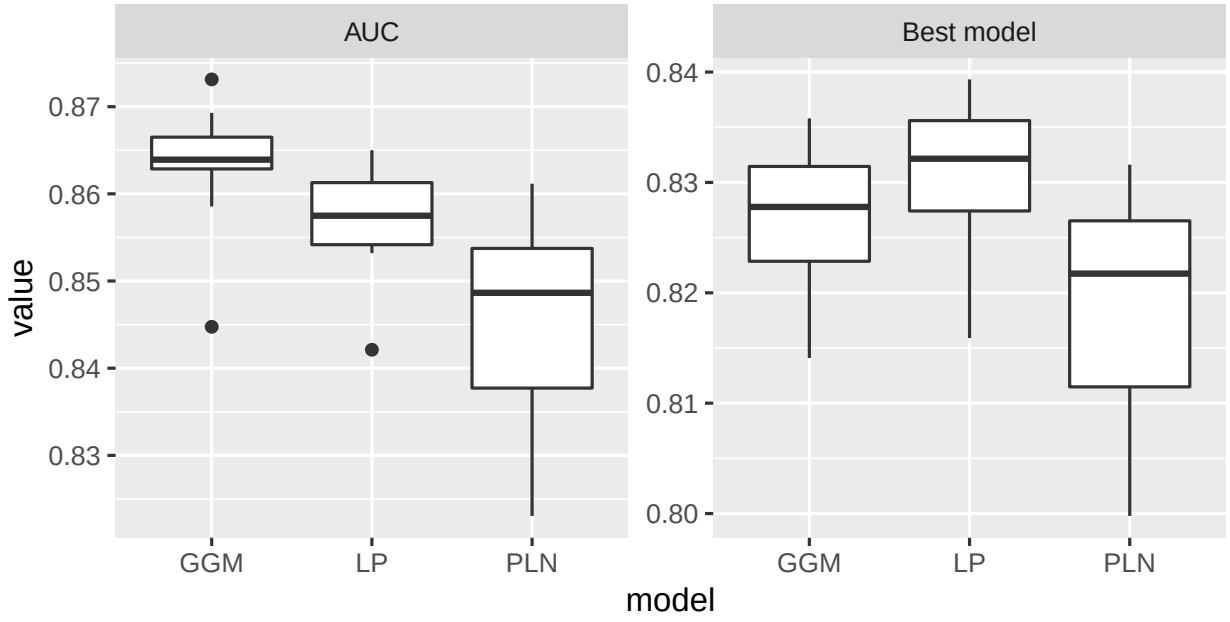

Figure 2.4: Left: The AUC for GGM and LP on social network over 10 runs; Right: The best average of precision and recall for GGM and LP on social network over 10 runs

Considering the fact that the current simulation scale is much smaller than actual SDFB data scale, we have also considered the computational complexity to understand how the data size will affect the model's performance. In the current simulation, the Poisson Log-normal is much more time consuming than the other two models thus makes it harder to be applied on larger data. Therefore we no longer consider it unless with a more efficient estimation method. The average time costs for each run is listed in Table 2.2.

Model	GGM	LP	PLN
Time (mins)	3.29 ± 0.12	7.77 ± 1.63	275.54 ± 36.83

Table 2.2: Average time cost for each run in the simulation.

2.5.3 Comparing the Gaussian and local Poisson graphical model

From the previous simulation, it seems that if we only want to select one estimated network when we care equally about precision and recall, we may choose the Local Poisson Graphical model. However, the AUC of the local Poisson graphical model is not always the best. In order to understand what features in the network lead to the difference in performance, we add another two networks to the simulation.

The third network is a scale-free network. When evaluating the degree distribution on the first two networks, we realize the ranges of the degree distribution are pretty similar. Even though the degree distribution for the social network is skewed left but there are no individuals with a super high degree.

When evaluating a similar size sub-network on SDFB, it seems that even within the sub-network, we may have some people with degree near 30 while in the first two simulated networks, the maximum degree is less than 15. Therefore, it seems that in the simulated social network, we are still lacking some people with a higher-than-average degree. Therefore, we generate a scale free network with linear preferential attachment which leads to a network with 464 nodes and 1386 edges. The degree distribution for the scale free network and the previous two networks are in Figure 2.5. We can see the third network tends to have a heavier tail on the degree distribution than the other two.

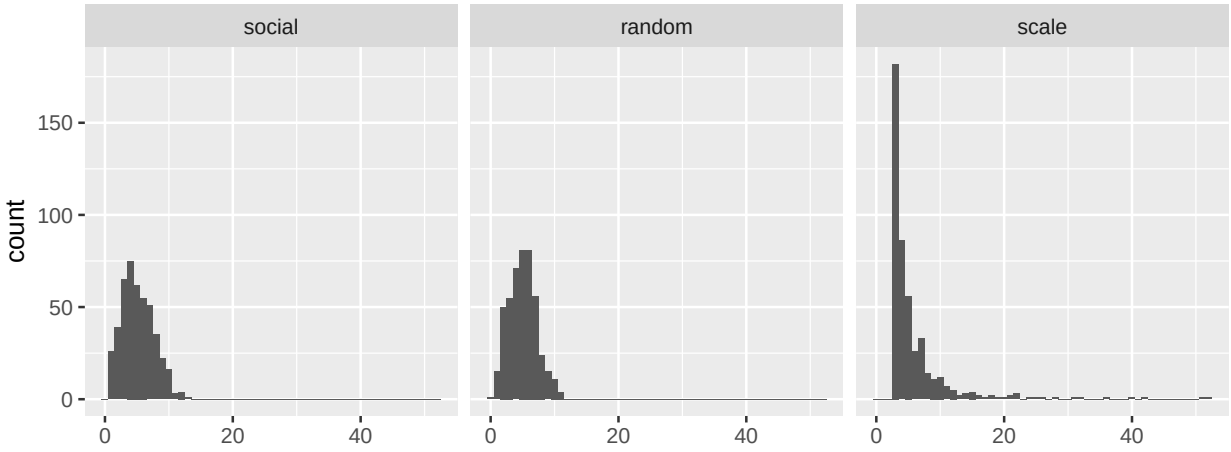

Figure 2.5: The degree distribution of the three networks in the simulation.

Another feature we have considered the clustering patterns in the network. Instead of having a random network with one major component, now we generate a network consists with four equal size components, which represents four clusters in the network. The size of the network is the same as previous random network, with 464 nodes and 1164 links.

The result for the scale-free network is given in Figure 2.6 and the result for random network with clusters is in Figure 2.7. For the two networks, the best average of precision and recall are always better for the local Poisson graphical model, which is consistent with the previous two simulations in subsection 2.5.2. When the network has some people with extremely high degree, like in the scale-free network, the local Poisson graphical model tends to perform better than the Gaussian Graphical model. However, if there exists clear cluster patterns, the performance of the two models in terms of AUC are similar.

In general, it seems that the existence of high degree nodes does affect the model performance, and if there is no clear cluster patterns in the network, the local Poisson model tends to perform better. Also, as a global model, the computational complexity of Gaussian Graphical model may be affected by the increasing size of data. Therefore, for the rest of the thesis, we will focus on and extend the local Poisson graphical model.

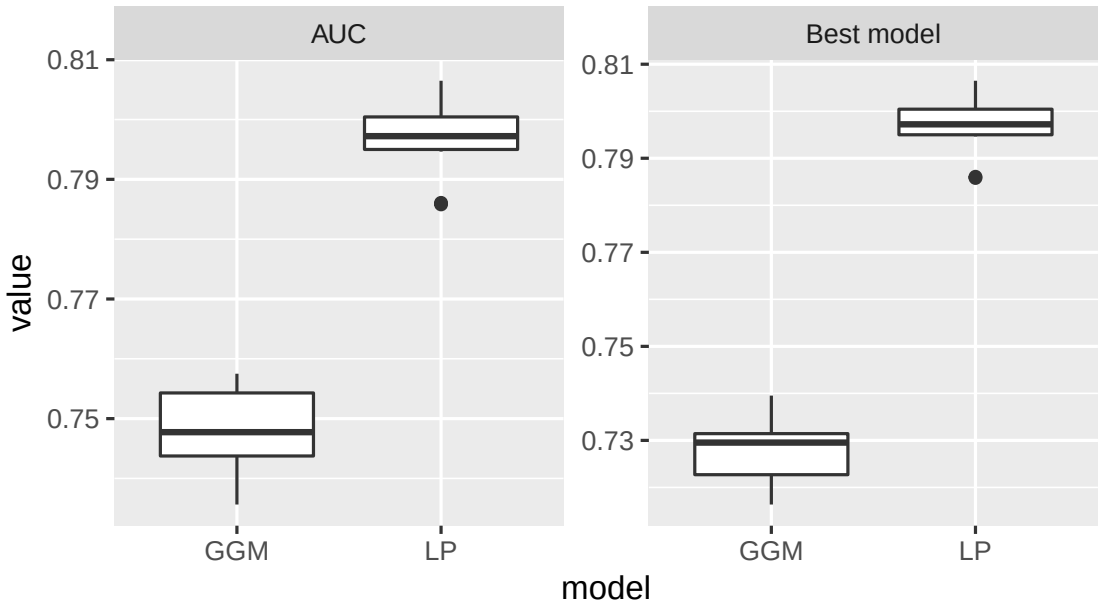

Figure 2.6: Left: The AUC for the GGM and the LP on scale-free network degree nodes over 10 runs; Right: The best average of precision and recall for the GGM and the LP on scale-free network degree nodes over 10 runs

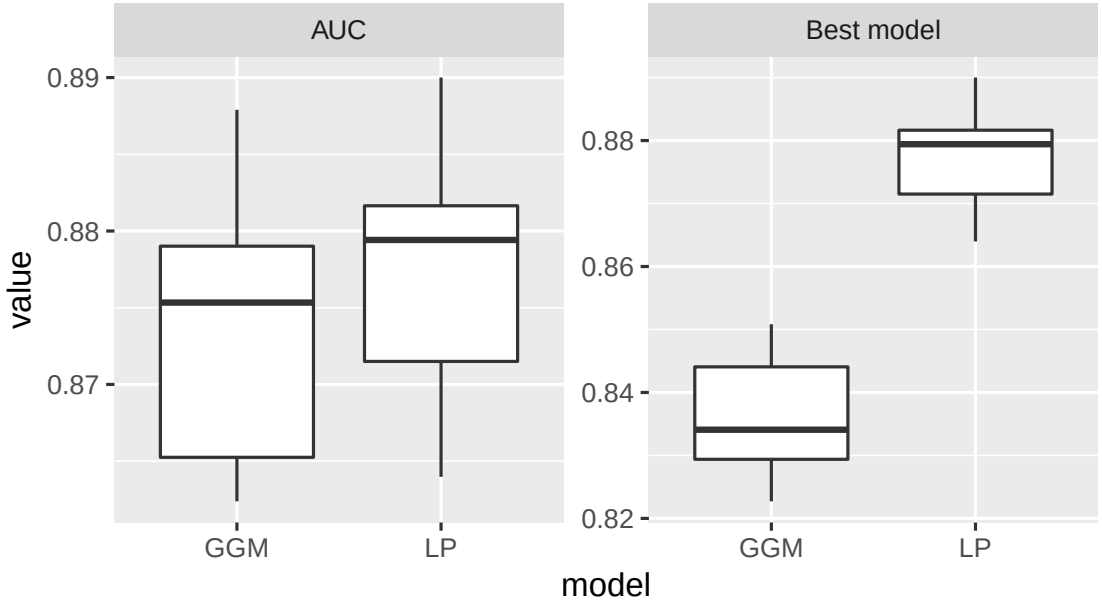

Figure 2.7: Left: The AUC for the GGM and the LP on random network with clusters over 10 runs; Right: The best average of precision and recall for the GGM and the LP random network with clusters over 10 runs.

Chapter 3

Covariate-dependent link penalization

There is no doubt that with the Local Poisson Graphical model, the SDFB project contributed a rich resource to support humanities research on early modern Britain. However, given this auspicious start, there is room for improvement. As we have stated in Chapter 1, homophily theory indicates that similar individuals are more likely to connect to each other than the dissimilar ones (McPherson et al., 2001). Even though with free-format text data like in biographies, in general it is hard to extract covariate information, the ODNB does provide us with a list of covariates for each person with a biography listed. The covariates do not represent all the information in the biography but we can start from there to think about how to incorporate that information into the model. In Chapter 4, we will discuss how to generate covariates directly from the text.

The idea to incorporating additional information into the Lasso regression model is not new. Yuan and Lin (2006) proposed group Lasso to add penalties to groups rather than individuals. Li et al. (2015) extended this method to a multivariate sparse group Lasso to incorporate arbitrary and group structures in the data. Their model provided a unique penalty for each node but also include a penalty for each group where the groups could overlap and even be nested. Zou (2006) proposed the adaptive Lasso which uses initial coefficient estimates without regularization to inform starting penalty weights. However, these papers did not include approaches for incorporating additional information into the penalties outside of group structure.

On the other hand, Boulesteix et al. (2017) proposed IPF-Lasso which assigns different penalty factors to all independent variables in their model that are a function of external information, and use cross validation to select penalty parameters based on model performance. In a similar vein, Zeng et al. (2020) use the Bayesian interpretation of the penalized regression, re-formulating Lasso regression as a Bayesian model. However, these approaches have not been implemented in the Poisson case, which involves different estimation challenges.

In this chapter, we first explore and extend both Boulesteix et al. and Zeng et al.'s approaches to the Local Poisson Graphical Lasso model in the context of the SDFB project. We will (1) show how to incorporate the node-wise covariates and information into the network model's penalty factors, (2) identify two potential methods for estimating penalty factors for potentially a large number of covariates (greedy approach and Bayesian approach), and (3) show how the inclusion of additional information into penalty estimation can significantly improve precision and recall, and may even help us to understand how the covariates affect the linking probabilities.

Next, we will modify Zeng et al.'s approach with a Laplace approximation which will decrease the computational complexity. This approach is promising in case one wants to fit a large data set. We will compare the Laplace approach with the double approximation method on model performance and computational complexity. Finally, we consider the effects of coefficient constraints in the model. In SDFB, all the links with negative coefficients are removed post-modeling since only positive coefficients indicates two people are mentioned together. We explore how the results change if we include this process into the model itself.

3.1 Including covariate information with penalty factors

We need a model to learn networks from text data that incorporates more information from the text than just the co-mentions. When adding additional covariates to our network model, there are several factors we need to consider. First, the model should be flexible enough to include the large number and variety of covariates available in historical text data. Even though the information provided in the SDFB data only includes people's name, birth/death year and a sentence to address people's historical significance, we expect there may be other text data coming in, like people's occupation and education. Among the available information, having a common last name is a strong indication that people belong to the same family, and thus should be useful in identifying the links. Birth/death year is also important since we are only interested in the physical social relation, like whether two people knowing each other or not. Historical significance, even though it is currently given in free format text (e.g. occupation and social identity) which also helps us to identify relationships. Each of these variables should be treated as at least one covariate in the model, and sharing both last name and occupation likely has a different impact on a possible relationship than just sharing last name. Moreover, covariate comparisons should not be restricted to just binary, e.g. match vs non-match, since the numerical information on distance and similarity may also be important to affect the linking probability.

Here we extend the Local Poisson Graphical Lasso model with a multiplicative factor for the penalty term that depends on individual covariate information inferred from the text. We start with the case of

binary covariates. For person j, we define the covariate matrix $Z^j \in \{0,1\}^{p\times m}$, with m covariates, by

$$Z^j_{kh} = \begin{cases} 1 & \text{if person } j, \text{ person } k \text{ have an equal value for covariate } h, \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

We here consider binary-valued matrices Z^j, but the approach proposed in this paper is also applicable to real-valued covariates. Examples of these include last name similarity and last name or social group commonality scores. In this case, vector Z^j would no longer be binary, but would also contain continuous-valued similarity scores. For example, if we want to account for misspelling when we are comparing the last names, instead of considering whether person j and person k have exactly the same last name, we can use their last name similarity, e.g., the Jaro-Winkler similarity the last names of person j and person k (Winkler, 1990). See Chapter 4 for a more detailed discussion of continous covariates.

For each covariate we include a different penalty factor. Thus, for each person j, the penalized Lasso estimators are given by

$$\begin{aligned} \hat{\Theta}_j = \arg\max_{\Theta_j} \sum_{i=1}^{n} \left[y_{ij}(y_{i,\neq j}\Theta_{\neq j,j}) - e^{y_{i,\neq j}\Theta_{\neq j,j}} \right] \\ - \sum_{k\neq j} |\rho_{kj}\Theta_{kj}| \quad \text{with } \log(\rho_{\neq j,j}) = Z^{j*}\alpha, \end{aligned} \tag{3.2}$$

where Z^{j*} is the matrix Z^j with the jth row taken out and prefixed by an all-one column vector and $\alpha \in \mathbb{R}^{m+1}$ denotes the penalty factor. The first element of α is α_0, an intercept controlling the overall shrinkage. If two individuals k and j share a common value on a covariate h, the penalty for parameter Θ_{jk}, indicating the link between them, is e^{α_h} times the overall penalty. Therefore, if having covariate h in common makes two people more likely to be connected, then α_h will be negative. Otherwise, it will be positive.

To illustrate this setup, suppose we have two covariates – last name and occupation – and consider the model for the name mentions Y_{ij} of Francis Bacon (person j) in document i. The $p \times 2$ covariate matrix Z^j indicates for the p individuals in the data whether they share their last name and occupation with Francis Bacon. An example of this matrix is shown in Table 3.1.

Matrix Z^{j*} equals matrix Z^j, but with the row of Francis Bacon taken out and prefixed by an all-one column vector. The penalty factor in this case is given by $\alpha = (\alpha_0, \alpha_{\text{LN}}, \alpha_{\text{OC}})$, where α_0 is the penalty intercept and α_{LN} and α_{OC} are the penalty factors corresponding to sharing a last name and sharing an occupation, respectively. Their effects on the penalty for parameter Θ_{jk} are given in Table 3.2. Notice that the signs of the penalty factors indicate how the covariates affect the linking probability. If the sign

is negative, then sharing the covariates lead to a decrease on the penalty thus making two people are more likely to be connected. The scale of the penalty factors indicate the scale of the effects.

Birth and death date are covariates that deserve special treatment in this framework, since if two individuals were not alive at the same time, they could not have had a social connection, that is the two people cannot have known each other personally. To address this, Warren et al. (2016) removed the links between people who were not alive at the same time post-network estimation. Given our penalty factor structure, we can instead include birth and death year information directly into the model. We set the penalty factor for the lifespan overlap covariate to infinity and so do not link people with non-overlapping birth and death years. Including infinite penalties into the model serves the same purpose as post-modeling removal of 'impossible' links, but will largely decrease the computational complexity since it deceases the dimension of the predictors during the estimation.

Once we know the values of the penalty factors, we can solve the optimization problem in Equation 3.2. For each person j, we fit a Poisson regression model including an L1 penalty to enforce sparsity. We estimate model parameters via penalized maximum likelihood using cyclical coordinate descent, as implemented in the R package `glmnet` (Friedman et al., 2010). This method consecutively optimizes the objective function given as part of expression (3.2) over each parameter while keeping the others fixed, and cycles until convergence.

After estimating the edge parameters Θ_{jk}, we only interpret positive estimates as an indication of the existence of a link, as proposed by Warren et al. (2016). A negative Θ_{jk} would imply that if a document mentions person j more it would mention person j less:this is not indicative of a relationship between persons j and k. Also, note that both Θ_{jk} and Θ_{kj} reflect the relation between persons j and k. Here, we adopt the "OR" rule, meaning that after estimating the edge parameter vectors for persons j and k, we say that there is a social tie between j and k when at least one of $\hat{\Theta}_{jk}$ and $\hat{\Theta}_{kj}$ is positive. The "AND" rule would require both $\hat{\Theta}_{jk}$ and $\hat{\Theta}_{kj}$ to be positive to claim a social tie, likely resulting in higher specificity, but lower recall. The "OR" rule can avoid missing links due to collinearity. For example, if two people's mentions are highly correlated, like a couple sharing highly similar social relations, when their mentions are both predictors in the model, it is possible that the Lasso model will only choose one of their coefficients to be positive while

	Same last name as Francis Bacon	Same occupation as Francis Bacon
Nicolas Bacon	1	1
Anne Bacon	1	0
Francis Bacon	1	1
Walter Raleigh	0	1
Queen Elizabeth I	0	0
⋮	⋮	⋮

Table 3.1: Example excerpt of covariate matrix Z^j, when j refers to Francis Bacon.

Person j and k...	Penalty ρ_{jk}
share no common covariate	e^{α_0}
only have the same last name	$e^{\alpha_0+\alpha_{\text{LN}}}$
only have the same occupation	$e^{\alpha_0+\alpha_{\text{OC}}}$
share both last name and occupation	$e^{\alpha_0+\alpha_{\text{LN}}+\alpha_{\text{OC}}}$

Table 3.2: Lasso penalty parameters ρ_{kj} as in (3.2) for parameters Θ_{jk} in a model with penalty factors depending on last name and occupation.

forcing the other one to be zero. Using the"OR" rule can help identify the missing links since each link has two chances to be picked from both directions.

