Statistical Latent Space Models for International Classification of Diseases (ICD) Codes

by

Cheng Ma

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Statistics)
in the University of Michigan
2024

Doctoral Committee:

Professor Ji Zhu, Chair
Professor Judy Jin
Professor Liza Levina
Professor Kerby Shedden

Cheng Ma
chengmc@umich.edu
ORCID iD: 0009-0006-0558-1686

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

LIST OF APPENDICES

APPENDIX

ABSTRACT

The increasingly widespread use of Electronic Health Records (EHRs) offers significant opportunities to improve patient care insights and inspire extensive healthcare research. The International Classification of Diseases (ICD) codes, a crucial component of EHR data, have attracted significant research interest due to their potential to improve clinical decision-making. This dissertation focuses on developing novel statistical models for ICD code embedding and exploring their applications.

Most existing healthcare research works borrow word embedding techniques from natural language processing (NLP) and apply them to ICD codes. However, significant differences in the structure, meaning, and usage exist between ICD codes and natural language words, making word embedding methods not entirely suitable for modeling ICD codes. The first part of this dissertation proposes a new latent space zero-inflated Poisson model to characterize the co-occurrence of ICD codes. This model associates each ICD code with a latent vector and assumes the co-occurrence of two ICD codes depends on the relative positions of the corresponding latent vectors. By utilizing a zero-inflated Poisson distribution, the proposed model effectively addresses the abundant zeros commonly observed in practice. Theoretically, we establish error bounds for the estimation of the latent vectors. Furthermore, we demonstrate the effectiveness of our model using the MIMIC-III EHR dataset, showing that the learned latent vectors are useful predictors for downstream tasks. This indicates that our model has the potential to improve patient outcome predictions and advance EHR-based research.

Designed in the 1970s, the Ninth Revision of ICD (ICD-9) no longer meets the medical needs of healthcare providers and patients. In October 2015, hospitals in the United

States transitioned from ICD-9 to ICD-10 codes. Consequently, the healthcare domain faces challenges in transferring and merging historical data and applications to this new system. Addressing this, the second part of the dissertation proposes a joint latent space zero-inflated Poisson model that learns embeddings for both versions of ICD codes simultaneously, as well as a transformation that maps ICD-9 codes to the newer system. To demonstrate the practical value of the model, we design an ICD code translation task using the Nationwide Readmissions Database (NRD) and show that our proposed model outperforms all existing approaches in this task.

Although the latent space zero-inflated Poisson model has proven effective in modeling ICD codes, focusing solely on pairwise co-occurrences may overlook higher-order information among ICD codes. In the third project, we treat ICD codes in EHR as a hypergraph, where the set of ICD codes in a medical record forms a hyperedge. Specifically, we consider a latent space model based on the determinantal point process for hypergraphs. Direct estimation of parameters using the likelihood function is however numerically unstable. To overcome this, we develop an algorithm based on a mixture of the ordinary likelihood and the pseudo-likelihood. This proposed algorithm is shown to be more stable compared to using the ordinary likelihood alone, particularly when the number of hyperedges is not large. We apply this approach to a readmission prediction task on the MIMIC-III EHR dataset and demonstrate its practical value. We also establish theoretical guarantees for the consistency of the mixed likelihood estimation.

CHAPTER 1

Introduction

Over the past two decades, there has been an exponential increase in the volume of electronic information stored in electronic health records (EHRs) (Shickel et al., 2017). Originally designed for collecting and storing healthcare information, EHRs have attracted considerable research interest. Their applications include but are not limited to: EHR information extraction (Wu et al., 2015; Fries, 2016), representation learning (Tran et al., 2015; Lv et al., 2016; Choi et al., 2017b), and outcome prediction (Miotto et al., 2016; Choi et al., 2016). Among the various types of data included in EHRs, the International Statistical Classification of Diseases and Related Health Problems (ICD) is a crucial component.