Therefore, estimating the value of penalty vector α is essential for determining the edge parameters. In the following two sections, we discuss two approaches to estimate the penalty vector:using greedy search (Section 3.1.1) and using the reformulation of Lasso regression in the Bayesian framework (Section 3.1.2).

3.1.1 Greedy approach

One way to estimate the penalty factor α is by defining a grid of penalty parameter values and evaluating the corresponding models, selecting the values that minimize the prediction error, the mean square error (Boulesteix et al., 2017). It is also possible to use other metrics like AIC or BIC. However, this approach is generally computationally feasible only when the number of covariates is small (say, no more than four). Greedily searching the parameter space allows for inclusion of more covariates. Our proposed greedy algorithm for α is described in the following; the pseudo code is give in Appendix A1. Starting with all $\alpha_h = 0$, i.e. no penalty adjustment, the algorithm first iterates over all covariates in random order. For each covariate h and a gridded range of pre-specified α_h values, we use cross-validation to choose the baseline $\hat{\alpha}_0$ (holding all other α_h penalty parameters fixed) and calculate the corresponding MSE. We then choose the $\hat{\alpha}_h$ corresponds to the lowest MSE. After randomly iterating through all covariates, the algorithm repeatedly randomly iterates through all covariates again, looking for possible updated $\hat{\alpha}_h$ values, stopping when no further α_h tuning leads to a decrease in MSE.

To use this algorithm, we need to specify the search range for α_h and the step size d, which is the minimum incremental value for each step. We recommend starting with a search range for α_h such as $[-1.2, 0.5]$ (values outside that range have diminishing impact on the multiplicative factor value) and a relatively large step size d (e.g., 0.1). The search range can be enlarged if the margins are hit during the initial estimation. Decreasing the step size d can of course lead to a more fine-grained solution but will depend on any present computational constraints. We could also choose the search range $[a_n, b_n]$ using prior information, such as which covariates are expected to be influential and approximately how they might affect the chance of two individuals to be connected. For example, if we know a covariate h is likely associated with an increased

chance of a link, we could initially limit the search range of α_h to the negative numbers. This type of search range adaptation can also decrease the number of algorithm iterations and computational time.

3.1.2 Bayesian approach with Laplace prior

Lasso estimates can equivalently be derived as the Bayesian posterior modes under independent Laplace priors for the parameters to which shrinkage is applied (Tibshirani, 1996). Therefore, we can use the Bayesian framework to estimate the penalty parameters α. To this end, we complement model at Equation (3.2) with a Laplace prior for the edge parameters:for $k \neq j$,

$$\Theta_{kj} \sim \text{Laplace}\left(0, b_{kj}\right), \qquad b_{kj} \propto \rho_{kj}, \qquad \log(\rho_{\neq j,j}) = Z^{j*}\alpha. \tag{3.3}$$

Notice that α only influences the penalties on the edges and not the node parameters θ_j. We will specify the exact form of b_{kj} later in this section. We here extend work by Zeng et al. (2020) on incorporating covariate-dependent penalty factors in the Lasso term in linear regression and linear discriminant analysis (LDA) models to the Local Poisson Graphical Lasso model.

We use an empirical Bayesian approach to estimate the penalty parameters α. First, for each person j, we approximate the marginal log-likelihood of α, denoted by $l_j(\alpha)$, marginalizing over the coefficients $\Theta_{\neq j,j}$. The estimate of α is given by

$$\hat{\alpha} = \arg\max_{\alpha} \sum_{j=1}^{p} l_j(\alpha). \tag{3.4}$$

Note that we maximize the sum of the marginal distributions, because we need a global penalty factor over all people instead of for one specific person j. Since the $l_j(\alpha)$ are not convex, we use a Majorization Minimization procedure (Zeng et al., 2020) to estimate $\hat{\alpha}$. We then use the $\hat{\alpha}$ as input for the penalized maximum likelihood estimation of the model, as summarized at the Equation 3.2.

Since the Poisson regression likelihood and the Laplace prior are not conjugate pairs, there is no closed form expression for the marginal likelihood of α. We here present a general outline of how we approximated $l_j(\alpha)$, approximating both the Poisson regression likelihood and the Laplace prior – see Appendix A.2 for the full derivation.

First, we apply the log-gamma transformation to approximate the Poisson regression likelihood by a multivariate Gaussian distribution (Chan and Vasconcelos, 2009). In order to avoid log(0) in our derivation, we add 1 to all the observed outcomes y_{ij}, that is, define $y^*_{ij} = y_{ij} + 1$.

Second, we assume Θ_{kj} follows the Laplace prior $\Theta_{kj} \sim \text{Laplace}(0, \frac{\rho_{kj}}{2\hat{\sigma}_j^2})$, where $\hat{\sigma}_j^2 = \sum_{i=1}^{n} \frac{1}{y^*_{ij}}$ is the estimated variance in Gaussian distribution approximating the Poisson likelihood. We approximate this

prior by a normal distribution with the same variance (Zeng et al., 2020), yielding

$$\Theta_{\neq j,j} \sim \mathcal{N}(0, V^j), \tag{3.5}$$

where $V^j \in \mathbb{R}^{(p-1)\times(p-1)}$ is a diagonal matrix with $V^j_{kk} = 2\sigma_j^2 \exp^{-2Z_k^{j*}\alpha}$, in which Z_k^{j*} is the kth row of the covariate matrix Z^{j*}. Combining the two, we can approximate the log-likelihood of α for person j and find

$$-l_j(\alpha) \propto \log|C_\alpha| + \log(y_j^*)^\top C_\alpha^{-1} \log(y_j^*), \tag{3.6}$$

where $C_\alpha = \sigma_j I^2 + y_{\neq j} V^j y_{\neq j}^\top$, $y_{\neq j}$ denotes data matrix y excluding the jth column, and $\log(\cdot)$ is applied element-wise to $y_j^* = (y_{1j}^*, \ldots, y_{nj}^*)^\top$. Integrating this in expression (3.4), we can estimate the penalty factors.

3.2 Application of the greedy and Bayesian approaches

3.2.1 Simulation

We use the same 10 simulated data sets corresponding to the simulated social network as discussed in Section 2.5.1. We first apply both Greedy and Bayesian approaches to estimate the penalty factor α and then use the α to reconstruct the network. We anticipate all α should be negative, leading to a smaller penalty if two people share the same last name or are in the same group.

Figure 3.1 shows the distributions of $|\hat{\alpha}|$ for the greedy and Bayesian approach. All $\hat{\alpha}$ are negative, indicating if two people have the same last name or social group membership, they are more likely to be linked. (In general, we plot the magnitudes to allow for easier comparison, particularly when we have both positive and negative $\hat{\alpha}_h$.) The larger the absolute value of $|\hat{\alpha}_h|$, the stronger the covariate effect is on the penalty. Here we see that the Bayesian approach gives more similar $\hat{\alpha}$ values across the ten runs, correctly identifying last name as the most important covariate and group B as having a slightly stronger effect than the other two groups. The greedy method gives $\hat{\alpha}$ values that are more varied and do not reflect the network design. For example, although last name has a non-zero effect, it is not substantially larger than the other $\hat{\alpha}_h$ for the social groups. One potential reason for the consistency differences is that the Bayesian approach is trying to optimize the log likelihood of α while the greedy algorithm tries to optimize the model performance based on the MSE which may find multiple combinations of $|\hat{\alpha}_h|$ that lead to similar results. For example, if two people share the same last name and the same social group, a smaller penalty on either last name or social group or both can help with recovering the link.

For each generated document by person matrix, we also use the $\hat{\alpha}_h$ for both the greedy and Bayesian approach to estimate the network and calculate the corresponding precision and recall. We compare these

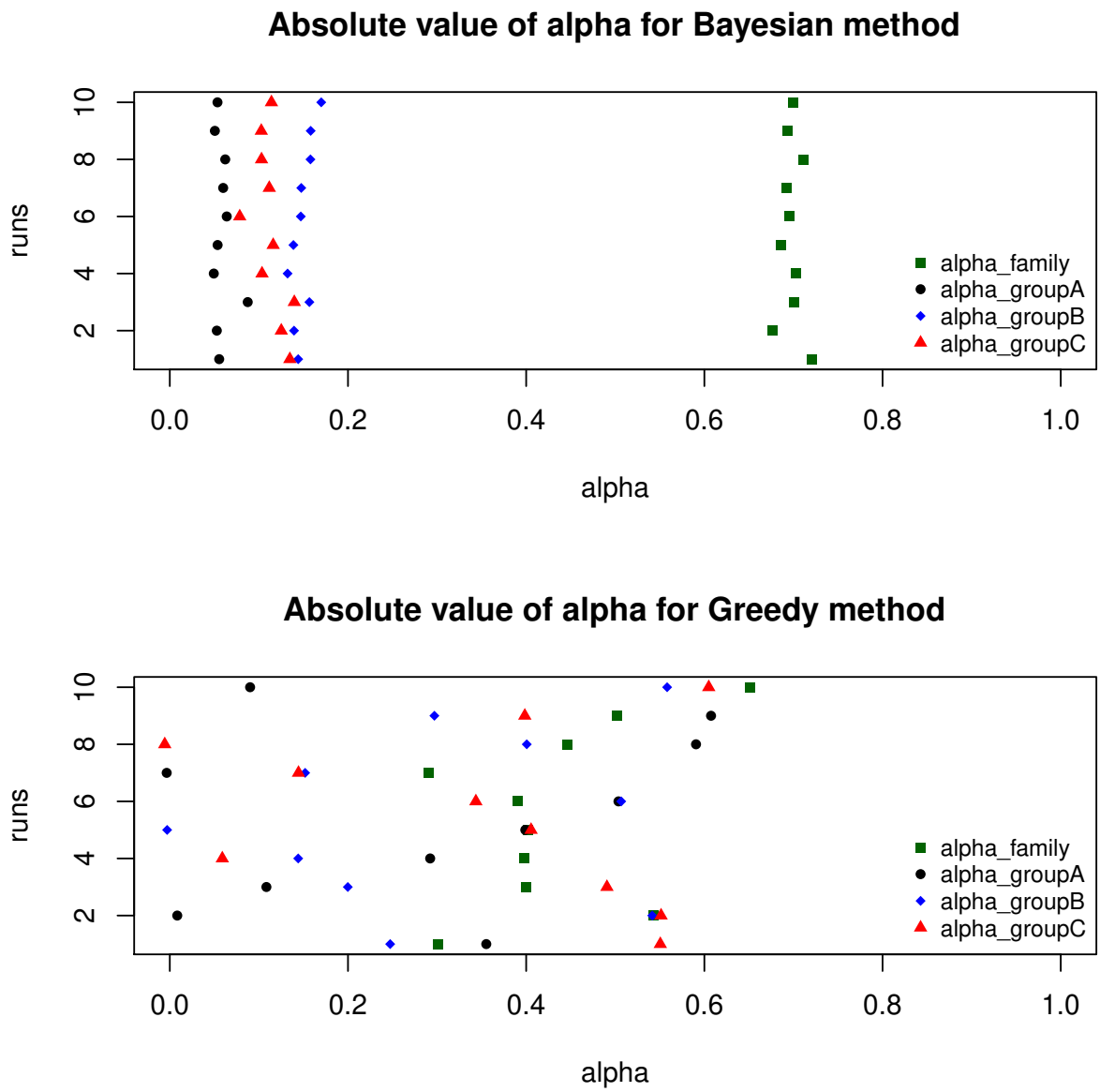

Figure 3.1: The distribution of $|\hat{\alpha}_h|$ estimated by the greedy and Bayesian approaches over ten runs. Both approaches generally pick the last name as the most important covariate. The Bayesian approach tends to give more consistent values while the greedy approach estimates have a larger variance.

values to those for the model without penalty adjustment. Figure 3.2 shows the resulting distributions for precision, recall, and the average of the two. We see that the model with penalty adjustment has improved precision, regardless of estimation approach, while the recall for all three options remains similar. The slight improvements in the average of the two follow.

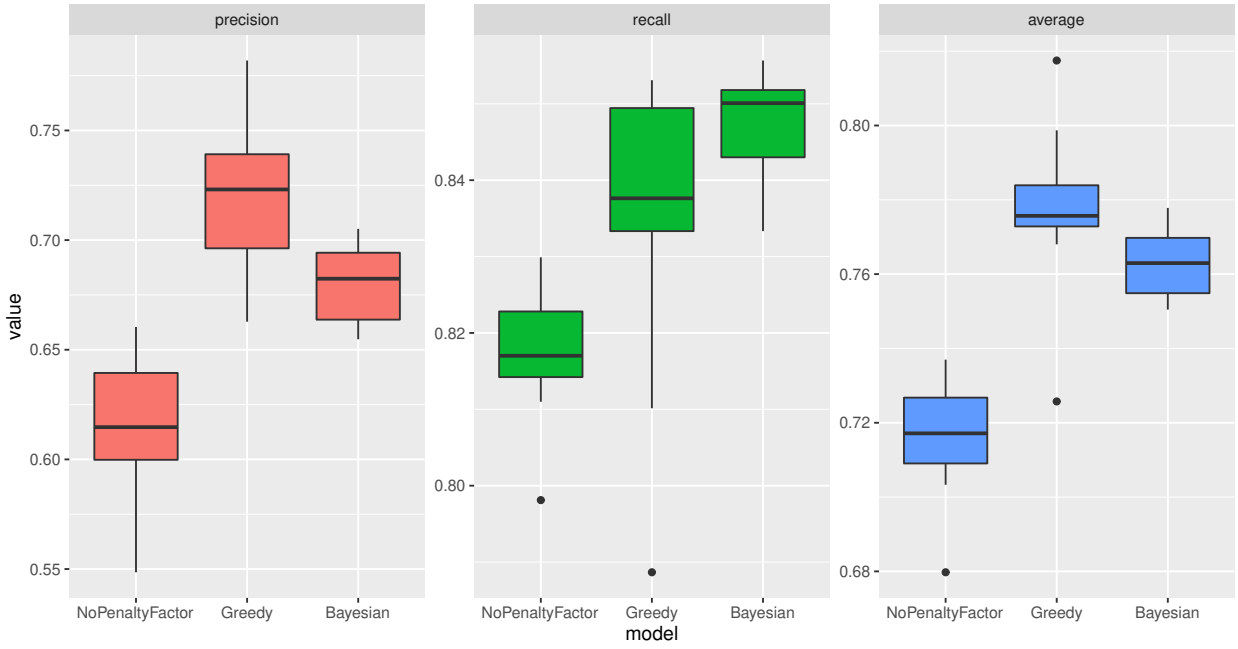

Figure 3.2: The distribution of precision and recall for the model without penalty factor, with penalty factor estimated using the greedy approach, and with the penalty factor estimated using the Bayesian approach. With penalty adjustment, both estimation approaches show an improvement in precision without a substantial change in recall. There is no significant difference on the average of precision and recall between the two estimation approaches.

Now examining the predicted network structure, we see that all three model/estimation approaches overestimated the true number of links in the original simulated network (1164). On average across the ten document by person matrices, the model without penalty adjustment detects 1662.1 links. With penalty adjustment, the greedy estimation approach averages 1434.7 links, and the Bayesian approach averages 1358.1 links, both an improvement over the original model.

We then take a closer look at the estimated network structure for our ten family, 100 people sub-community (Figure 2.1) for two of the simulated document by person matrices. For Run 10 (top row of Figure 3.1), the greedy and Bayesian estimation approaches give similar $|\hat{\alpha}_h|$ for last name, but the greedy approach gives slightly larger $|\hat{\alpha}_h|$ values (in magnitude) for the social group covariates. Therefore, we expect more links between people with the same group membership when using the estimates from the greedy approach compared to those of the Bayesian approach.

The relevant estimated networks for Run 10 can be seen in Figure 3.3 and corresponding number of links are in Table 3.3. We can see that for the network estimated by the model without a penalty adjustment, the false positive links exists across the whole network but with for the network estimated with penalty adjustment, both the number of false positive and false negative links decreases. The predicted networks with $\hat{\alpha}$ from the greedy and Bayesian approach are similar, but there are slightly more false positive links

cross group A and B for this sub-community with $\hat{\alpha}$ from the Bayesian approach and more false positive links within group C for the greedy approach. Note that this is in line with the observation that the absolute values of group-related penalties factors, for Run 10, are much larger for the greedy approach for group B and C. Thus greedy approach tends to pick up more within-group links while Bayesian may picks more cross-groups links.

We also examine Run 1 where the $|\hat{\alpha}|$ are quite different between the two estimation approaches (see Figure 3.1). The greedy method tends to give a much smaller penalty change for last name but a larger penalty change on social groups A and C, although we do note that the $|\hat{\alpha}_h|$ for group B is incorrectly estimated to be smaller than those for groups A, C. The corresponding predicted networks are depicted in Figure 3.4 and the corresponding number of links are in Table 3.4. The networks corresponding to the penalties estimated by the greedy and Bayesian approach are more dissimilar for for Run 1 than for Run 10, like the penalties themselves. Compared to the Bayesian method, the absolute value of group A penalty factors are larger for the greedy approach, leading to the detection of more links between people within group A in this subset.

In summary, our simulation study gives some evidence that including covariate information through penalty adjustment can improve the performance of Local Poisson Graphical Lasso model in the context of estimating social networks from co-mention/count data derived from text. With respect to differences in the two estimation approaches, we see that the Bayesian approach tends to give more consistent results; however, we note that, given its global estimation and computational tasks (e.g. matrix inverse calculations), it will be slower than the greedy algorithm.

3.2.2 Six Degrees of Francis Bacon:1500-1575

We illustrate the model proposed in this Chapter by an application to part of the data used in the SDFB project (Warren et al., 2016), focusing on the period between 1500 and 1575. We compare the results of the models with and without covariate-dependent penalty factors. We consider the interpretability of the penalty factors, how they affect which network links are estimated, and approximate the precision of the models with and without penalty factors using Wikipedia as a reference.

We first extract all documents from the SDFB database that contain references to individuals who were born and passed away between 1500 and 1575. This results in 2003 documents on 420 people. Over 83% of

	True Positive	False Positive	False Negative
No penalty factor	129	20	29
Greedy	137	12	21
Bayesian	139	12	19

Table 3.3: The number of true positive, false positive, and false negative links for Run 10.

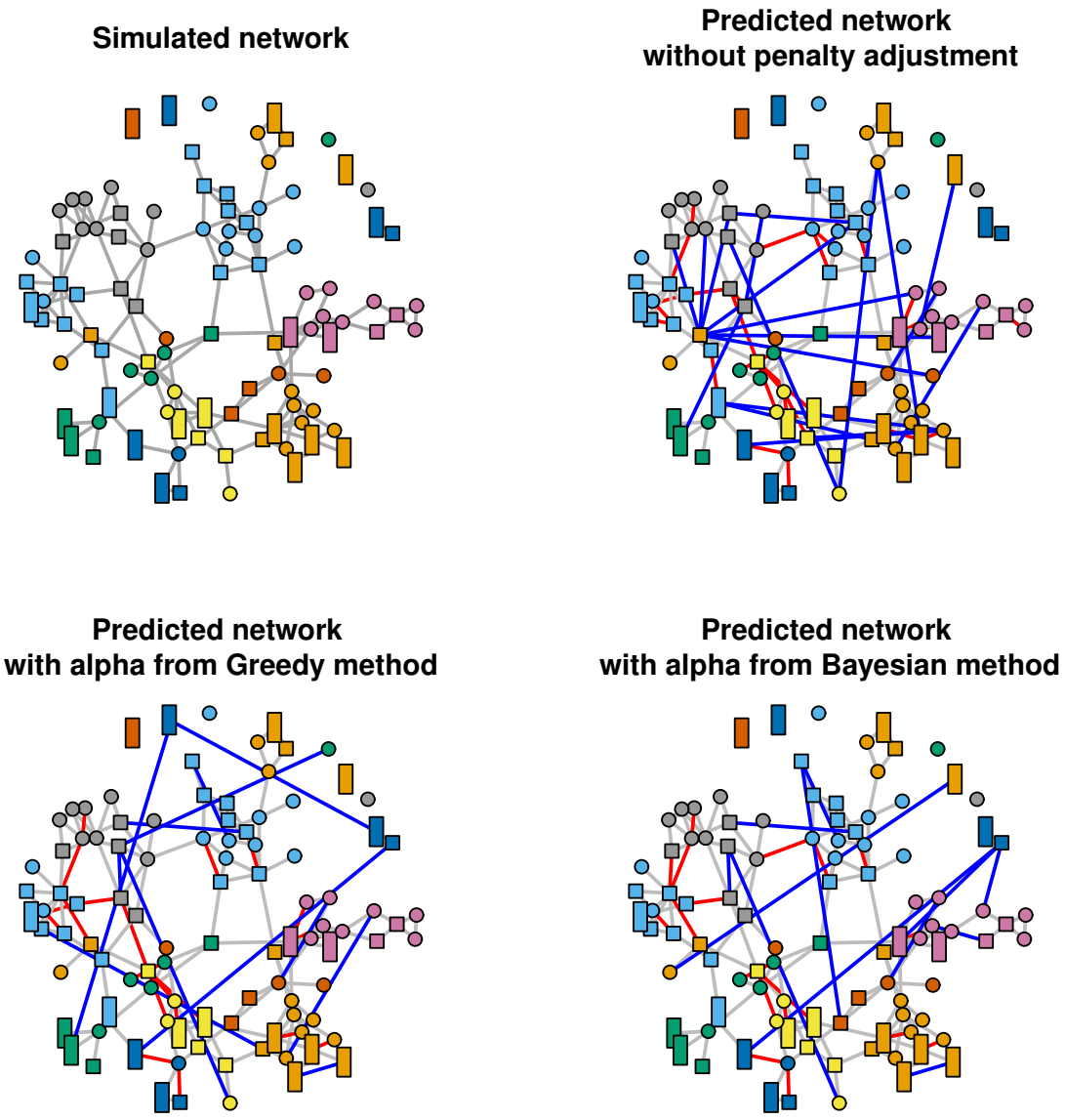

Figure 3.3: Predicted ten-family community network for Run 10 for the model without penalty factor, with the penalty factor estimated using the greedy approach and with the penalty factor estimated using the Bayesian approach. Grey line:True Positive; Blue line:False Positive; Red line:False negative. Node color:last name; Node shape:social groups. In Run 10, the $\hat{\alpha}_h$ are similar for last name; the greedy approach gives slightly higher values for group covariates. We see fewer false positives using the greedy approach.

them (394) are male, about 8% (34) are female, and for the rest the gender is unknown. Women who appear in these data are usually associated with men in the data through family or marriage.