The International Classification of Diseases (ICD) is a medical classification system containing codes for diseases and medical conditions. There are over 13,000 different codes in the Ninth Revision of the ICD (ICD-9). Each ICD code represents a specific disease or medical condition (e.g., the ICD-9 code 250.00 represents "Diabetes mellitus without mention of complication, type II or unspecified type, not stated as uncontrolled"). While primarily designed for recording and billing purposes, the structured and standardized format of ICD codes and their low missing rate make ICD codes very helpful for analyzing EHR data. However, the large number of different ICD codes makes it challenging to utilize ICD codes in statistical models directly. A common approach to deal with this issue is to map ICD codes into a low-dimensional space, and most existing works borrow word embedding models from natural language processing (Shi et al., 2021; Choi et al., 2017b; Cai et al., 2018). However,

there exist significant differences in the structure, meaning, and usage of ICD codes and natural language words. For instance, words in context have order, but ICD codes in a medical record are usually treated as a set. In addition, while natural language words can be ambiguous, ICD codes are designed to be precise and specific. Because of these differences, word embedding models can not be applied to ICD codes directly and are arbitrarily modified to fit ICD codes. Consequently, word embedding models designed for the unique properties of natural language words may not be entirely suitable for modeling ICD codes. As a result, ICD code embedding models that are designed for ICD codes specifically are needed.

In 2015, hospitals in the United States changed from the Ninth Revision of the ICD (ICD-9) to the Tenth Revision of the ICD (ICD-10), resulting in significant gaps in electronic health record (EHR) data. There are many articles discussing the challenges and problems associated with this transition (Topaz et al., 2013; Khera et al., 2018; Hamedani et al., 2021). A primary challenge for researchers and healthcare providers is mapping between the two versions of ICD codes. The General Equivalence Mappings (GEM) (Butler, 2007) provides a mapping between the two versions of ICD codes. However, only a part of the codes have a one-to-one conversion from ICD-9 to ICD-10, some other ICD-9 codes are mapped to multiple ICD-10 codes. In addition, multiple different ICD-9 codes could be mapped to a single ICD-10 code (Hamedani et al., 2021). This leads to difficulties in merging EHRs with ICD-9 codes to the new system. Additionally, the transition has affected the usage of ICD codes. For instance, research finds the monthly prevalence of the ICD code for subarachnoid hemorrhage was stable before the transition, and became increased after the transition. However, the actual disease prevalence remained stable (Hamedani et al., 2021). Furthermore, the prevalence of 6 out of 16 neurologic diagnoses (37.5%) experienced significant changes after the transition. Clearly, the transition affects the usage of ICD codes and could cause the wrong estimation of disease prevalence. This situation has led to recommendations for studies to limit themselves to a single ICD coding system, potentially

resulting in the loss of many samples. Thus, the development of an effective ICD code mapping model becomes necessary.

This dissertation addresses the above problems and challenges respectively. In Chapter 2, we consider the ICD code embedding problem. Based on the network latent space models (Hoff et al., 2002; Hoff, 2003; Ma et al., 2020), we propose a latent space zero-inflated Poisson model that associates ICD codes with latent positions. The proposed model assumes the co-occurrence time of two ICD codes follows a zero-inflated Poisson distribution (Lambert, 1992). Parameters of the zero-inflated Poisson distribution are determined by the relative positions of the corresponding latent vectors. Compared with a simple Poisson distribution, the zero-inflated Poisson distribution handles the abundant zeros in the co-occurrence matrix better (Loeys et al., 2012). However, the introduction of the zero-inflated Poisson distribution also leads to difficulty in providing theoretical guarantees for the latent space zero-inflated Poisson model. Existing works on similar models rely on the great property of the exponential family (Ma et al., 2020; Zhang et al., 2022). The zero-inflated Poisson distribution does not belong to the exponential family, and the existing approaches cannot be applied to the proposed model. To overcome the difficulties, we developed error bounds of estimation for the proposed model by using the Peano reminder of the Taylor Theorem. To show the effectiveness of the proposed model in real-world applications, we design a readmission prediction task using the MIMIC-III EHR dataset (Johnson et al., 2016) and apply the proposed model to it. The result demonstrates that the latent positions learned by the proposed model are informative and can serve as effective predictors. We compare the proposed model with the popular word embedding models, such as Skip-gram (Mikolov et al., 2013) and GloVe (Pennington et al., 2014), and the proposed latent space zero-inflated Poisson model outperforms all other models.