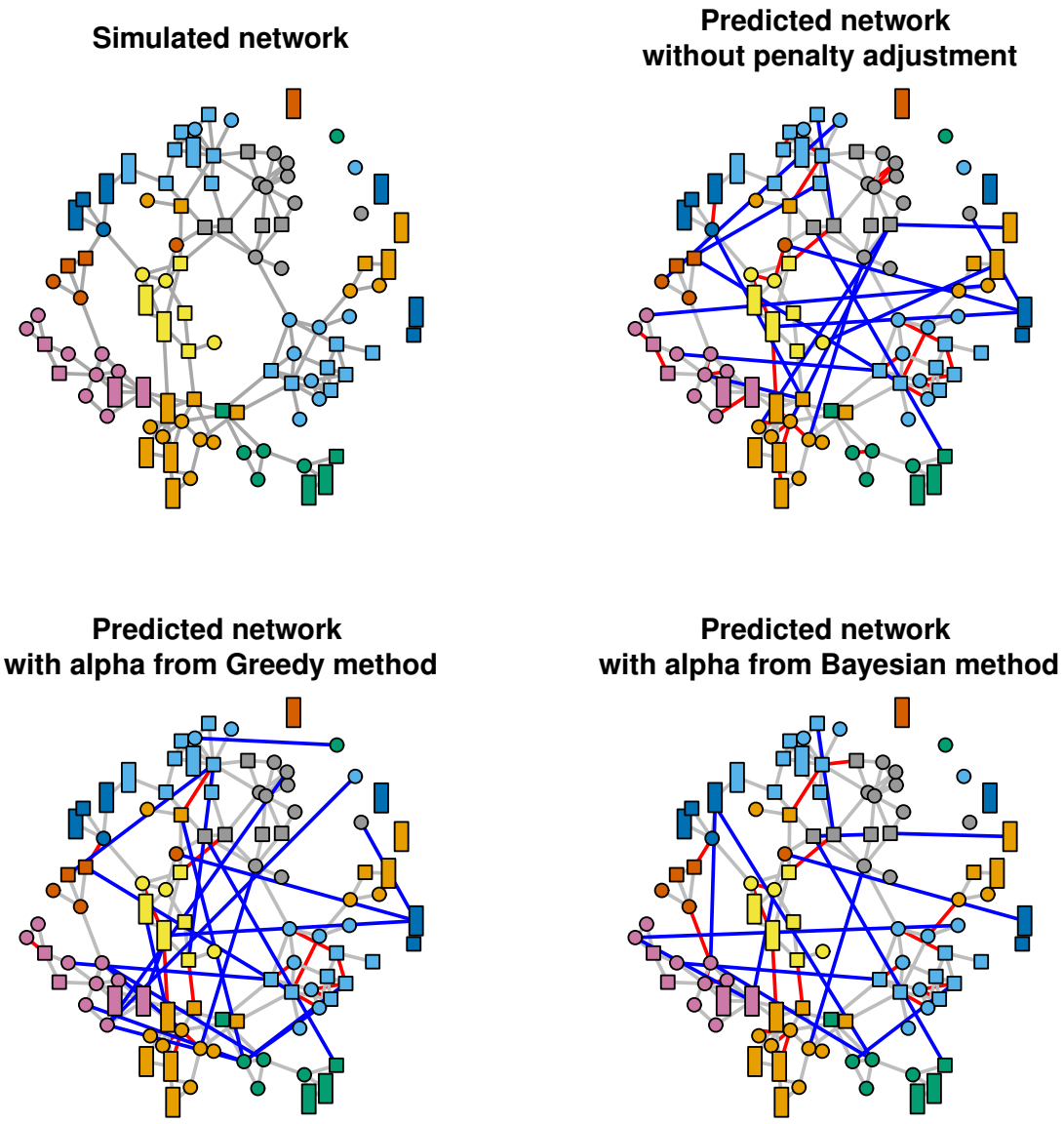

Figure 3.4: Predicted ten-family community network for Run 1 for the model without penalty factor, with the penalty factor estimated using the greedy approach and with the penalty factor estimated using the Bayesian approach. Grey line:True Positive; Blue line:False Positive; Red line:False negative. Node color:last name; Node shape:group membership. In Run 1, the greedy approach tends to give a smaller penalty on last name but a larger penalty on social groups which gives us more false positives and a few fewer false negatives within social groups.

Apart from last name and birth and death year, we here consider three other covariates, related to individuals' occupation. We distinguish three groups:the Writer group (the occupation variable in the data contains the words "poet", "writer" or "author"), the Church group (occupation contains "church",

	True Positive	False Positive	False Negative
No penalty factor	121	17	33
Greedy	139	21	19
Bayesian	138	14	21

Table 3.4: The number of true positive, false positive, and false negative links for Run 1.

"religious", "bishop" or "catholic"), and the Royal group (occupation contains "royal", "king", "queen" or "regent").

Table 3.5 includes some descriptives of the data. Since we have limited the data to people who were alive in a period of 75 years, the lifespan of most pairs of people overlapped. Compared to the simulated data, the proportion of pairs with shared last name is much smaller. This indicates a more diverse last name distribution (the most common last name "Stewart" is the last name of royalty during this period and appeared for only 9 individuals, while other last names appeared for no more than 5 individuals), but also suggests that as long as two people shared the same last name, the chances of them belonging to the same family and knowing each other are high. Among all occupations that were listed in the data, the writer and the church-related occupations are most popular. Individuals with a royal-related occupation tend to be closer connected than other people, which is why we consider this group, even though not that many people are part of it. People can have multiple group membership across the three groups. Five individuals are part of more than one group, like Roger Ascham, who was an author and a royalty tutor, and John Seton, who was a Roman Catholic priest as well as a writer on logic.

Table 3.5: Descriptive statistics of the SDFB data on people from the period 1500–1575.

Group	Number of people	
Writer	40	
Church	49	
Royal	19	
In multiple groups	5	
	Number of pairs	
Same last name	117	(0.13%)
Lifespan overlaps	81625	(92.8%)

We estimated the penalty parameters α using the Bayesian approach outlined in Section 3.1.2 since the Bayesian approach tends to give more consistent and reasonable estimation on the α. We find that

$$\begin{aligned} \hat{\alpha}_{lastname} &= -1.853 \qquad \hat{\alpha}_{writer} = \quad 0.369 \\ \hat{\alpha}_{church} &= -1.262 \qquad \hat{\alpha}_{royal} = -0.801. \end{aligned} \tag{3.7}$$

From the size of the penalty factors, the last name is the most important covariate, indicating that if two people share the same last name, this is a strong indication that they may know each other. It is interesting that not for all groups the penalty factor is negative:if two people are both a writer, they are less likely to be connected. It is possible that being a writer is an occupation for which little collaboration is required, so that the writers did not socialize much with their peers. On the other hand, if two people are both related to the church or the royal family, this increases their chance of being linked.

Next, we compare the networks generated by the Local Poisson Graphical Lasso model with and without penalty adjustment. The overall penalty level for both models is the one minimizing the MSE. For the model without penalty adjustment, the estimated network consists of 156 links and for the model with penalty adjustment, the estimated network consists of 135 links. Although they partially overlap, the two networks have also contain many different links. There are 40 links that are only picked up by the model with penalty adjustment picks up and 61 links that are picked up only by the model without penalty adjustment.

How do the penalty factor values α relate to the difference between the two estimated network? To answer this question, we consider the percentage of links estimated by the two that had covariates in common (see Table 3.6).

Table 3.6: Numbers and percentages of links estimated by the models with and without penalty adjustment, for whom the corresponding people had a common covariate.

Covariate	With penalty adjustment model	Without penalty adjustment model
Last name	19 (8.9%)	16 (5.8%)
Writer group	4 (3.0%)	4 (2.6%)
Church group	2 (3.0%)	6 (5.2%)
Royal group	8 (5.9%)	4 (2.6%)

As expected based on the negative penalty factor estimate for the last name and the royal group, the model with penalty adjustment picks up more links between people with the same last name or both related to the royal family. To be more specific, the model with penalty adjustment detects four additional links without losing the seven links that were estimated by the model without penalty adjustment model. However, the proportion of links between individuals from the writer or the church group does not differ much between the two models. Both models select one link between two people in the writer group. The model with penalty adjustment even picks one link less within the church group, even though the negative penalty factor α_{church} indicates that links between people within the church group are penalized less. Note that the difference in within-group estimated links only contributes a small portion of the difference among the estimated networks. This suggests that changing the penalty on the links between people within the same groups also affects the links that are not within those groups.

Finally, we approximate the precision of the estimated networks by looking for evidence for links on Wikipedia. For a link involved with two people, I looks for the Wikipedia pages for either person and as long one of the persons' Wikipedia document contains the other one's name, we consider this as evidence that a link exists. Of the 135 links that are picked by the model with penalty adjustment, we find evidence for 62 (45.9%). Of the 156 links that are picked by the model without penalty adjustment, we find evidence for 67 (42.6%).

There are 95 links overlapped in both groups. For the 61 links that were only detected by the model without penalty adjustment, we notice that some people show up repeatedly. George Wishart, who is listed as "evangelical preacher and martyr" and Thomas Wynter, who is listed as "clergyman" should both belong to the church group. However, when we first defined the group, we did not pick up words like "preacher" and "clergyman" to include the in the Church group, which causes the model with penalty adjustment did not pick up the links for them and also may lead to the lower linking rate in the church group in Table 3.6. This indicates it is important to systematically define the groups with all the synonyms considered and classified since people may use different words to indicates similar meanings in the literature work. On the other hand, for the 40 links that were only detected by the model with penalty adjustment, we also have some people show up repeatedly, like Katherine Seymour, who belongs to the royal family, which related to decreasing penalty for the royal family member. We also have Margaret Roper and Nicholas Udall in the group, who are authors closely related to the royal family.

There is no doubt that an in-depth analysis of these results would require the help from experts on British history, but from these preliminary analyses, it seems that the model with penalty adjustment yields a more precise and conservative estimate of the relationships.

3.3 Additional approach:Laplace approximation

Even though the simulation in Section 3.2 indicates that our greedy approach and Bayesian approach can estimate the penalty factors and thus improve the network reconstruction. Both of them are time-consuming thus hard to apply to larger data set. Therefore, in this section we explore whether there exists a more efficient way to estimate the penalty factor. For the Bayesian approach, we use the double approximation to model both Poisson regression likelihood and Laplace prior, since they are not conjugate pairs. However, approximating both of them to Gaussian ignores the Poisson nature of the data and using the L1 norm to replace the L2 norm did not enforce model selection during the estimation of α. Therefore, we decide to take an additional way to estimate objective function in the Bayesian approach.

Recall that in the Bayesian approach in Section 3.1.2, for a local case with a specific person j, we need to estimation marginal likelihood of α, which is

$$
\begin{aligned}
L_j(\alpha) &= \int_{\mathbb{R}^p} \prod_{i=1}^{n} p(Y_{ij} \mid Y_{i,\neq j} = y_{i,\neq j}, \Theta_j) \prod_{k \neq j} p(\Theta_{kj} \mid \alpha) \mathrm{d}\Theta_j \\
&= \int_{\mathbb{R}^p} \prod_{i=1}^{n} \frac{1}{y_{ij}^*!} e^{\lambda(y_{i,\neq j}) y_{ij}^*} e^{-e^{\lambda(y_{i,\neq j})}} \prod_{k \neq j} \frac{e^{Z_k^{j*}\alpha}}{4\sigma_j^2} e^{-\frac{\exp(Z_k^{j*}\alpha)}{2\sigma^2}|\Theta_{kj}|} \mathrm{d}\Theta_j
\end{aligned}
\tag{3.8}
$$

Instead of approximating both the Poisson regression likelihood and the Laplace prior in Equation (A.11) with a Gaussian distribution, we now use the Laplace approximation to directly approximate the integral. We will compare the network reconstruction performance (measured by precision and recall) of the Laplace approximation to double approximation.

The Laplace approximation is a technique used to approximate integrals of the form $\int e^{Mf(x)} \mathrm{d}x$. Usually, the method assumes that $f(x)$ is a twice-differentiable function. Notice that in our case, the Laplace prior is not differentiable when $x = 0$. We attach the proof that why the Laplace approximation still can be used to approximate the integral in Appendix A.3. Moreover, to simplify the calculation, we adopt another form of Laplace approximation which was proposed in Butler (2007).

We let

$$
g(\Theta_j) = \frac{1}{n}\Big(\sum_{i=1}^{n} -log(Y_{ij}!) + Y_{i,\neq j}\Theta_j Y_{ij} - \exp(Y_{i,\neq j}\Theta_j)\Big), \tag{3.9}
$$

$$
h(\Theta_j) = \prod_{k \neq j} \frac{e^{Z_k^{j*}\alpha}}{4\sigma_j^2} e^{-\frac{\exp(Z_k^{j*}\alpha)}{2\sigma^2}|\Theta_{kj}|} \tag{3.10}
$$

where the function $g(\Theta_j)$ is maximized at Θ_j^*. As $g(\Theta_j)$ is the Poisson regression log-likelihood, Θ_j^* is the MLE when there is no regularization. The Laplace approximation yields

$$
\begin{aligned}
L_j(\alpha) &= \int_{\mathbb{R}^p} e^{ng(\Theta_j)} h(\Theta_j) \mathrm{d}\Theta_j \\
&\approx (\frac{2\pi}{n})^{(p/2)} \frac{e^{ng(\Theta_j^*)} h(\Theta_j^*)}{|-H(g(\Theta_j^*))|^{1/2}} \quad \text{as } n \to \infty
\end{aligned}
\tag{3.11}
$$

where $H(\cdot)$ denotes the Hessian. To calculate the exact likelihood is time consuming but notice that we only need to maximize the likelihood to achieve the penalty factor α. The only part in Equation (3.11) that contains α is $h(\Theta_j^*)$. Therefore, to estimate $\hat{\alpha}$, we only need to maximize

$$
h(\Theta_j^*) = \prod_{k \neq j} \frac{e^{(Z_k^{j*}\alpha)}}{4\sigma_j^2} e^{-\frac{\exp(Z_k^{j*}\alpha)}{2\sigma^2}|\Theta_{kj}^*|} \tag{3.12}
$$

with respect to α. Notice that Equation (3.12) is only for one specific person j. We need a global α that optimizes every local likelihood. To get a global estimate for α, we take

$$\begin{aligned}
\arg\max_{\alpha} l(\alpha) &= \arg\max_{\alpha} \sum_{j=1}^{p} \log(L_j(\alpha)) \\
&= \arg\max_{\alpha} \sum_{j=1}^{p} log(\prod_{k\neq j} \frac{\exp(Z_k^{j*}\alpha)}{4\sigma_j^2} \times \exp(-\frac{\exp(Z_k^{j*}\alpha)}{2\sigma_j^2}|\Theta_{kj}^*|)) \qquad (3.13) \\
&= \arg\max_{\alpha} \sum_{j=1}^{p} (log(\frac{1}{4\sigma_j^2})^{p-1} + \sum_{k\neq j}(Z_k^{j*}\alpha - \frac{\exp(Z_k^{j*}\alpha)}{2\sigma_j^2}|\Theta_{kj}^*|)) \qquad (3.14) \\
&= \arg\max_{\alpha} \sum_{j=1}^{p} \sum_{k\neq j}(Z_k^{j*}\alpha - \frac{\exp(Z_k^{j*}\alpha)}{2\sigma_j^2}|\Theta_{kj}^*|) \qquad (3.15)
\end{aligned}$$

This is a convex function that can be optimized directly.

3.3.1 Simulation

We apply the same simulation setting as discussed in Section 3.2. We first observe the $\hat{\alpha}$ estimated with Laplace approximation and the results for 10 runs are listed in Figure 3.5. Almost always, the method

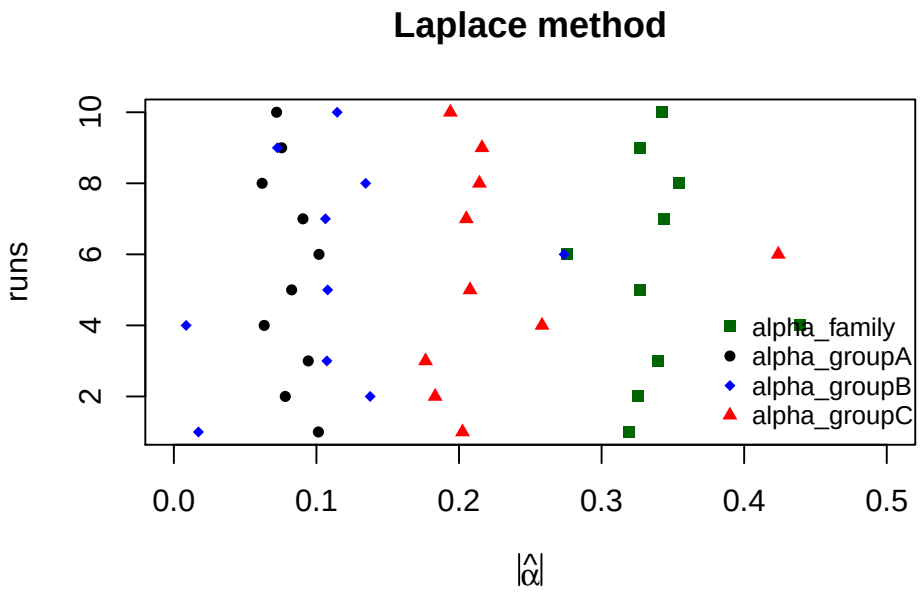

Figure 3.5: The distribution of $|\hat{\alpha}_h|$ estimated by the Laplace approximation over ten runs. The approach consistently pick the last name as the most important covariate. The approach also picks group C as the most important group.

selects family name as the most important covariate. However, compared to Figure 3.1, it incorrectly picks a smaller group with medium density, Group C, to be the most important social group, and this behavior is consistent.

Even though the method may fail to estimate the right α value it may still lead to an improvement in precision and recall. The best precision and recall estimated in 10 runs are listed in Figure 3.6 The Laplace

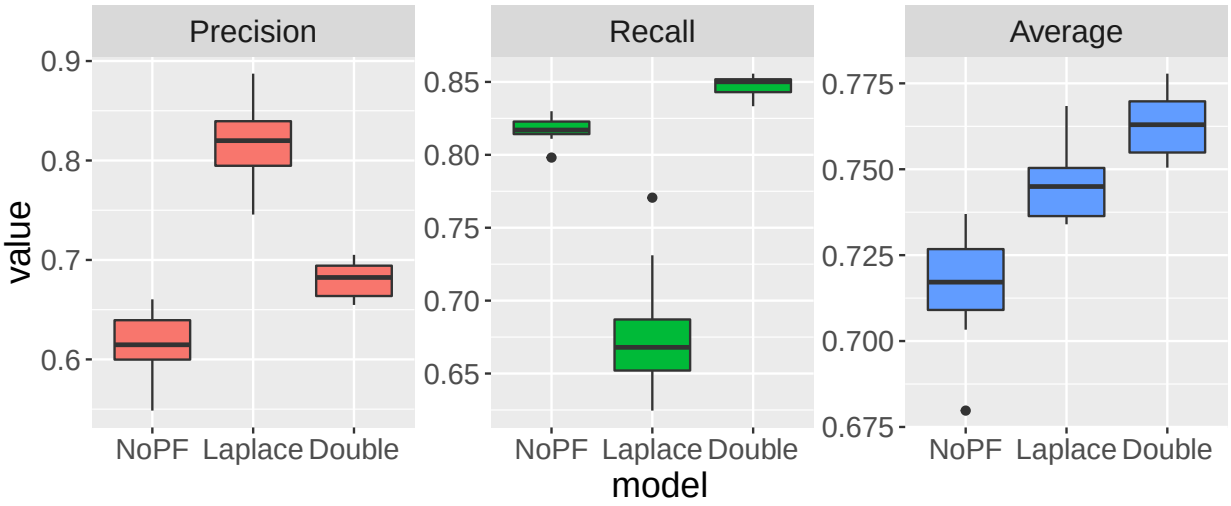

Figure 3.6: The distribution of precision and recall for the model without penalty factor, with penalty factor estimated using the Laplace approach, and with the penalty factor estimated using the Bayesian double approximation approach. Laplace approximation tends to pick a relatively smaller network as the best network, but on average, it is significantly better than the baseline without penalty factors but still worse than the double approximation.

approximation tends to pick a relatively smaller network as the best network, giving a higher precision but lower recall. On average, it is better than the baseline but still worse than the double approximation. However, the estimation is much faster (approx 1 hr vs 6 hrs), since double approximation needs to repeatedly estimate coefficients and α until convergence while the Laplace approximation only estimates Θ^* once.

We also try to understand why the Laplace approximation favors group C, the median density group with the fewest number of links. Therefore, we perform a few additional simulations to see how the $\hat{\alpha}$ varies in different cases.

First, we decrease the number of groups to be two and let 75% of the people belong to group A and 25% of the people belong to group B, and 150 links are added to each group besides the random links. We assume that the $\hat{\alpha}$ for group B should be always larger than group A since group B is denser. The result is in Figure 3.7. It seems that in general group B does have a slightly larger scale on $\hat{\alpha}$ but the difference is not large.

Next, we check how $\hat{\alpha}$ changes if the group sizes are the same while the number of links is different. We let group A and B both have 50% of the total population but add 200 links to group A and 100 links to

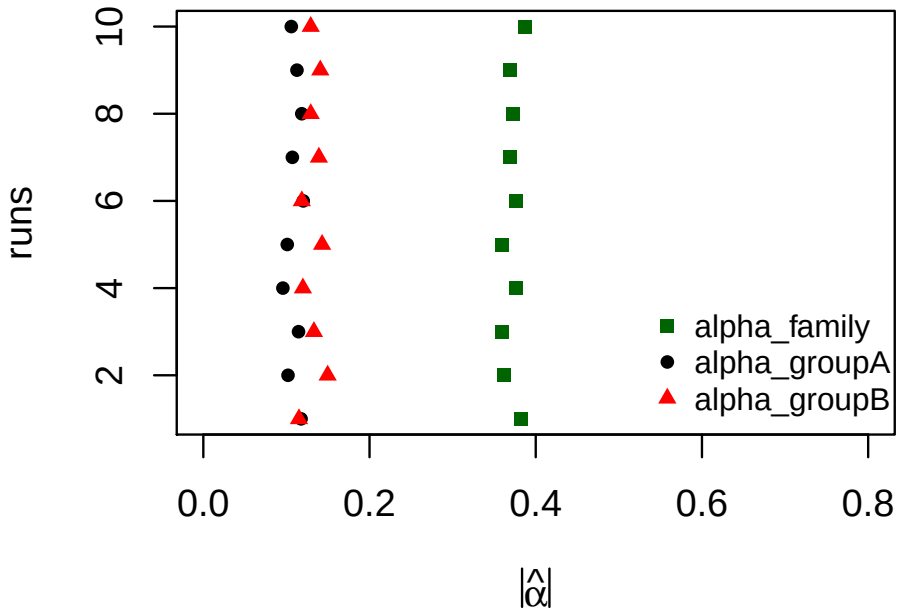

Figure 3.7: The distribution of $|\hat{\alpha}_h|$ estimated by the Laplace approximation with 2 groups with equal number of links over ten runs. The approach predicts group B is slightly more important than group A, but the difference is minimal.

group B. In this case, we expect that the $\hat{\alpha}$ for group A should be always larger than group B since group A is denser. The result is in Figure 3.8. In this case, there is no difference on $\hat{\alpha}$ estimated for both groups. Combining this with what we have found in the previous simulation, it seems that the algorithm favors the group with a smaller size regardless of the number of links within the group.

To verify this idea, we extend the group number to be three and let group A, B, C contains 1/2, 1/3 and 1/6 of the population while 150, 100 and 50 links are added to the groups respectively. The relation of density of groups should be $C \approx 2B \approx 4A$, thus we expect the scale of $\hat{\alpha}$ should also follow the order of CBA with decreasing values. The results are given in Figure 3.9. We find group C with the smallest group size to be the most important covariate, followed by group A, the one with the largest group size and also most links, while group B with the middle level of size and links is the least important. In general, it seems that the algorithm favors the small groups in the network.

Apart from group size and density, we have also considered whether other characteristics of the data may affect the result. We notice that the $\hat{\alpha}$ is estimated with Θ^*, the edge coefficients without regularization. However, for large sparse data, the estimation of Θ^* may be hard. Therefore, we explore whether the sparsity of the data may affect the value of $\hat{\alpha}$. We maintain the group size and the number of links to be the same as in Section 3.2 but only increase the error and true signal parameter from 0.004 to 0.1 in Section 2.5.1.

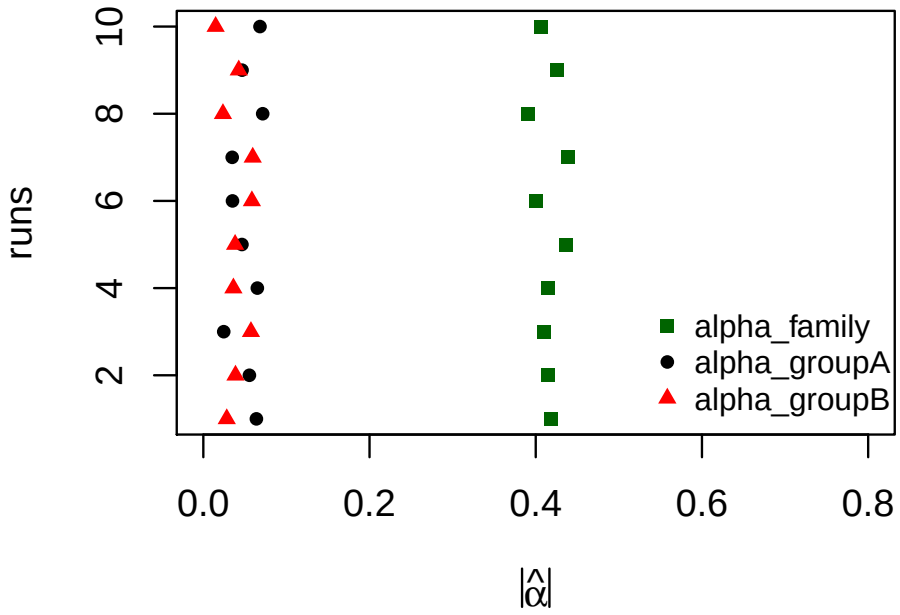

Figure 3.8: The distribution of $|\hat{\alpha}_h|$ estimated by the Laplace approximation with 2 groups with equal number of group size over ten runs. The approach predicts group A and group B are equally important.

Figure 3.10 shows that decreasing the sparsity does not change the order of the scale of $\hat{\alpha}$, and group C is still the most important one. Therefore, we can conclude that the sparsity of the data does not affect the estimation of α.

Another cause we have considered is the independence between groups. In the original simulation setting, people could only belong to one of the groups. Therefore, the covariate of groups is not independent. Now we let each person have a 50% chance to be in group A, a 25% chance to be in group B, and a 25% change to be group C. One person can belong to multiple groups at the same time. Figure 3.11 shows that Group C is still considered the most important one but the advantages are much smaller than we have seen in Figure 3.5. It could that with the Laplace approximation, whether the covariates independent do affect the estimation of the penalty factors.

From the simulation, it seems that the $\hat{\alpha}$ estimated with Laplace approximation does not directly reflect the density of the group. Unfortunately, the simulations fail to fully explain the reason. Moreover, there are a few theoretical reasons why the Laplace approximation may not be a good choice involved with Poisson data. If the mean of Poisson regression is large then the posterior may not be a Gaussian-like shape. Therefore, even though the Laplace approximation may be useful for future modeling of large data, we will continue the thesis using the double approximation.

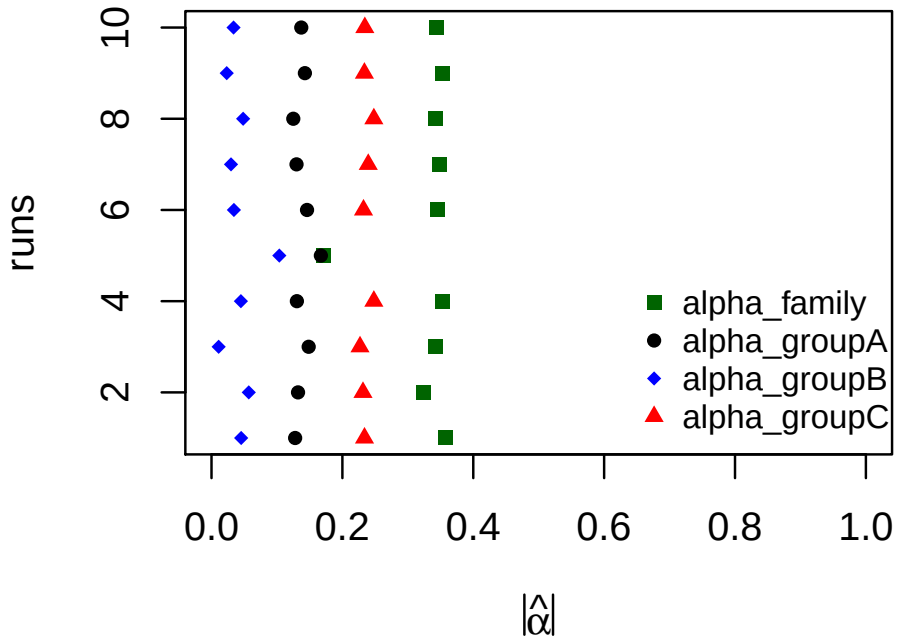

Figure 3.9: The distribution of $|\hat{\alpha}_h|$ estimated by the Laplace approximation with 3 groups with similar density over ten runs. The approach predicts group C with smallest group size to be the most important one, follow by group A, the one with largest group size and also most links while group B with middle level of size and links is the least important.

3.4 Compare the effects on coefficient constraints

3.4.1 Motivation and setting

In all analyses up to here, we chose to get rid of the negative coefficients as a post-modeling process since if the edge parameter Θ_{jk} is negative, it suggests that if we mention person j more, we are likely to mention person k less, which does not indicate a relationship between person j and person k. Therefore, one natural question here is whether we can incorporate this feature into the model. Including this post-modeling step may provide a more elegant solution but also may affect the links. Therefore, here we explore what happens if we constrain all the coefficients to be non-negative, and how this changes the α values and predicted links happens.

We use the social network simulated in Section 2.5.1 as the simulation set, but try to compare the $\hat{\alpha}$ values and predicted network with and without edge positivity constraints. When estimating the $\hat{\alpha}$, for the greedy approach, we fix the order of adding covariates and the fold of cross-validation, thus the change in $\hat{\alpha}$ should only be caused by the constraints on the coefficients, instead of other randomnesses in the approach. For the Bayesian approach, we also constrain the edge parameters to be non-negative then to maximize the

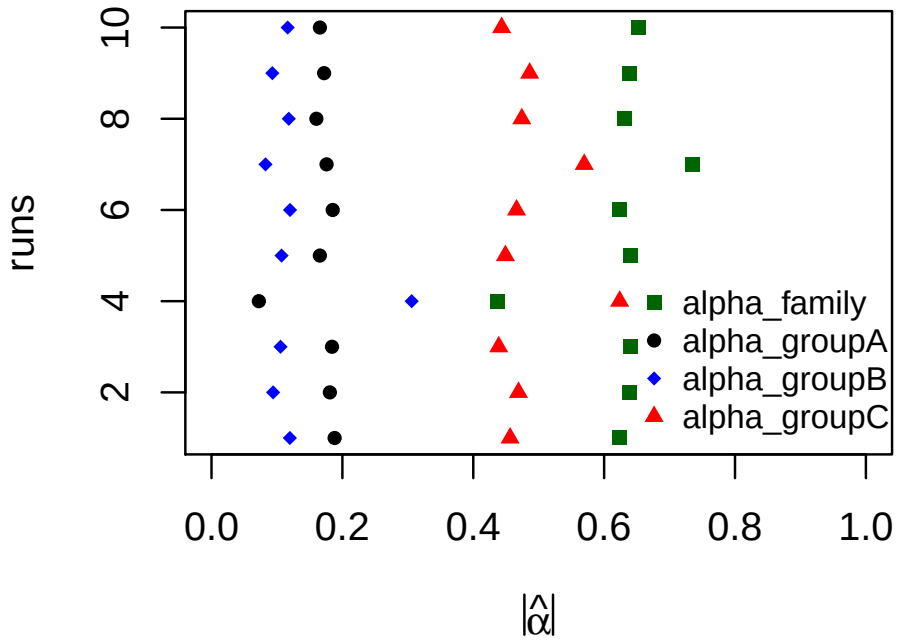

Figure 3.10: The distribution of $|\hat{\alpha}_h|$ estimated by the Laplace approximation with 3 groups with dense data over ten runs. The approach predicts similar distribution of $\hat{\alpha}$ as in Figure 3.5.

log-likelihood during the estimation. When estimating the edge coefficients, we simply let the lower limits option in the `glmnet` package be zero so that the coefficients will all be non-negative.

Once we get the edge coefficients, we also try to compare the difference between the estimated networks. We explore how different the two estimated networks are and whether two models will pick some specific links due to the constraints of edge coefficients. For example, do the constraints cause the model to favor links that are well separated or the links that have mutual connections? Do the constraints cause the model to pick up more links for important people, ones who have more connections than average? Do the constraints pick up links to connect an isolated people? To assess these differences, we evaluate the estimated networks through the following metrics:

- The number of estimated links and the precision and recall, to measure how well the models predict the network with and without constraints.
- The percentage of common links of the estimated network, to measure how different the two estimated networks are. For the set of links that are picked up by one model and not the other, are they true positive or false positive?

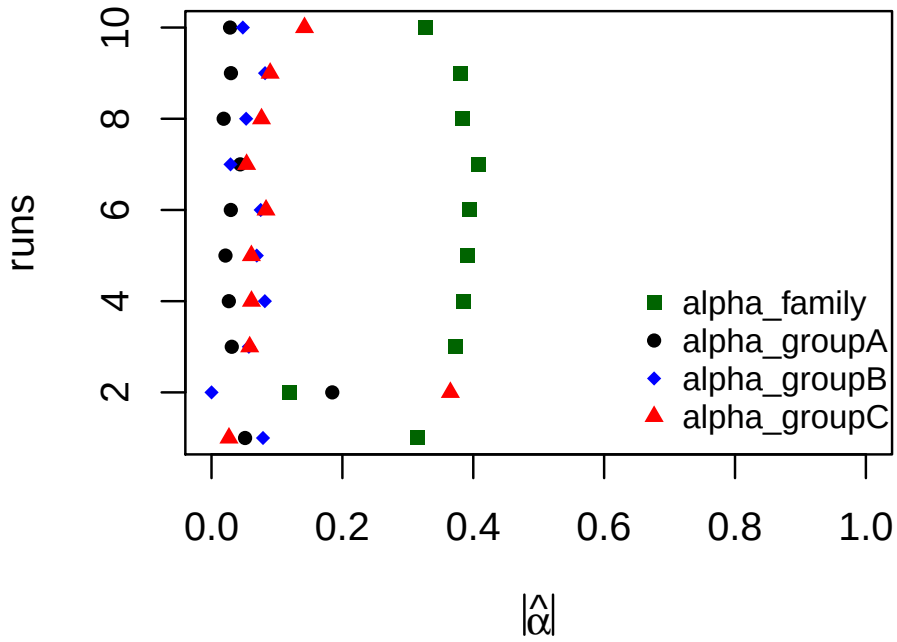

Figure 3.11: The distribution of $|\hat{\alpha}_h|$ estimated by the Laplace approximation with 3 groups with independence over ten runs.

- Average distances and distance distribution of the false positive links which only predicted by either model: long average distances indicate the model makes mistakes by picking the links that are well separated. A peak around distance 2 would indicate that the model picks more links to form triangles in the network.

- Degree distribution for the estimated network and false positive links which only predicted by either model: For estimated network, the degree distribution of the estimated network should be consistent with the original simulated network. If not, does the network over- or under-estimate the degree? We also zoom in to see how the degree of false positive links estimated by both models affect the general degree distribution.

- Number of false positive links that connect isolated nodes or components in the simulated network. It indicates whether the constraints will tend to connect the isolated components or not and whether the constraints will tend to average and even the degree.

3.4.2 Results

We first compare the $\hat{\alpha}$. For the greedy approach, the $\hat{\alpha}$ remain the same with and without constraints. It shows that having a constraints may not lead to a significant effect on the effects of the covariates. However, this may also due to the fact that the searching grid of the $\hat{\alpha}$ is not fine enough. On the other hand, the $\hat{\alpha}$ in the Bayesian approach did change. The $\hat{\alpha}$ values in Bayesian approach are displayed in the Figure 3.12. From Figure 3.12, it seems that the constraints do affect the $\hat{\alpha}$ values of the Bayesian approach a little bit. However, the signs do not change and the relative scales are also similar. Therefore, it seems that the constraints of the coefficients affect the overall penalties more than the covariates' penalties.

Next we focus on the difference between the estimated networks. The number of estimated links and precision and recall are given in Figure 3.13 and Figure 3.14. The simulated network has 1164 links, and all four estimation procedures tend to predict a slightly larger network. The Bayesian approach without edge positivity constraints on average estimates the largest network but with an edge positivity constraint, the number of links drops. Compared to the average of precision and recall in Figure 3.14, it seems that the decreases of links does not lead to an increase in accuracy. Instead of getting rid of the false links, the edge positivity constraints lead to an missing true links. On the other hand, the greedy approach tends to give a different result. With edge positivity constraints, the model tends to predict a larger network but the performance also drops. Therefore, it seems that adding constraints to the model does not lead to recovering more true links. In conclusion, it seems that the constraints do not lead to the improvement in the model performance.

Next we try to understand whether there is difference in the estimated networks. The distance distribution is one of the metrics to indicate the network structure and the distance distributions of data set 1 is given in Figure 3.15. It seems that all models, Bayesian and greedy, with and without constraints, successfully capture the true distribution.

Notice that among all the estimated links, approximately 80% are the same thus it may be hard to capture the difference when we look at the overall distribution. Therefore, we zoom in to see the links that are predicted by one model but not the other, and where are made mistakes. Figure 3.16 presents the average distances in the true network for pairs of individuals incorrectly linked by one model but not the other. For example, the left-most box indicates the average distance in the true network for pairs of individuals incorrectly linked by the Bayesian model but not the Bayesian model with edge positivity constraints. The figure indicates that the model without edge positivity constraints tends to pick more wrong links that are more separated from each other while the model with edge positivity constraints tends to pick the links with more common friends.