In Chapter 3, we address the problem of lacking an ICD code mapping model. We propose a joint latent space zero-inflated Poisson model. The model consists of two latent space zero-inflated Poisson models and a linear transformation. The two latent space zero-

inflated Poisson models learn embeddings for ICD-9 and ICD-10 codes respectively. The linear transformation is learned to align the embeddings of the two types of codes using a high-quality code dictionary. The code dictionary contains a small amount of high-quality matched code pairs. Each pair of matched codes has one ICD-9 code and one ICD-10 code with an identical medical meaning. The dictionary is generated using the ICD General Equivalence Mappings (GEM) (Butler, 2007). While the two latent space models update the ICD code embedding, the transformation aligns the embeddings of each pair of matched ICD codes. Considering that the relationships between diseases are independent of the ICD coding system, the relative positional relationships of the medical code embeddings should similarly remain unchanged. Therefore, aligning the embeddings of matched ICD codes in the dictionary also leads to the alignment of other ICD codes with similar medical meanings. The model makes it possible to integrate data and applications using ICD-9 codes with the newer ICD-10 system. We designed an ICD code translation task with the Nationwide Readmissions Database (NRD), which has both medical records with ICD-9 codes and those with ICD-10 codes. Our proposed model has the best performance among all approaches, demonstrating its practical value.

Although the latent space zero-inflated Poisson model has demonstrated its effectiveness in the ICD code embedding problem, it was developed solely from pairwise co-occurrences of ICD codes. Popular word embedding methods, such as GloVe (Pennington et al., 2014) and Skip-gram (Mikolov et al., 2013), are also trained using pairwise information. Focusing solely on pairwise information can result in the loss of higher-order information on the relationships and interactions among groups of more than two diseases and medical conditions. In Chapter 4, we treat ICD codes as a hypergraph, where the set of ICD codes in the same record forms a hyperedge. Specifically, we consider a latent space model based on the determinantal point process (Kulesza et al., 2012) for hypergraphs. Direct estimation of parameters using the likelihood function as proposed in Yu and Zhu (2023) is however numerically unstable. To overcome this, we introduce the mixed likelihood, which is a combination of the ordinary

likelihood and the pseudo-likelihood. We design an algorithm based on the mixed likelihood. The proposed algorithm is shown to be more stable and accurate compared to using the ordinary likelihood alone, particularly when the number of hyperedges is not large. Theoretically, we show that the pseudo-likelihood is actually the ordinary likelihood function of a sub-graph, which ensures the consistency of the mixed likelihood estimation. We apply the proposed approach to the same predictive task as in Chapter 2, and it shows enhanced performance compared to the estimation with ordinary likelihood.

CHAPTER 2

A Latent Space Zero-Inflated Poisson Model for ICD Code Embedding

2.1 Introduction

The exponential growth of electronic health records (EHR) inspired significant research interest in recent years. Statistical models have proven to be highly effective in analyzing EHR data and predicting patient outcomes, such as readmission rates and death risk, which have the potential to improve patient care. EHR data contains a diverse range of healthcare-related variables, including patient demographics, lab results, medications, and diagnoses. Among these variables, International Classification of Diseases (ICD) codes play a critical role in EHR records. Each medical record is assigned a set of ICD codes that provide essential clinic information, such as diagnoses and procedure details. Due to their well-structured format and low incidence of missing data, ICD codes serve as highly reliable variables in healthcare research.

Utilizing ICD codes directly in statistical models is challenging due to the large amount of different ICD codes. This issue is not unique to healthcare and is a problem that natural language processing researchers face as well. To address this issue, word embedding models have been proposed to map each word to a vector in a lower-dimensional space, making it easier to use various language models. Many existing works in healthcare research borrow word embedding techniques to model ICD codes, including Shi et al. (2021); Nguyen et al.

(2018); Choi et al. (2017b); Feng et al. (2017); Choi et al. (2017a); Cai et al. (2018). However, ICD codes and words have distinct characteristics. For example, words are ordered within a sentence, while ICD codes are typically considered unordered. Furthermore, natural language words can be ambiguous, while ICD codes are designed to be precise and specific. These significant differences make directly applying word embedding models to ICD codes not entirely suitable. Therefore, a more appropriate model specifically designed for ICD codes is necessary.

In this Chapter, we propose a latent space zero-inflated Poisson model, to model the co-occurrence patterns of ICD codes. Specifically, the co-occurrence time of two ICD codes is the number of times they appear in the same health record. To connect the ICD code embedding problem with network modeling, we can view each ICD code as a node and the co-occurrence time of two ICD codes as the weight of the edge connecting them. This way, the co-occurrence structure of ICD codes can be seen as a network with count-weighted edges. One of the most popular network models is the latent space model proposed in Hoff et al. (2002). This model assumes that a lower-dimensional vector in a latent space can represent each node in the network, and the latent vectors can reveal the behaviors of the nodes. For example, nodes with close latent vectors are more likely to be connected or interact with each other. The latent space model has been proven to be powerful for modeling real-world networks in various fields, including social networks, citation networks, and biological networks (Ward and Hoff, 2007; Ward et al., 2007; Friel et al., 2016). Our proposed model extends the latent space model by utilizing the zero-inflated Poisson distribution.