To verify this idea, we also plot the number of false-positive links with a mutual connection in the true network in Figure 3.17. The number of false-positive links with a mutual connection is higher in the model

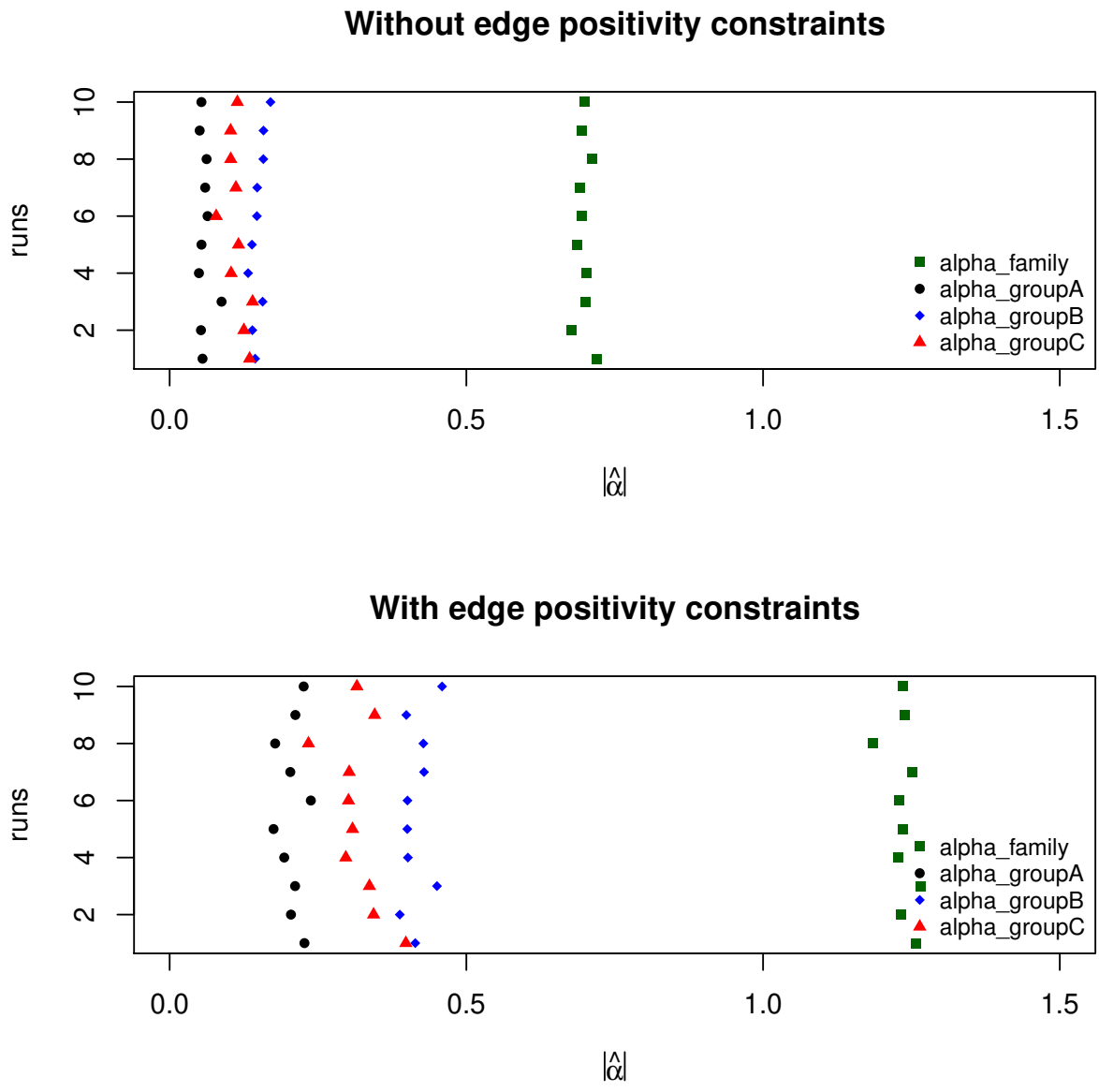

Figure 3.12: The $\hat{\alpha}$ values in Bayesian approach with and without edge positivity constraints. The actual $\hat{\alpha}$ values becomes larger with positivity constraints but the relatively scales are generally the same.

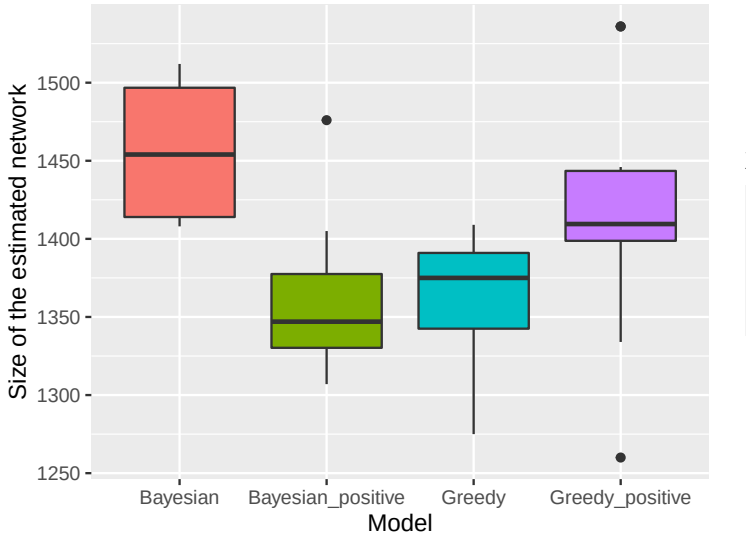

Figure 3.13: Boxplots of the estimated size of the network. All four estimation procedures tend to estimate a network slightly larger than the true simulation network. However, for the Bayesian approach, without edge positivity constraints, picking more links means picking more true links while for the greedy approach, with edge positivity constraints, picking more links but does not improve the recall.

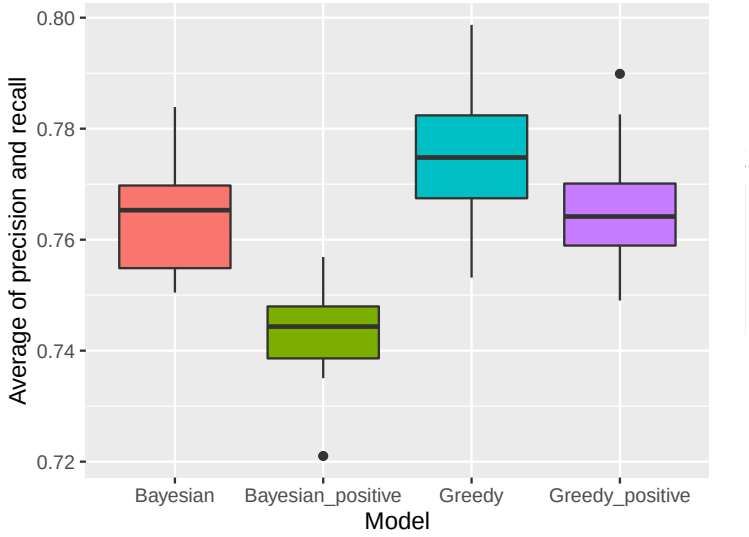

Figure 3.14: Boxplots of estimated average of precision and recall. With a edge positivity constraints, the overall performance, measured by the average of precision and recall, decreases. The constraints hit Bayesian approach more due to the no longer appropriate approximation of Laplace prior to normal prior during the estimation.

with edge positivity constraints. Therefore, we can conclude that the constraints model tends to introduce links to close triangles in the network. In general, if two people are well separated in the true network, even though the model incorrectly links them, it is easier to be detected compared to the case involved with

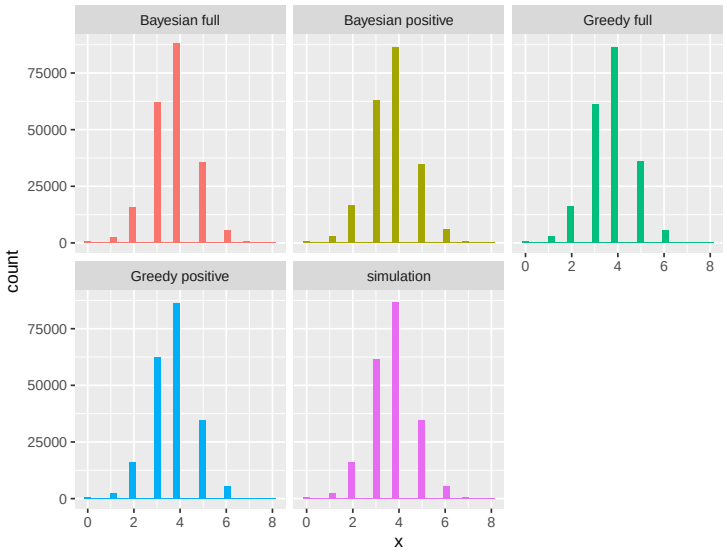

Figure 3.15: The distance distribution of simulated network and estimated network for data set 1. All models correctly capture the distance distribution in the true simulated network.

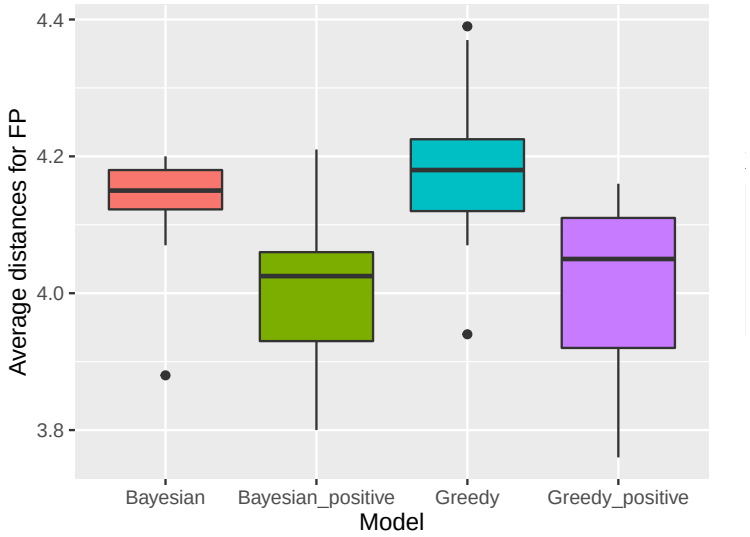

Figure 3.16: The average distances in the true network for pairs of individuals incorrectly linked by one model but not the other. The model with positivity constraints tends to pick more false positive links with nodes having shorter distance, while the model without positivity constraints tend to pick more false positive links with nodes that are well-separated.

"friends' friends". If we do want to be conservative about the "friend's friends" links, we thus probably need to avoid using the edge positivity constraints.

We also have checked the degree distribution which is depicted in Figure 3.18. In general, the estimated degree distribution is consistent with the true simulation network. However, notice that all the models tend

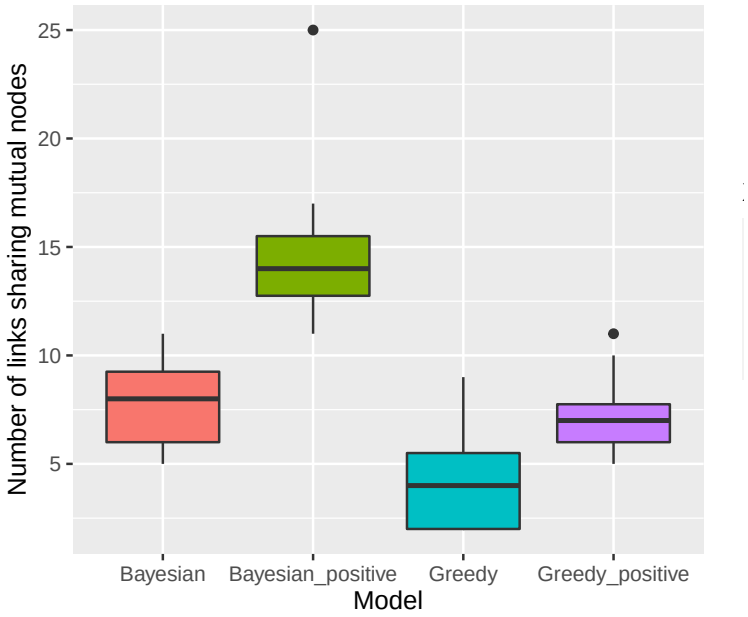

Figure 3.17: The number of false positive links with at least one mutual friend in the true network for 10 runs. The model with positivity constraints tend to pick more false positive links with at least one mutual friends.

to have a heavy tail on the degree distribution compared to the true simulation network. The largest degree in the simulation network is 13 while all models have a maximum degree around 30 to 40. The heavy tails may be related to our assumption with the Poisson model but also may be caused by how we simulated data. In the simulation process, we assume people have similar chances to be linked and there are no "popular" people, e.g., a king or queen may naturally connect to more people than average. On the other hand, from the degree distribution, it seems there is no significant difference in the degree generated by the model with and without edge positivity constraints.

As we have done in the distance distribution, we have also checked the difference of degree on the link sets predicted by one model but not the other. The result is given in the Appendix A.4 since there is no significant difference on the average degree. Therefore, adding the edge positivity constraints should not affect the overall degree distribution predicted by the local Poisson graphical lasso model.

The last thing we have checked is the number of links that involve at least one isolated node in the true simulated network. This is a special case in the degree that is not presented in the distribution plots and boxplots. The number indicates whether the model tries to connect the isolated nodes to be huge components. The result is also in the Appendix A.4 and again we also did not observe a significant difference between the various models.

In conclusion, even though including the edge positivity constraints helps us to avoid removing links with negative coefficients in a post-processing step, it does not help us to improve the overall performance

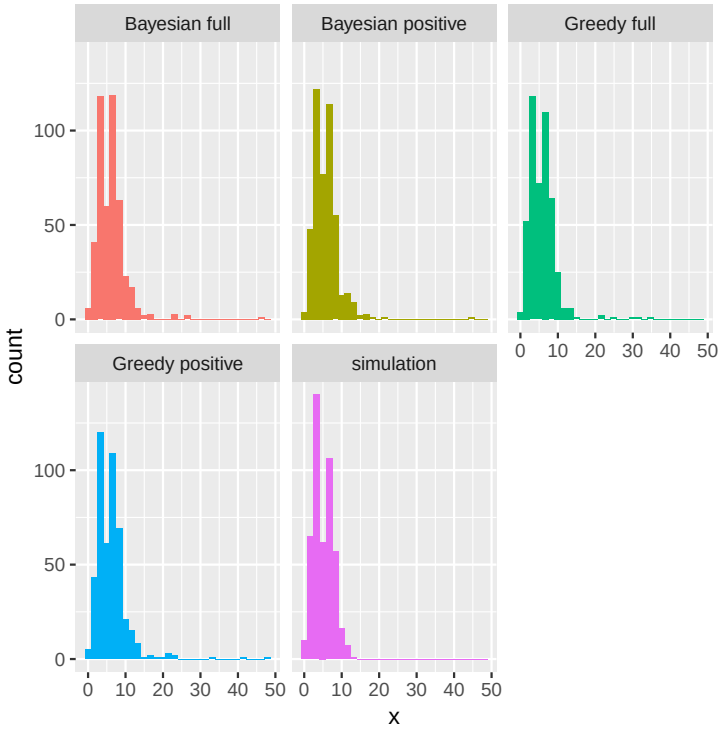

Figure 3.18: The distance distribution of the simulated network and the estimated networks. Overall there is no significant difference on the degree distribution estimated by different model but all the estimated degree distributions tend to have a heavy tail compared to the true simulation network.

of the network reconstruction, particularly for the Bayesian approach. The edge positivity constraints tend to introduce more false-positive links and form more triangles in the network.

Chapter 4

Including continuous covariates

In the previous chapter, we have described how to include binary covariates into the Local Poisson Graphical Model to improve the accuracy of network reconstruction. However, in some cases, binary covariates may not be enough to define the relations between people. Therefore for person j, we now define the covariate matrix $Z^j \in \mathbb{R}^{p\times m}$, with m covariates, by

$$Z^j_{kh} = \text{distance or similarity between person } j \text{ and } k \text{ on covariate } h \tag{4.1}$$

Notice that definition of Z^j_{kh} is an extension of the previous definition on the binary case, and Z^j_{kh} can also include a mix of binary and continuous covariates.

The hardest part of the definition 4.1 is about how to define the concepts"distance" and "similarity". For some covariates, it is relatively obvious, like the geographical distance or the last name similarity. However, some other text information may need a further process to convert it into numeric data. In this thesis, we present two applications of including continuous covariates into the local Poisson graphical lasso model.

4.1 Continuous covariates to deal with name misspellings

The approaches described in Chapter 3 have multiple applications. Besides applying to the historical text data, like biography to infer the historical social network, it can also be applied to fictional text to infer the social network in a novel. However, compared to data from a novel, historical text data tends to have a lower qualities. In most of the historical text data, transcript and misspelling errors are unavoidable. Even in official documents, sometimes you will have a person's name with more than one way to spell. For example, Anthony Ascham is an English astrologer whose last name can also be recorded as "Askham". Therefore, it is important to consider applying record linkage techniques, like edit distance or phonetics code into the modeling procedure.

Edit distance is a way to quantify how dissimilar two strings (e.g., words) are to one another by counting the minimum number of operations required to transform one string into the other. Common distances include Jaro-Winkler and Levenshtein indices. Edit distance can catch a lot of typo errors in a text. Meanwhile, another source of the spelling error in the historical documents is caused by similar pronunciation. Therefore, besides only using edit distance to compare two strings, another common approach is to first convert strings into phonetic codes and then compare the codes. Common algorithms for such conversion include Soundex and Metaphone.

Recall that in early simulation and real data examples, the last name is always the most important covariate to indicate whether two people have a relationship or not. So far, we assumed that last name data are spelled accurate and thus a binary covariate that only indicates whether two last names are the same or not is enough. Now suppose in the community, there exists some people's last names are similar, for example, Baker and Bacon, Walter and Waller. Besides that, there is also a chance that some letters in the last names accidentally have been switched.

In this case, we assume that adding one round of noise is to randomly select 20% of the last names and change one letter for each. Two rounds of noise to do the same thing again after the first round. Figure 4.1 shows the distribution of Jaro-Winkler distance on last names, with no noise, adding one round of noise, adding two rounds of noise, and adding three rounds of noise.

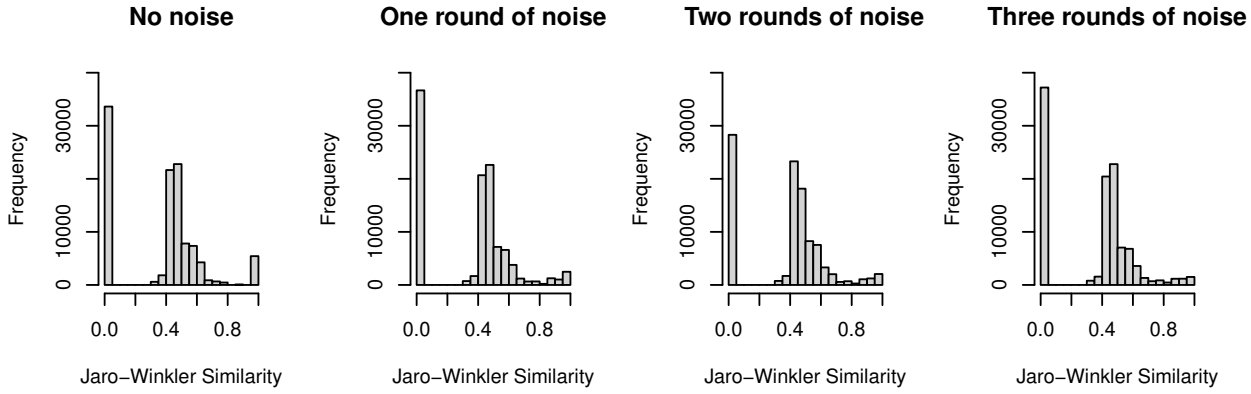

Figure 4.1: The distribution of Jaro-Winkler similarity of last names for each pair of individual in the simulation community.

When there is no noise added to the data, there are clear peaks when the Jaro-Winkler similarity is 0 or 1, which indicates a completely different or exact match. There is also a peak in the middle around 0.5, which indicates that there exist some similar last names in the community. When the first round of noise is added to the data, the proportion of exact matches largely decreases. After the second round of noise is added to the data, not only does the proportion of exactly matched decrease, the proportion of completely different is

also decreasing, since some letter switches may also make two different strings more similar. After the third round of noise is added, the proportion of completely different name increases again, while the proportion of exact matches is still decreasing.

The three rounds of noise are corresponding to how the noise affects the similarity of strings. We would like to see how different covariates work under different levels of noise. In this simulation, we use three methods to define the covariates for the last name:

1. Binary comparison on whether two last names are the same
2. First compares the last name pairs through Jaro-Winkler similarity. If the similarity is larger than 0.8, we consider two strings are matched (coded as 1). Otherwise, it is not matched (coded as 0).
3. Use Jaro-Winkler similarity as a continuous covariate.

The first method is the one we used in the previous simulation. The second method, even though still defining a binary covariate, includes Jaro-Winkler similarity to introduce a soft comparison boundary pre-modeling. The boundary value 0.8 is chosen based on Figure 4.1. The third method directly includes the Jaro-Winkler similarity, so the covariate is not binary anymore.

The new simulation community contains 464 people with 1170 relations, which is similar to the previous simulation setup. Besides of the 20 random last names, we have also include 5 pairs of similar last names: "Bacon" and "Baker"; "Hatton" and "Hobart"; "Smythe" and "Smith"; "Murray" and "Morton"; "Waller" and "Walter". After we generating the family members and network relations, we add one/two/three round(s) of noise to the last name. For each round, we generate a document-by-person matrix with 2000 documents. Then we apply three different methods to calculate the covariate for the last name, and finally, use the Bayesian method to calculate the penalty factor α for each method and corresponding ROC. The α value for each round and each method are showed in Figure 4.2.

Figure 4.2 indicates that the level of noise does not change the value of α significantly but the method of defining covariates does. The two binary covariates tend to have similar values while the continuous covariate tends to have a much smaller scale. One of the potential reasons for the behavior is that even though introducing the Jaro-Winkler similarity may help to catch the fuzzy names in the text, the middle-level Jaro-Winkler similarity may weaken the continuous covariates. From the histogram in Figure 4.1, we can see there is a large proportion of Jaro-Winkler similarity are around 0.5. Those should not be caused by the typos thus they should be interpreted as non-match, just like the pairs with 0 similarities. For example, suppose we have three pairs of people, pair A is two people with the last name "Bacon" and "White" respectively and the Jaro-Winkler similarity is 0; pair B is two people with last name "Bacon" and "Cecil" and the Jaro-Winkler similarity is 0.47 and pair C is two people with last name "Bacon" and "Bakon" and the Jaro-Winkler similarity is 0.89. Simply looking at the last names, we know only pair C has a higher

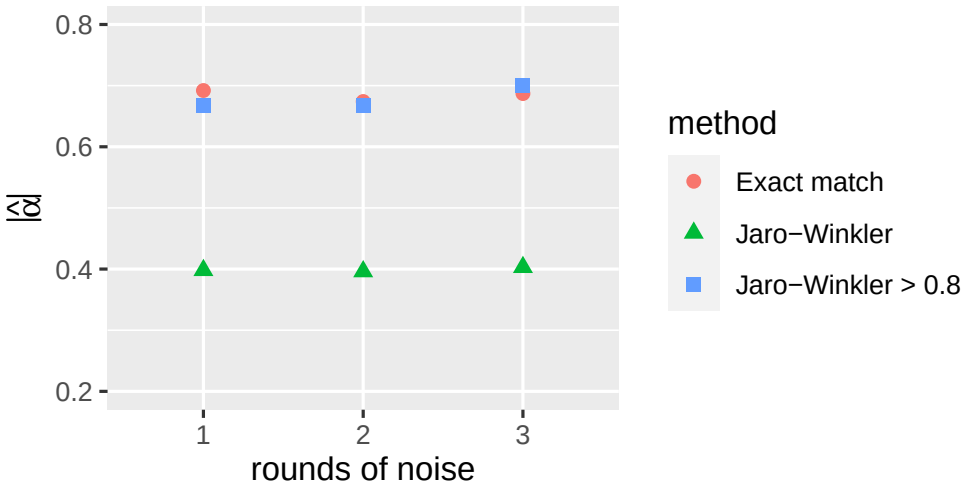

Figure 4.2: The $\hat{\alpha}$ estimated with Bayesian approach with one/two/three round(s) of noise. There is no significant difference on the three rounds and two binary covariates tend to give similar $\hat{\alpha}$ while the value for continuous covariate drops.

probability to be in the same family. For pair A and B, even though 0.47 is larger than 0, it does not indicate pair B is more likely to be in the same family. In this case, using a continuous covariate may try to distinguish the low-level similarity and the middle-level ones, and thus make the covariate less useful.

We have also compared the ROC for each noise round and each method with the estimated $\hat{\alpha}$ and the result is given in Figure 4.3. In all three rounds, although the difference is not significant, the binary covariates with a soft cut-off tend to have consistently better precision and recall than the other two. When the noise level is large, like with three rounds of noise, continuous covariates start to perform better than the binary covariate without soft boundary (exact match). Therefore, even though we can incorporate a continuous covariate for the last name, it seems that using a binary covariate with a soft boundary is more suitable to address the covariate of the last name when there are misspellings in the text.

4.2 Continuous covariates with people's historical significance

Besides the last name, some other individual characteristics can also be incorporated in the model as a continuous covariate, for example, one's occupation. In the analysis of Chapter 3, we have only considered whether two individuals are sharing a specific occupation or not, which led to a binary covariate for each selected occupation. However, in the previous analysis, we only worked with three occupation categories and it has been manually decided whether each person belongs to a group or not, since people may use different words to describe similar occupations. Even we can manually classify the occupations, when the size of data is increasing and we have a more and more different occupations, having a separate covariate for each

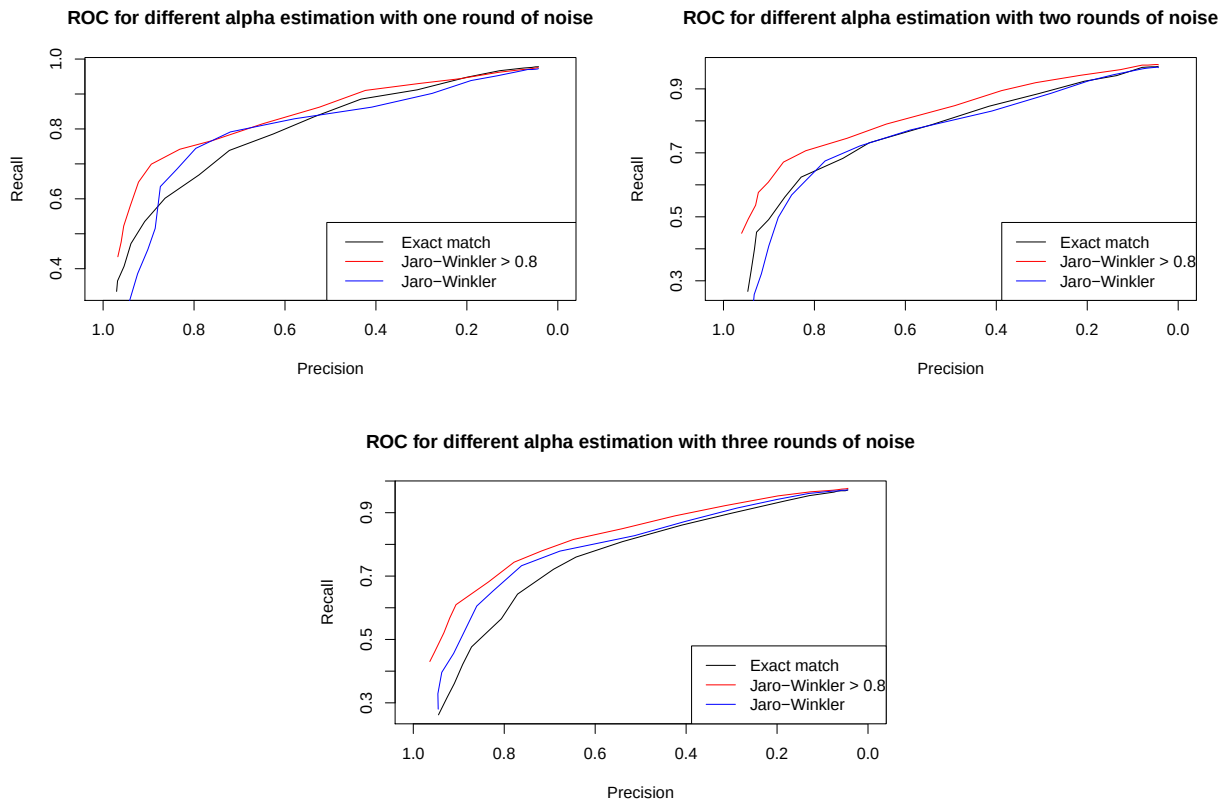

Figure 4.3: ROC for each noise round and each methods with the $\hat{\alpha}$

occupation largely increase the computational complexity when we try to compare the penalty parameters $\hat{\alpha}$.

In ODNB's data, two pieces of information are related to people's social identities. First of all, some people may have a sentence of summary of their historical significance, usually indicates their occupation or their family relationship. For example, "lord chancellor, politician, and philosopher" for Francis Bacon and "maid of honor to Elizabeth I" for Elizabeth Southwell. To contrast with the historical significance, we also have accessed to a manually labeled covariate summarizing people's occupations. Some of the labels are the same as the historical significance while some others are very different. For example, for Francis Bacon, both occupation label and historical significance are "lord chancellor, politician, and philosopher", and he is the only one in the data with such label. On the other hand, the label "politician" includes such a large range of historical significance text, including "army officer", "administrator", "Church of Scotland minister" and "member of parliament".

Besides the individual information, we also have a separate list of social groups during the time. There are different types of social groups, some of them are the larger group involved with social identities, for

example, the Judges and Catholics, while some the groups are much smaller social circles and literature groups, such as Translators of the King James Bible and Castalian Band.

4.2.1 Using network node distances to measure the closeness of the text

In Chapter 3, we have shown how to add occupation as one binary covariate to indicate whether two people sharing a specific occupation or not based on the historical significance information. However, when evaluating the real data examples, we realize that not all historical significance entries are about people's occupation and even for the ones that indicate on occupation, there is no unique way to record similar jobs. The way we manually construct the groups in Section 3.2.2 are inefficient, not quite accurate, and only uses a small portion of the data. Even though now we have the manually labeled data, we should still explore methodologies to quickly access the text information to avoid further label cost.

First, we want to take a closer look at the text data of historical significance. Among the 464 people in the real data example in Section 3.2.2, 374 of them have information on historical significance. In Figure 4.4, we show the word cloud of most frequent 100 entries. The size of the words represents their frequency. Even though we have seen a lot of occupation words, like courtier and printer, there are also a few high-frequency words that are more associated with identities, like rebel and murder victim. Moreover, a bunch of people has multiple occupations and identities listed. To address the multiple identities, when we compare whether two people are sharing the same occupation we assume that we need a partial match so that as long as one of the words in people's historical significance is the same, it should be considered a "match". For example, if person j's entry is "writer and artiest" while person k's entry is "writer", we should consider it is a "match": they share the same occupation.

With the idea of a partial match, we first create a Network among historical significance roles. In Figure 4.5, each node represents a distinct historical significance entry while if there is an edge between two nodes, then at least one of the words in the historical significance roles are the same, after getting rid of all the stop words, like "and", "of", "in", etc. The network is one large component with a bunch of isolated points. Simply from the virtualization, there is no clear clustering pattern in the network. We have also labeled the nodes we used to manually code the three occupation groups. As indicated in Figure 4.5, those groups also do not form a clear clustering pattern. The church group is relatively compact while the poet group and royal group are more sparse. This may be because we manually selected some people to be in the same group when their historical significance indicates synonyms, for example, Bishop and clergyman.

The colored points show only a part of the information available, which indicates there is a lot of text information we have not considered so far. Therefore, we like to develop a way that can quickly explore and classify the information in historical significance entries, so that we can create covariates related to people's occupation and identities. If we have other similar short text information for people's education or title, we

Figure 4.4: Word cloud for top 100 entries of historical significance. The size represents the frequency.

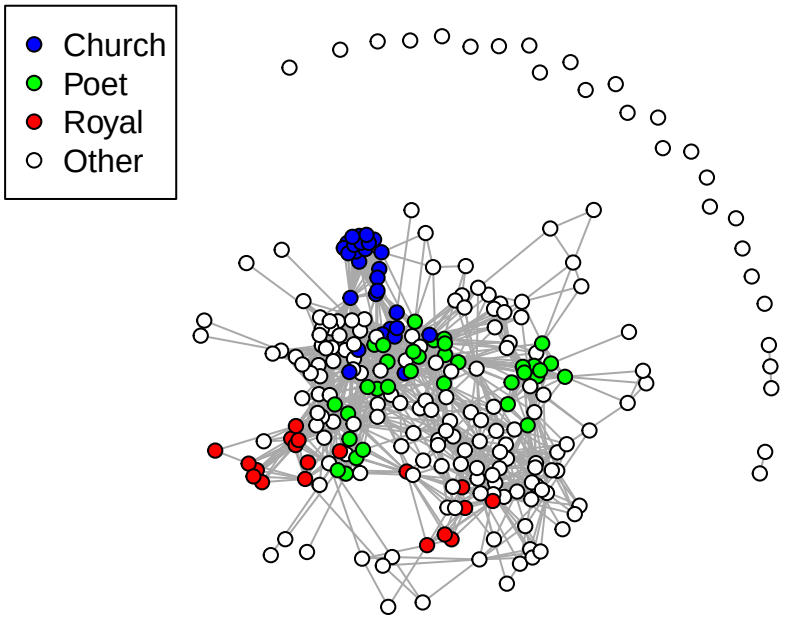

Figure 4.5: Network among historical significance roles in the 1500-1575 SDFB data: if two historical significance entries nodes have at least one word in common (excluding the stop words), then they are linked.

can use a similar approach to create corresponding covariates. When we try to create such covariates there are several things we need to consider:

1. In Section 3.2.2, we created one covariate for each type of occupation. Having one covariate for a specific type of occupation can help us learn how people in this particular type of career connect especially if we do have interest in that occupation. However, it is not practical to do this in the general case when we try to include all kinds of historical significance, since the computational complexity will be too large. Therefore, we would like to use fewer covariates to represent the information.

2. Previously, we only considered binary covariates to represent whether the two people's occupations match or not. However, doing so, we miss all the cross-occupation information. For example, suppose person j is a nobleman, person k is a politician, and person l is a translator. In the previous setting, person j will be labeled as a non-match for both person k and person l. However, from common sense, we would expect person j to be closer to person k than person l. Therefore, we want the covariates to reflect such distance.

3. In the ODNB, there are no standard categories for the historical significance. Therefore, people may use different words to describe similar occupations. For example, people can be recognized as courtiers or politicians, while both words describe similar jobs. Also, we may have occupations that are naturally associated with each other. For example, lawyers and judges should work together and politicians usually come from the nobility. Therefore, only looking at the pairs who shared the same information may lead us to miss a lot of information. When we create the covariates, we need to add a certain soft boundary to compare the text information.

To address the first two points, we start with the occupation network we have created in Figure 4.5. In the network, the shortest distance between nodes can be viewed as the closeness between people's occupation and identity. For example, in the occupation network, we may have the following two shortest paths starting from "writer" extracted from SDFB data:

(1) Writer $\longrightarrow$ Writer and translator $\longrightarrow$ Translator

(2) Writer $\longrightarrow$ Writer and protestant martyr $\longrightarrow$ Evangelical theologian and martyr $\longrightarrow$ Theologian and military engineer $\longrightarrow$ Military engineer

The length of path (1) is 2 and the length of path (2) is 4. Therefore, we can conclude that the writer is closer to a translator than a military engineer, which fits the common sense. Let l_{jk} represent the length of the shortest path between person j and k, and let l_{jk} be ∞ if person j and person k are in different network components. Figure 4.6 shows the distribution of l_{jk}. When l_{jk} is larger, the effect on penalties should be smaller. Also, notice that not everyone in the ODNB has a entry of historical significance. Therefore, the

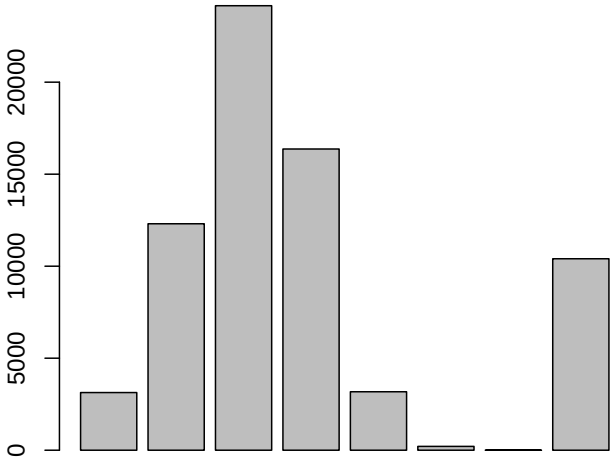

Figure 4.6: The barplots of distances (the length of the shortest path) between any two people with entries on historical significance.

covariate should only control the penalties between those people with a historical significance entry and who are also connected. Thus, we can define following covariates Z_k^j for the historical significance based on the network in Figure 4.5:

$$Z_k^j = \begin{cases} \frac{1}{l_{jk}} & l_{jk} < \infty \text{ and both person } j \text{ and person } k \text{ have entries on historical significance} \\ 0 & \text{Otherwise} \end{cases} \tag{4.2}$$

The above-mentioned definition provides one continuous covariate that represents people's closeness through occupation and identity. It is easy to create without any additional information and it works well when the text is short. We can also adapt the network such that nodes are linked only if they have a noun in common.

However, in this way we do miss the ability to interpret the effect on penalties within a certain occupation, like what we did in Section 3.2.2. One potential solution is that we can create a sub-network from Figure 4.5 so that we have another covariate to control the additional penalty limited to a certain group of people. For example, suppose we have a special interest in the printers and what to know how if two people are printers this affects the probabilities of them knowing each other. Then we can take out all the printers related nodes and form a sub-network and corresponding covariate. In that case, we can interpret how if two people are both printers this affects the probability of linking on top of the effect of being sharing the same occupation. However, to generate such a sub-network, we need a way to classify people who are printer-related.

4.2.2 Using relaxed word mover distance to measure the closeness of the text

However, Figure 4.5 only considers whether there is a common word in the historical significance entries. We would like to add some more links to the network, such that people with similar jobs but described in different ways or people who are usually working together can also be connected. Moreover, the network in Figure 4.5 is created by checking whether there is an exact word overlapped. Even though the ONDB data are relatively clean without any typos, still, people may use different words to describe similar occupations or identities. For example, a person can be described either as "diplomat" or "courtier". Therefore, it seems that we need to consider synonyms in text.

To overcome this problem, we first considered natural language processing methods like topic models. However, the phrases we have are short, and based on the historical significance network in Figure 4.5, it is hard to see the clustering patterns among those phrases. Therefore, it is hard to decide the number of topics. In the end, we therefore adopted the relaxed word mover's distance (WMD) (Kusner et al., 2015). This approach has two steps. First, it manages to identify synonyms by fitting a word embedding model, like GloVe, on a "dictionary" to convert all words into vectors so that the meaning of the words are represented by the location of the vectors in the high dimensional space (Pennington et al., 2014). Second, given the distances between each word, it manages to measure the distances between the actual text by aggregating the minimum changes to convert one text to the other.

To create a dictionary, we adopt the latest version of the ONDB data with contains manually labeled occupations for each biography owner. For the 13309 biography owners, there are 4423 different occupation labels. We first merge all the historical significance texts with the same occupation label into one document. Now we have one document for each occupation label and this document contains all the words ODNB authors have used to describe the corresponding occupation. An example of the process is given in Figure 4.7.

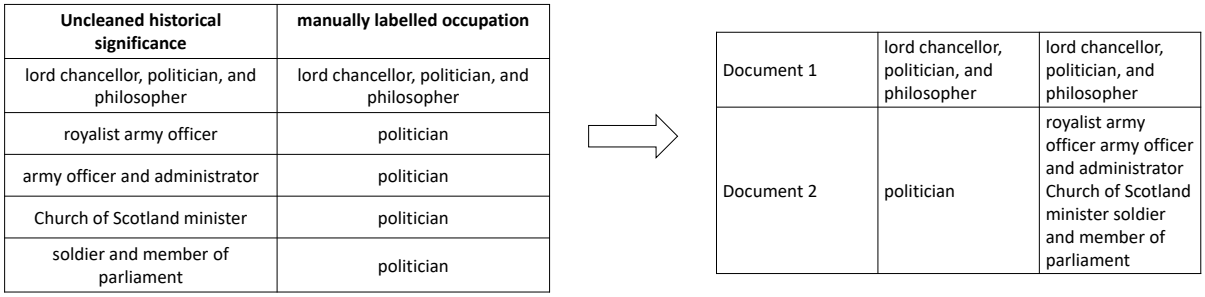

Figure 4.7: An example of merging the historical significance text into occupation documents.

Once we have the occupation documents, the next step is to fit a word embedding model, like GloVe, to find the distances between the words. GloVe assumes that if two words' meanings are closer, they should appear together in the text more often. Thus, we first create the co-occurrence matrix C, where C_{ij} represents how many times word i and word j appear in the same occupation documents. Then we can define the cost function S_{ij} for each word pair j and j:

$$S_{ij} = |w_i^T w_j + b_i + b_j - log(C_{ij})|, \tag{4.3}$$

where w_i and w_j are vectors for words and b_i, b_j are scalar biases to present each person's popularity. Then the goal is to minimize the sum of all pairwise cost functions to find w.

Once we have the distances between words, we can calculate the distances between texts. The approach is similar to the edit distance where we calculate how many changes we need to do to change one word to the other. Instead, here the distance represents how many word changes we have to do to change one text to the other. After calculating distances between all pairs of words, the final distance on the text is the sum of weighted minimum changes to convert one text to the other. Here is an example of text changing. Suppose we have three phrases:

Text A: courtier

Text B: diplomat and writer

Text C: writer

In the historical significance network, Text A and B will not be linked since they do not contain a common word. However, courtier and diplomat are synonyms thus they should be closer than the distance between courtier and writer. We let $d_{a,b}$ be the Euclidean distance between word a and word b, let t_{AB} be the cost to change from Text A to Text B. Notice that t_{AB} may be different from t_{BA}. We use the symmetry distance as the final similarity which is represented by S_{AB} and it is the same as S_{BA}. The cost to change from A to B is

$$t_{AB} = d_{\text{courtier, diplomat}}, \tag{4.4}$$

since there is only one word in A and the minimum change is to convert it to the closest meaning word in text B, which is "diplomat".

The cost to change from B to A is

$$t_{BA} = \frac{1}{2} d_{\text{courtier, diplomat}} + \frac{1}{2} d_{\text{courtier, writer}}. \tag{4.5}$$

Notice that there are two words (after getting rid of the stop word "and") in Text B and only one word in Text A. Both "diplomat" and "writer" have to change to courtier and each of them show up once among the total sentence length.

The cost to change from from A to C is

$$t_{AC} = t_{CA} = d_{\text{courtier, writer}}. \tag{4.6}$$

. In this case, since there is only one word in each document, the distance is symmetrical.

Once we have the cost of changes, the similarity is equal to one minus the cost. Here is the final similarity between the three texts in the example.

	Courtier	Diplomat and Writer	Writer
Courtier	1	0.867	0.634
Diplomat and Writer	0.751	1	0.882
Writer	0.634	1	1

The final similarity between A and B is defined as the maximum of the similarity from both directions. The reason why we choose maximum value is that in the network approach, we do not differentiate the partial match and exact match. To match what we have done earlier, we should also treat partial match as a match here. For example, for Text B and C, as long as the similarity from C to B is 1, it should be considered as a match even though the other direction similarity is lower.

Thus, we have

$$\begin{aligned} S_{AB} = S_{BA} &= max(0.751, 0.867) = 0.867 \\ S_{AC} = S_{CA} &= max(0.634, 0.634) = 0.634 \\ S_{BC} = S_{CB} &= max(1, 0.882) = 1 \end{aligned} \tag{4.7}$$

In general, the resulting similarity makes sense. It has caught all the partial matches we used to have in the network like "diplomat and writer" and "writer" which has the highest similarity. Moreover, it also correctly indicated that Text A and B are closer than Text A and C.

Instead of using the actual values, we can also just take a soft boundary. For example, as long as the similarity is larger than 0.7, we consider the texts to be similar enough to add another edge into the historical significance network in Figure 4.5. We will not explore this direction in the following discussion.

Even though using NLP tools can help to solve the problem of synonyms in the text, it largely relies on the accuracy of the dictionary. Although we can always extract definitions for words through a formal dictionary, the meaning and context of the words may change over time. Thus, for historical research data, it is highly likely that it requires certain labeling input in the beginning. On the other hand, once the dictionary is created, it will be much easier to extract the synonyms and compare text entries.

4.2.3 Simulation

To illustrate our ideas that use the network connecting people's identities, we use the real data example in Section 3.2.2. We will compare five cases here:

- No penalty adjustment on the historical significance
- Penalty adjusted with three selected occupation groups. Each occupation has one separate covariate.
- Penalty adjusted with all manually labeled occupations. One binary covariate represents whether two people share the same historical significance entry.
- Penalty adjusted with the occupation network. One continuous covariate represents the distance between two people's historical significance. The distance is represented by the length of the shortest path between two historical significance roles.
- Penalty adjusted with NLP tools. One continuous covariate represents the distance between two people's historical significance. The distance is represented by the Word mover's distance between the text.

We report both the number of links and the precision. Notice that we only use partial data to generate the network, thus it is hard to compare the precision and recall with the SDFB network which was generated by nearly 20000 documents and 100 document-by-person matrices. Thus, we only focus on the relative comparison of the precision values. The numbers are compared through Wikipedia. For each estimated link, we script each person's Wikipedia page through the R package `WikipediR` with their search name and check whether the other one's name has appeared. All results are summarized in Table 4.1.

	No. of links	Precision (%)
No occupation	142	19.2
Selected occupation	149	22.8
Full occupation	173	24.3
Occupation network approach	130	24.6
WMD approach	161	20.5

Table 4.1: The precision and number of links being identified with each approaches to create covaraites to represent historical significance.

Simply from the numeric results, it seems that the network approach does provide a better reconstruction with high precision. However, Wikipedia is not an official record thus the numbers may not be fully accurate. Therefore, we also check the links that are identified by one model but not the other.

First, we compare the full occupation approach and occupation network approach. Table 4.2 presents the only three links that are identified by the occupation network approach but not by the full occupation

approach. Two of the three links are cases of a partial match which indicates that only considering whether the occupation is an exact match maybe not be enough.

Name 1	Name 2	Path
Thomas Wyatt soldier and rebel	Adrian Poynings soldier	1
Elizabeth Bowes protestant exile	George Bowes soldier and rebel	3
Simon Renard diplomat	Antoine de Noailles soldier and diplomat	1

Table 4.2: Sample links that identified by occupation network approach but not the full occupation approach

On the contrary, 21 links have been selected by the full occupation approach but not by the occupation network approach. We select several sample links in Table 4.3. These are the links that do not have common words in their historical significance, thus for either approach, they are hard to detect. However, the $\hat{\alpha}$ for historical significance in the network approach has a much larger scale compared to the one in the manually labeled approach (1.6 vs 0.3). Relatively speaking, not sharing a common word in the text may tend to get a heavier penalty with the occupation network approach thus may lead to the model missing those links.

Name 1	Name 2	Path
Henry Howard clergyman	Hadrianus Junius humanist and diplomat	3
King Edward king of England and Ireland	Lady Jane Grey noblewoman and claimant to the English throne	2
Thomas Wynter clergyman	Richard Morison humanist and diplomat	4

Table 4.3: Sample links that identified by full occupation approach but not the occupation network approach

We also compare the network approach and the WMD approach. In Table 4.4, we have a sample of the links that are identified by the WMD model but not by the network model. A common characteristic for those links is that even though there are no common words across their historical significance, the occupations do have a certain connection. WMD models catch the similarity while the occupation network approach does not.

On the other hand, the links identified by the occupation network approach but not by the WMD approach, some of which are shown in Table 4.5, are harder to interpret. It is possible that in some cases, WMD fails to catch the similarity. For example, for the first pair, Jane Seymour and Anne Boleyn, their historical significance roles are almost identical but WMD still believes they are not equal, as indicated by a WMD score lower than 1. Note that the occupation network approach still has a much lower $\hat{\alpha}$ value compared to the WMD approach (-1.6 vs. -0.5). The lower $\hat{\alpha}$ value in the occupation network approach

Name 1	Name 2	WMD	Path
Patrick Hepburn magnate	James Stewart nobleman	0.779	3
Bartholomew Traheron protestant writer and reformer	Richard Tracy religious activist	0.601	2
Steven Mierdman printer and bookseller	Francisco de Enzinas humanist scholar	0.306	4

Table 4.4: Sample links that identified by WMD approach but not the network approach

thus leads to a stronger decrease in the penalty on edges between pairs of individuals who have similar historical significance roles, and thus these individuals are more likely to be linked in the network approach.

Name 1	Name 2	WMD	Path
Jane Seymour queen of England, third consort of Henry VIII	Anne Boleyn queen of England, second consort of Henry VIII	0.855	1
Anne Askew writer and protestant martyr	Katherine Parr queen of England and Ireland sixth consort of Henry VIII	0.280	2
Hugh Paulet soldier and administrator	Adrian Poynings soldier	1	1

Table 4.5: Sample links that identified by network approach but not WMD approach

Finally, we notice that some people frequently appear in the above-mention comparison, like King Edward and Anne Boleyn. It seems that there are strong inconsistencies between approaches when it comes to identifying the links involving these people. For example, both the full occupation approach and the occupation network approach pick more than 4 links for King Edward, while the WMD approach picks none. On the other hand, both the full occupation approach and the WMD approach pick exactly 1 link for Anne Boleyn, while the network approach picks 4. In general, it seems that the network approach tends to pick more links among the same set of people while the other two cover more candidates. For people with relatively long historical significance texts, it is likely that with the occupation network approach, they are more likely to be connected with other identities (as long as one word overlaps, they may get a high similarity score), resulting in a decrease on the penalties on the corresponding links.

Chapter 5

Conclusions

In this thesis, we first have compared different graphical models (the local Poisson graphical model, the Gaussian graphical model, and the Poisson log-normal model) reconstructing a social network based on name co-mention counts. All the models have advantages and disadvantages but as a local estimated model, the local Poisson graphical lasso model does perform better on both AUC and average of precision and recall across different network settings. Moreover, we anticipate that with the increasing dimension, the local model should be less time-consuming than the global models. Also, we find that the network structure also affects model performance. If the degree distribution of a network is highly right-skewed, then the local Poisson graphical lasso model will significantly outperform its competitors. If a network has clear clustering structures, then the local Poisson graphical lasso model tends to perform slightly worse.

Next, we proposed the idea to include covariates information to improve the estimation of social networks from text data using a local Poisson graphical lasso model. The covariate information is incorporated through the L1 penalty: we penalize the parameters representing the edges between two individuals depending on the extent to which they have covariates in common. We use the penalty factor α to represents how the common covariates affect the penalties. To estimate the penalty factors, we have discussed two approaches: a greedy algorithm and a Bayesian framework. Both simulation and real data examples are implemented to show the validation of the approach given binary covariates. We have also discussed the possibility of replacing the double approximation in the Bayesian framework with the Laplace approximation. Even though the Laplace approximation does provides a faster solution and leads to an improvement in the model performance compares to the model without penalty factors, the average precision and recall are still worse than what we can achieve with the double approximation. Also, the scale of $\hat{\alpha}$ is harder to interpret since it does not fully align with the group densities as with double approximation. We also perform a further exploration on the effect of limiting coefficients to be non-negative. We find that even though including the edge positivity constraints helps us to avoid removing links with negative coefficients in a post-processing

step, it does not help us to improve the overall performance of the network reconstruction, particularly for the Bayesian approach.

Finally, we extend the methodology to include continuous covariates and discuss several applications about how to generate continuous covariates from last name and short summary text. The simulation study indicates that for the last name, if the name records are fuzzy with typos, including record linkage techniques to add a soft boundary to the name match will be helpful. For short paragraphs and phrases, we can use network distance and NLP tools, like Word Mover Distances (WMD) to measure their closeness. We find that using network distance (the length of the shortst path between nodes) may be able to improve the precision of network reconstruction. Even though the recall may be lower than the manually coded group, the methodology is fully automatic and thus can be easily applied to a large data set. On the other hand, even though WMD can help to pick up some edge cases where people use different words to describe similar occupations or identities, in general, it brings more noise into the covariates thus leading to low performance.

There is some potential future development of this work. In the Bayesian approach, we have tried both the double approximation and the Laplace approximation when trying to estimate the marginal likelihood of α. However, both approaches assume that the posterior has uni-modal Gaussian shape and we have not analytically evaluated the effect of both approximations. Even though the simulation study yields that the penalty factor estimated by both approximations gives results comparable to those for the greedy approach, we may want to look for other approximation methods that maintain the nature of Poisson data.

As an extension of the above-mentioned point, if we want to constrain the edge parameters to be non-negative, for the Bayesian approach, we may consider a gamma prior with shape parameter equal to 1 instead of Laplace prior. In that case, even if we want to approximate the prior distribution by a normal distribution, we may need to consider a normal distribution with a non-zero mean. Moreover, in this thesis, we start with a document-by-person matrix but in fact, converting the raw text to a numerical matrix is much more complicated than simply counting names. People can be mentioned by partial names, like only their first name or last name in the text. Also, in historical text, it is common to have people with the same name. The machine learning tools like NER that used in SDFB cannot fully distinguish and distribute the names accurately enough. Even though we can add covariates to models to indicate whether there are multiple people with the same name, it is hard to generate one single accurate document-by-person matrix in the pre-modeling process.

Also, even though we have discussed several applications of how to generate covariates from text data, there are more improvements we can consider. For the last name, even though we have shown the typos in a text can be caught with pre-modeling record linkage, the major misspelling problems in a historical text are usually caused by similar pronunciation. Thus, we could rely on an additional Soundex dictionary to measure the closeness of names. We did not have such data noise in the ODNB data but it may be interesting to find some other historical data source to measure how important it is to include last name similarity.

The comparison of the short phrases (e.g., about historical significance) relies a lot on how well people generate them originally. If we only would have the full biographies or other longer historical text data, it would be interesting to explore how to automatically extract information like occupation or identities. If we choose not to extract the information, would comparing the full text lead to a good measure of similarity between people? In that case, we may even have another approach that does not use a graphical model to determine whether two people know each other. For example, if we can decide that both people's biographies have discussed a common historical event, we may consider that they are linked.

In the thesis, we only have talked about how to add covariates to the model. However, it is possible that not all the covariates are necessary and including too many covariates may lead to large computational complexity. In the context of the Bayesian approach with double approximation, we observe that the scales of the penalty parameters indicate how important the covariates are in affecting the penalty. However, we did not give an analytical solution to how large the scale has to be so that the covariates are necessary. Thus, it would be relevant to develop a test to decide whether a covariate is useful or not. Such a test may also tell us more about how people's characteristics affect their connections.

Finally, we have applied the model to a subset of the SDFB data. The complete SDFB data contain over 19,000 documents with references to over 13,000 people. Both approaches we have proposed to estimate the penalty factor, and especially the Bayesian approach, will be slow when dealing with large data. Considering other optimization approaches to improve computational efficiency is an interesting avenue for future research.

Bibliography

Aitchison, J. and Ho, C. (1989). The multivariate poisson-log normal distribution. *Biometrika*, 76(4):643–653. 11

Allen, G. I. and Liu, Z. (2012). A log-linear graphical model for inferring genetic networks from high-throughput sequencing data. In *2012 IEEE International Conference on Bioinformatics and Biomedicine*, pages 1–6. IEEE. 6, 9, 15

Almquist, Z. W. and Bagozzi, B. E. (2019). Using radical environmentalist texts to uncover network structure and network features. *Sociological Methods & Research*, 48(4):905–960. 2

Backhouse, R. E. (2007). Economists in cambridge: a study through their correspondence, 1907–1946. 1

Bartlett, M. S. and Kendall, D. (1946). The statistical analysis of variance-heterogeneity and the logarithmic transformation. *Supplement to the Journal of the Royal Statistical Society*, 8(1):128–138. 76

Bonato, A., D'Angelo, D. R., Elenberg, E. R., Gleich, D. F., and Hou, Y. (2016). Mining and modeling character networks. In *International workshop on algorithms and models for the web-graph*, pages 100–114. Springer. 2

Boulesteix, A.-L., De Bin, R., Jiang, X., and Fuchs, M. (2017). IPF-LASSO: integrative-penalized regression with penalty factors for prediction based on multi-omics data. *Computational and Mathematical Methods in Medicine*, 2017. 21, 22, 25

Butler, R. W. (2007). *Saddlepoint approximations with applications*, volume 22. Cambridge University Press. 36

Calvo-Armengol, A. and Jackson, M. O. (2004). The effects of social networks on employment and inequality. *American Economic Review*, 94(3):426–454. 7

Chan, A. B. and Vasconcelos, N. (2009). Bayesian Poisson regression for crowd counting. In *2009 IEEE 12th international conference on computer vision*, pages 545–551. IEEE. 26, 76

Chiquet, J., Mariadassou, M., and Robin, S. (2018). Variational inference for sparse network reconstruction from count data. *arXiv preprint arXiv:1806.03120*. 12

Choi, Y., Coram, M., Peng, J., and Tang, H. (2017). A poisson log-normal model for constructing gene covariation network using rna-seq data. *Journal of Computational Biology*, 24(7):721–731. 11

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics. 4

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441. 11

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1. 10, 24

Johansen, T. P. (2017). *Diffusing Useful Knowledge: Science, Economy, and Print in the 1830s England*. PhD thesis, Aarhus University, Department for Philosophy and History of Ideas. 1

Karlis, D. (2003). An em algorithm for multivariate poisson distribution and related models. *Journal of Applied Statistics*, 30(1):63–77. 15

Kossinets, G. and Watts, D. J. (2009). Origins of homophily in an evolving social network. *American Journal of Sociology*, 115(2):405–450. 7

Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. (2015). From word embeddings to document distances. In *International conference on machine learning*, pages 957–966. 60

Lauritzen, S. L. (1996). *Graphical models*, volume 17. Clarendon Press. 10

Li, Y., Nan, B., and Zhu, J. (2015). Multivariate sparse group lasso for the multivariate multiple linear regression with an arbitrary group structure. *Biometrics*, 71(2):354–363. 21

Marsden, P. V. (1990). Network data and measurement. *Annual Review of Sociology*, 16(1):435–463. 2

McPherson, J. M. and Smith-Lovin, L. (1982). Women and weak ties: Differences by sex in the size of voluntary organizations. *American Journal of Sociology*, 87(4):883–904. 7

McPherson, M., Smith-Lovin, L., and Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444. 6, 21

Medjedović, J. (2021). Human life histories as dynamic networks: using network analysis to conceptualize and analyze life history data. *Evolutionary Psychological Science*, 7(1):76–90. 1

Mohamed, Z. T. (2020). *Studies in Early Modern Social Networks, 1400-1750*. PhD thesis, Harvard University. 7

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543. 60

Prentice, R. L. (1974). A log gamma model and its maximum likelihood estimation. *Biometrika*, 61(3):539–544. 76

Sinclair, D. and Hooker, G. (2019). Sparse inverse covariance estimation for high-throughput microrna sequencing data in the poisson log-normal graphical model. *Journal of Statistical Computation and Simulation*, 89(16):3105–3117. 12

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288. 26

Üsdiken, B. and Pasadeos, Y. (1995). Organizational analysis in North America and Europe: A comparison of co-citation networks. *Organization Studies*, 16(3):503–526. 2

Warren, C. N., Shore, D., Otis, J., Wang, L., Finegold, M., and Shalizi, C. (2016). Six degrees of Francis Bacon: A statistical method for reconstructing large historical social networks. *Digital Humanities Quarterly*, 10(3). 2, 3, 6, 24, 30

Winkler, W. E. (1990). String comparator metrics and enhanced decision rules in the Fellegi-Sunter model of record linkage. In *Proceedings of the Section on Survey Research Methods*, pages 354–359. American Statistical Association. 23

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67. 21

Zeng, C., Thomas, D. C., and Lewinger, J. P. (2020). Incorporating prior knowledge into regularized regression. *bioRxiv*. 21, 22, 26, 27, 77

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429. 21

Appendix

Appendix A

Additional algorithms, derivations and results

A.1 The algorithm of greedy method

Input : document-by-person matrix Y, covariate matrices Z^{j*} for $j = 1, \ldots, p$, function MSE(α) evaluating the MSE for the model with penalty α, search range $[a_h, b_h]$ for $h = 1, \ldots, m$, grid size d

Output: $\hat{\alpha} = [\hat{\alpha}_1, \ldots, \hat{\alpha}_m]$

Initialization: $\hat{\alpha} \leftarrow [0, 0, ...0]$, repeat $\leftarrow$ true, $MSE_{old} \leftarrow MSE(\hat{\alpha})$, $MSE_{new} \leftarrow \infty$

```
while repeat do
    repeat ← false
    order ← a random permutation of 1 to m
    for ( s ← 1 to m by 1 ) do
        h ← order[s]
        for ( α̂*_h ← a_h to b_h by d ) do
            α̂* ← [α̂_1, ..., α̂*_h, ..., α̂_m]
            MSE_new ← MSE(α̂*)
            if MSE_new < MSE_old then
                MSE_old ← MSE_new
                α̂ ← α̂*
                repeat ← true
            end
        end
    end
end
```

Algorithm 1: Greedy algorithm to estimate $\hat{\alpha}$

A.2 Approximating the marginal likelihood $L_j(\alpha)$

As mentioned in Equation (2.1), we model the number of times person j appears in document i using Poisson regression,

$$Y_{ij} \mid Y_{i,\neq j} = y_{i,\neq j}, \theta, \Theta \sim \text{Poisson}(e^{\lambda(y_{i,\neq j})}), \tag{A.1}$$

where

$$\lambda(y_{i,\neq j}) = \theta_j + \sum_{k \neq j} y_{ik}\Theta_{kj}, \tag{A.2}$$

with a covariate-depedent Lasso penalty on the Θ_{kj} or, equivalently, a Laplace prior (see expression (3.3)). To estimate the values of penalty parameters α in a Bayesian framework, we here approximate their marginal likelihood.

We first approximate the Poisson likelihood by a normal distribution, using the log-gamma approximation (Bartlett and Kendall, 1946; Prentice, 1974; Chan and Vasconcelos, 2009). Recall a Gamma random variable $\mu \sim \text{Gamma}(a, b)$ with distribution

$$p(\mu \mid a, b) = \frac{1}{\Gamma(a)b^a}\mu^{a-1}\exp^{-\frac{\mu}{b}}, \tag{A.3}$$

then the transformed random variable $\log(\mu)$ has a log-gamma distribution and for large a, the log-gamma distribution is approximately to $\mathcal{N}(\log(a) + \log(b), a^{-1})$. Let $b = 1$ and $a \in \mathcal{Z}^+$, and let $\eta = \log(\mu)$ then we have

$$\begin{aligned} p(\eta \mid a, 1) &= p(\mu = \exp^\eta \mid a, 1) \times \frac{\partial}{\partial \eta}\exp^\eta \\ &= \frac{1}{(a-1)!}\exp^{\eta a}\exp^{-\exp^\eta} \\ &\approx \text{G}(\eta \mid \log(a), a^{-1}), \end{aligned} \tag{A.4}$$

where $G(x|\mu, \Sigma) = (2\pi)^{-d/2}|\Sigma|^{-1/2}\exp(\frac{1}{2}||x-\mu||^2_\Sigma)$ is the equation of a multivariate Gaussian distribution, $||x||^2_\Sigma = x^T\Sigma^{-1}x$.

Since our data y are often sparse, to avoid $\log(0)$ in the remainder of this derivation, we add 1 to all response values y_{ij} and define $y^*_{ij} = y_{ij} + 1$. Using the approximation derived in equation (A.4), we find that

$$\frac{1}{(y^*_{ij}-1)!}e^{\lambda(y_{i,\neq j})y^*_{ij}}e^{-e^{\lambda(y_{i,\neq j})}} \approx \text{G}\left(\lambda(y_{i,\neq j}) \mid \log(y^*_{ij}), \frac{1}{y^*_{ij}}\right). \tag{A.5}$$

Then the Poisson likelihood can be written as

$$\begin{aligned}\prod_{i=1}^{n} p(Y_{ij}^* \mid Y_{i,\neq j} = y_{i,\neq j}, \Theta_j) &= \prod_{i=1}^{n} \frac{1}{y_{ij}^*!} e^{\lambda(y_{i,\neq j}) y_{ij}^*} e^{-e^{\lambda(y_{i,\neq j})}} \\ &\approx \prod_{i=1}^{n} \mathrm{G}\left(\lambda(y_{i,\neq j}) \mid \log(y_{ij}^*), \frac{1}{y_{ij}^*}\right)\end{aligned} \tag{A.6}$$

Since the variance $1/y_{ij}^*$ is likely to be similar across the documents i, we set $\hat{\sigma}_j^2 = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{y_{ij}^*}$.

At this point, we specify the Laplace prior defined in equation (3.3) by

$$\Theta_{kj} \sim \text{Laplace}\left(0, \frac{\rho_{kj}}{2\sigma_j^2}\right). \tag{A.7}$$

We approximate this Laplace prior by a normal distribution with the same variance (Zeng et al., 2020). That is, we can approximate $\beta_j \sim \mathcal{N}(0, \frac{2}{\tau_j^2})$ by $\beta_j \sim \text{Laplace}(0, \tau_j)$. Therefore, we can approximate the distribution of edge parameters $\Theta_{\neq j,j}$ by

$$\Theta_{\neq j,j} \sim \mathcal{N}(0, V^j) \tag{A.8}$$

where $V^j \in \mathbb{R}^{(p-1)\times(p-1)}$ is a diagonal matrix with $V_{kk}^j = \frac{2\sigma_j^2}{e^{2Z_k^{j*}\alpha}}$ in which Z_k^{j*} is the kth row of the covariate matrix Z^{j*}. Now we can write out the marginal likelihood of α:

$$\begin{aligned}L_j(\alpha) &= \int_{\mathbb{R}^p} \prod_{i=1}^{n} p(Y_{ij}^* \mid Y_{i,\neq j} = y_{i,\neq j}, \Theta_j) \prod_{k\neq j} p(\Theta_{kj} \mid \alpha) \mathrm{d}\Theta_j \\ &= \int_{\mathbb{R}^p} \prod_{i=1}^{n} \frac{1}{y_{ij}^*!} e^{\lambda(y_{i,\neq j}) y_{ij}^*} e^{-e^{\lambda(y_{i,\neq j})}} \prod_{k\neq j} \frac{e^{(Z_k^{j*}\alpha)}}{4\sigma_j^2} e^{-\frac{e(Z_k^{j*}\alpha)}{2\sigma^2}|\Theta_{kj}|} \mathrm{d}\Theta_j \\ &\approx \int_{\mathbb{R}^p} \frac{|\sigma_j I_n|^{-1/2}}{(2\pi)^{\frac{N}{2}}} e^{-\frac{1}{2}||\lambda(y_{\neq j}) - \log(y_j^*)||^2_{\sigma_j I_n}} \frac{|V^j|^{-1/2}}{(2\pi)^{\frac{p}{2}}} e^{-\frac{1}{2}||\Theta_j||_{V^j}} \mathrm{d}\Theta_j.\end{aligned} \tag{A.9}$$

where $y_j^* = (y_{1j}^*, \ldots, y_{nj}^*)^\top$, $y_{\neq j}$ denotes y excluding the jth column, and within the norm, $\lambda(\cdot)$ operates on the columns of $y_{\neq j}$ and $\log(\cdot)$ is applied element-wise. Dropping terms that are not a function of Θ_j, expanding the norm term and completing the square within the integral, we obtain that the approximate marginal log-likelihood of α satisfies

$$-\, l_j(\alpha) \propto \log|C_\alpha| + \log(y_j^*)^\top C_\alpha^{-1} \log(y_j^*) \tag{A.10}$$

where $C_\alpha = \sigma_j I^2 + y_{\neq j} V^j y_{\neq j}^\top$.

A.3 Laplace approximation for a non-differentiable prior

We'd like to approximate the integral

$$\begin{aligned} L_j(\alpha) &= \int_{\mathbb{R}^p} \prod_{i=1}^{n} p(Y_{ij} \mid Y_{i,\neq j} = y_{i,\neq j}, \Theta_j) \prod_{k\neq j} p(\Theta_{kj} \mid \alpha) \mathrm{d}\Theta_j \\ &= \int_{\mathbb{R}^p} \prod_{i=1}^{n} \frac{1}{y_{ij}^*!} e^{\lambda(y_{i,\neq j}) y_{ij}^*} e^{-e^{\lambda(y_{i,\neq j})}} \prod_{k\neq j} \frac{e^{Z_k^{j*}\alpha}}{4\sigma_j^2} e^{-\frac{\exp(Z_k^{j*}\alpha)}{2\sigma^2}|\Theta_{kj}|} \mathrm{d}\Theta_j \end{aligned} \tag{A.11}$$

To simplify the notation, we let

$$g(\Theta_j) = -\frac{1}{n}\big(\sum_{i=1}^{n} - \log(Y_{ij}!) + Y_{i,\neq j}\Theta_j Y_{ij} - \exp(Y_{i,\neq j}\Theta_j)\big), \tag{A.12}$$

$$h(\Theta_j) = \prod_{k\neq j} \frac{e^{Z_k^{j*}\alpha}}{4\sigma_j^2} e^{-\frac{\exp(Z_k^{j*}\alpha)}{2\sigma^2}|\Theta_{kj}|} \tag{A.13}$$

then we have

$$L_j(\alpha) = \int_{\mathbb{R}^p} e^{-ng(\Theta_j)} h(\Theta_j) \mathrm{d}\Theta_j \tag{A.14}$$

where $h(\Theta_j)$ is not differentiable at the origin, but that it is positive, finite and continuous at the origin, and that it's twice-differentiable everywhere else. We also assume that the function g is minimized at Θ_j^*, which is far away from zero. We also define $\Delta(\Theta_j) = g(\Theta_j) - g(\Theta_j^*) \geq 0$ and we assume for a small $\delta > 0$, we have $g(\Theta_j) - g(\Theta_j^*) \leq \delta$ implies $|\Theta_j - \Theta_j^*| < \epsilon(\delta)$. Pick a $\delta > 0$ then we have

$$\begin{aligned} \int_{\mathbb{R}^p} e^{-ng(\Theta_j)} h(\Theta_j) \mathrm{d}\Theta_j &= e^{-ng(\Theta_j^*)} \int \exp(-n\Delta(\Theta)) h(\Theta_j) \mathrm{d}\Theta_j \\ &= e^{-ng(\Theta_j^*)} \Big(\int_{\Delta(\Theta_j)\geq\delta} \exp(-n\Delta(\Theta_j)) h(\Theta_j) \mathrm{d}\Theta_j + \int_{\Delta(\Theta_j)<\delta} \exp(-n\Delta(\Theta_j)) h(\Theta_j) \mathrm{d}\Theta_j \Big) \\ &\leq e^{-ng(\Theta_j^*)} \Big(e^{-n\delta} \int_{\Delta(\Theta_j)\geq\delta} h(\Theta_j) \mathrm{d}\Theta_j + \int_{\Delta(\Theta_j)<\delta} \exp(-n\Delta(\Theta_j)) h(\Theta_j) \mathrm{d}\Theta_j \Big) \\ &\leq e^{-ng(\Theta_j^*)} \Big(e^{-n\delta} + \int_{\Delta(\Theta_j)<\delta} \exp(-n\Delta(\Theta_j)) h(\Theta_j) \mathrm{d}\Theta_j \Big) \\ &= e^{-ng(\Theta_j^*)} \Big(e^{-n\delta} + \int_{|\Theta_j - \Theta_j^*|<\epsilon(\delta)} \exp(-n\Delta(\Theta_j)) h(\Theta_j) \mathrm{d}\Theta_j \Big) \end{aligned} \tag{A.15}$$

If δ is small enough, since Θ_j^* is far away from zero, then all the Θ_j such that $|\Theta_j - \Theta_j^*| < \epsilon(\delta)$ will also far away from zero, thus the domain of the new integral will satisfy the assumption of Laplace approximation, then we can get a upper bound which is

$$\int_{\mathbb{R}^p} e^{-ng(\Theta_j)} h(\Theta_j) \mathrm{d}\Theta_j \leq e^{-ng(\Theta_j)} \Big(e^{-n\delta} + (\frac{2\pi}{n})^{p/2} \frac{h(\Theta_j^*)}{|-H(g(\Theta_j^*))|^{1/2}} \Big) \tag{A.16}$$

where $H(\cdot)$ denotes the Hessian. Notice that the polynomial term will dominate the formula.

Meanwhile, we also have

$$\begin{aligned}\int_{\mathbb{R}^p} e^{-ng(\Theta_j)}h(\Theta_j)\mathrm{d}\Theta_j &= e^{-ng(\Theta_j^*)}\int \exp(-n\Delta(\Theta))h(\Theta_j)\mathrm{d}\Theta_j \\ &= e^{-ng(\Theta_j^*)}(\int_{\Delta(\Theta_j)\geq\delta}\exp(-n\Delta(\Theta_j))h(\Theta_j)\mathrm{d}\Theta_j + \int_{\Delta(\Theta_j)<\delta}\exp(-n\Delta(\Theta_j))h(\Theta_j)\mathrm{d}\Theta_j) \\ &\geq e^{-ng(\Theta_j^*)}\int_{\Delta(\Theta_j)<\delta}\exp(-n\Delta(\Theta_j))h(\Theta_j)\mathrm{d}\Theta_j\end{aligned} \tag{A.17}$$

Based on the same reason, we also have the lower bound as

$$\int_{\mathbb{R}^p} e^{-ng(\Theta_j)}h(\Theta_j)\mathrm{d}\Theta_j \geq e^{-ng(\Theta_j)}(\frac{2\pi}{n})^{p/2}\frac{h(\Theta_j^*)}{|-H(g(\Theta_j^*))|^{1/2}} \tag{A.18}$$

Therefore, by the sandwich theorem, we have

$$L_j(\alpha) = \int_{\mathbb{R}^p} e^{-ng(\Theta_j)}h(\Theta_j)\mathrm{d}\Theta_j \approx e^{-ng(\Theta_j)}(\frac{2\pi}{n})^{p/2}\frac{h(\Theta_j^*)}{|-H(g(\Theta_j^*))|^{1/2}} \tag{A.19}$$

A.4 Extra positivity constraints results

In the Section 3.4.2, we have compared multiple network characteristics with and without positivity constraints on the model edge coefficients. Follpowing are the comparison on average degree in Figure A.1 and the number of false positive links involved isolated nodes in Figure A.2 with and without positivity constrains. Both figures indicate that there is no evidence that the positivity constrains will affect the average degree or the number of false positive links involved isolated nodes in the reconstructed network.

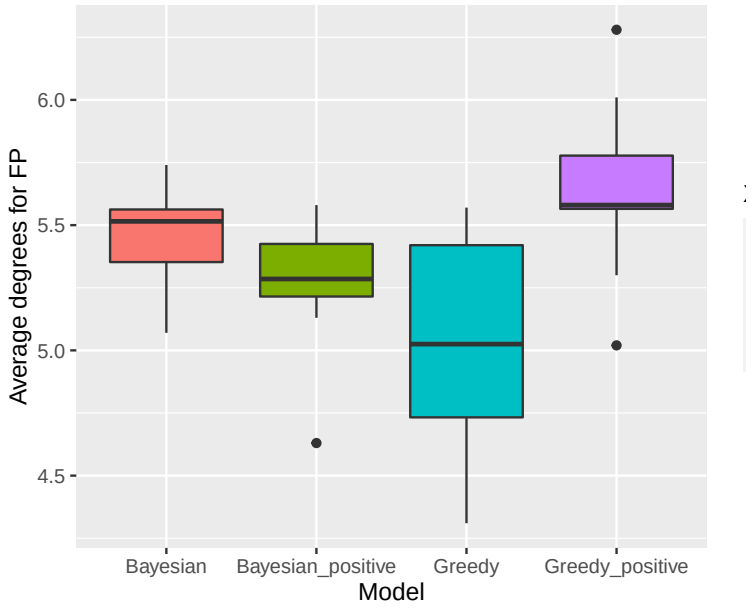

Figure A.1: The average degree in the true network for pairs of individuals incorrectly linked by one model but not the other. Even though the greedy approach with positivity constraints tends to connect with high degree nodes, the results are not consistent in Bayesian approaches. Therefore, there is no strong evidence to indicate that the positivity constraints will favor the high degree nodes.

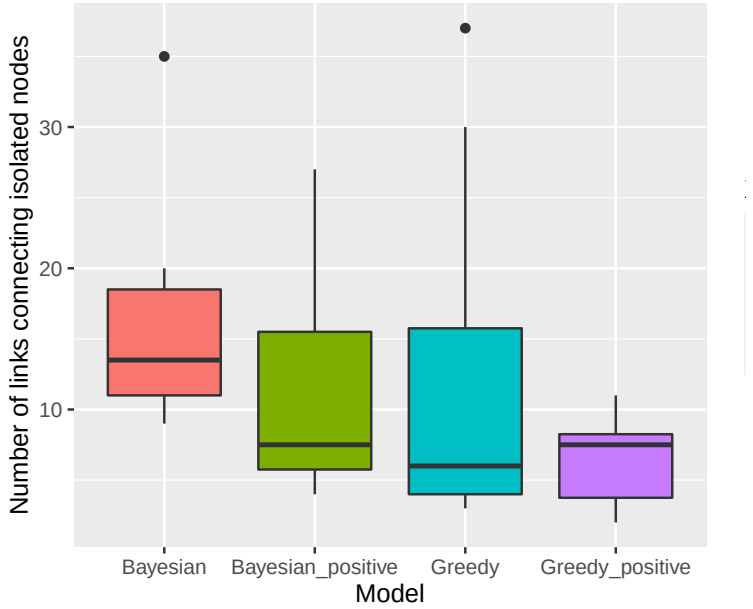

Figure A.2: The number of estimated links that involve at least one isolated node in the true network. It seems that on average the models with positivity constraints are less likely to connect the isolated nodes but the difference is not significant.

Vita

Education

Carnegie Mellon University

Department of Statistics and Data Science

Ph.D. in Statistics *Expected Summer 2021*

Master in Statistics *May 2018*

University of Wisconsin-Madison

Bachelor of Science *May 2016*

BS in Mathematics (GPA: 4.0/4.0) & BS in Statistics (GPA: 3.95/4.0)

Minor in Computer Science (GPA: 4.0/4.0)

General GPA: 3.906/4.0

Research Interests

Social Statistics, Social network reconstruction, Poisson Graphical Lasso Model, L1 penalty, Record Linkage, Statistical Education

Research Projects

Thesis: Learning a social network from text data using covariate information

(advised by N. Niezink, R. Nugent)

Historical social network analysis can help characterize the impact of influential figures but can also illuminate less well-known people who might be influential social connection points. Text data, such as biographies, can be a useful source of information to learn the structure of historical social networks but can also introduce challenges in identifying links. We extend the Local Poisson Graphical Lasso model with a (multiple) penalty structure that incorporates covariates giving increased link probabilities for people with shared covariate

information. We propose both Greedy and Bayesian approaches to estimate the penalty parameters for each potential covariate. Our primary application of interest is the Six Degrees of Francis Bacon (SDFB) project: `http://www.sixdegreesoffrancisbacon.com/`.

Assessing the resources and requirements of statistics education in forensic science
(advised by R. Mejia)
As part of a project funded by the Center for Statistics and Applications in Forensic Evidence (CSAFE), we assess the statistics requirements in accredited university programs in forensic science, through reviewing accreditation requirements, analyzing program admission requirements and curricula, and surveying heads of Forensic science departments. This pilot project aims to characterize the expectation of the American Academy of Forensic Sciences for statistics skills and their alignment with tasks performed by forensic scientists, statistics teaching resources available to forensics programs, and possible solutions for reducing any identified gaps.

Historical record linkage in Ohio state during early 20th century
(in collaboration with University of Michigan LIFE-M Project; PI: Martha Bailey, UM Economics)
My methodological work focused on modeling and estimating links between Ohio birth certificates in 1920s and Ohio 1940s census records. We developed a series of record comparison metrics in this context and a new model called "Highlander Probability Model (HPM)", which divided the traditional record linkage problems into two phases and focuses more on distinguishing the difference between highly similar entities. Similar methodology was also applied to North Carolina. *(paper in progress)*

Paper

Yang, X, Niezink, N and Nugent, R. "Learning Social Networks from Text Data using Covariate Information" Accepted at *Statistical Methods & Applications: Special Issue on Statistical Analysis of Networks*. Currently available at `http://arxiv.org/abs/2010.08076`

Conferences and workshops

- The Network Science Society 2020. Sep 17th-25th, 2020

  Learning Social Networks from Text data (Poster)

- Statistical Inference for Network Models (SINM): Sep 20th, 2020

  Learning Social Networks from Text data (Contributed Talk)

- JSM: Aug 1st - 6th, 2020

  Assessing the resources and requirements of statistics education in forensic science (Poster)

- 2020 Symposium on Data Science & Statistics: June 3rd-6th, 2020

  Learning a social network from Text data (Invited Talk)

- LIFE-M Project Board Meeting: June 1st-2nd, 2017 and Oct 25th - 26th, 2018

  Presentations on current work to LIFE-M collaborators and advisory board

- Working Group on Model-Based Clustering Summer Session: Ann Arbor, July 15th - 20th, 2018

  Historical Record Linkage with Highlander Probability Model (Poster)

Teaching and Mentoring Experience

- **Lecturer**
  - Summer 2020: 36-225, Introduction to Probability Theory, Online course with over 140 students, one of two instructors
- **Data Science Initiative (DSI) Fellow**
  - Spring 2020: Advising a group of five students with clients from Chain of Demand (chainofdemand.co) to analyze and understand fashion trend through social media and text analysis
  - Fall 2019: Advising a group of four students with clients from Ikos (ikos.rent) to understand the relation between housing properties, rent value and rent time.
  - Spring 2019: Advising two students on project with Christopher Warren, Department of English on characterizing the difference between the Shakespeare-related biographies and other biographies at the same time, mainly on the use of words and phrases.
- **Summer Undergraduate Research Experience**
  - Summer 2019 Carnegie Mellon Sports Analytics Camp: Advising a group of four students on project with Kostas Pelechrinis, School of Computing and Information, University of Pittsburgh to analysis and modify the penalty area in soccer games to promote fairness.
- **Curriculum Design**
- Incorporating e-learning into statistics-related courses to provide in-lecture interaction and data accessibility with ISLE, a browser-based interactive statistics & data analysis platform.
  See `http://www.stat.cmu.edu/isle/`

  - UC-Irvine, CRM/LAW C132 Forensic Science, Law, and Society, instructed by Professor Simon A. Cole: help to create interactive statistics modules for the course
  - CMU, ENG 76107 - Writing about Data, instructed by Professor David Brown: help to assess whether the statistics components, like data structure are reasonably presented

- **Teaching Assistant Experience**

- General TA duties include grading and office hours. Head TA also supervises duties for other TAs (401) and helps with the course projects and in-lecture discussion (303/311).
  - Fall 2019: 36-311, Statistical Analysis of Networks (Head TA)
  - Spring 2019: 36-303, Sample, Survey and Society (Head TA)
  - Fall 2018: 36-401, Modern Regression (Head TA)
  - Fall 2017: 36-461/661, Special Topics: Epidemiology
  - Summer 2017: 36-201, Statistical Reasoning and Practice
  - Spring 2017: 36-217, Probability Theory and Random Processes
  - Fall 2016: 36-217, Probability Theory and Random Processes