Like the latent space model and word embedding models, the proposed latent space zero-inflated Poisson model associates each ICD code to a vector in a lower dimension space and assumes the co-occurrence time follows a zero-inflated Poisson distribution. The parameters of the zero-inflated Poisson distribution are connected to the latent positions by an inner-product model. The latent positions learned by the proposed model can be then used as features in downstream tasks. Our case studies show that the learned vector representations

of ICD codes are clinically meaningful, as the vector representations of ICD codes for similar or associated diseases are closer in the latent space. These findings demonstrate the potential of our models to effectively capture the medical meaning of ICD codes and improve performance in downstream tasks in the healthcare field.

In real-world EHR datasets, we observed that certain ICD codes have never appeared together, resulting in abundant zeros in the co-occurrence matrix. The zero-inflated Poisson distribution in the proposed model can deal with this issue. Specifically, a zero-inflated Poisson distribution is a mixture of a Bernoulli distribution and a Poisson distribution. In the latent space zero-inflated Poisson model, the Bernoulli distribution determines if two ICD codes could appear in the same records, while the Poisson distribution determines the number of co-occurrence times (Mullahy, 1986; Cameron and Trivedi, 2013). Most works utilizing the zero-inflated Poisson distribution are done under regression settings (Rose et al., 2006; Lichman and Smyth, 2018), where well-designed covariates are needed. As far as we know, this is the first work that extends the latent space model to the zero-inflated Poisson distribution.

We also present a latent space Poisson model in this Chapter, which can be seen as a simpler version of the proposed model. Similar to the proposed model, the latent space Poisson model associates ICD codes with latent positions. However, it assumes the co-occurrence time follows a simple Poisson distribution. The main contributions of this paper are the following.

- We introduce a novel approach, the latent space zero-inflated Poisson model, to embed ICD codes into latent spaces. The proposed model focuses on modeling the co-occurrence matrix of the ICD codes. The model adopts a zero-inflated Poisson distribution to characterize the co-occurrence time. The zero-inflated Poisson distribution can model both the existence and the number of co-occurrences. The key idea of our models is to connect ICD codes with latent vectors that represent various characteristics of the codes. As a result, the latent vectors could serve as features for downstream

tasks. Additionally, we also present a simpler version of the model, the latent space Poisson model.

- We provide error bounds for estimates of the zero-inflated Poisson model. The mixture distribution in the model makes it challenging to derive theoretical results. We utilize the Peano remainder of the Taylor Theorem, instead of the commonly used Lagrange remainder (Ma et al., 2020; Zhang et al., 2022), to deal with the complicated likelihood function of the model. This allows us to obtain reliable and tight error bounds for our proposed model.

- Our proposed model learns latent vectors that capture valuable features and can be used in downstream tasks. To demonstrate the utility of these embeddings, we design a readmission prediction task using the MIMIC-III dataset (Johnson et al., 2019) and apply our proposed model to it. The results suggest that the learned embeddings improve the prediction performance compared to existing models.

The rest of the chapter is organized as follows. Section 2.2 presents the proposed models, the latent space Poisson model, and the latent space zero-inflated Poisson model, along with the fitting methods. In Section 2.3, we derive theoretical results for the estimation errors. In Section 2.4, we conduct simulation studies to investigate the effects of the number of nodes and the latent space dimension and compare the proposed method to related models. We apply the proposed method to a real-world dataset, MIMIC-III, and demonstrate its practical value in Section 2.5. Finally, Section 2.6 concludes the chapter with a discussion. For simplicity, we omit "latent space" and use just the "Poisson model" and "zero-inflated Poisson model" to refer to the proposed models when there is no ambiguity throughout the rest of the chapter.

2.2 Model

In this section, we begin by defining the notation used throughout the chapter. Then, we present the Poisson model and the proposed zero-inflated Poisson model, along with the identifiability conditions for each. Finally, we introduce two algorithms that are used to fit these models.

2.2.1 Notation

Assume that we have the co-occurrence structure of n ICD codes denoted by a symmetric co-occurrence matrix $X \in \mathbb{N}^{n\times n}$, where $X_{ij} = X_{ji}$ is the number of co-occurrences of the ith code and the jth code. We assume X_{ij} to be random variables taking values in the natural numbers $\mathbb{N} = \{0, 1, 2, ...\}$. We use $\circ$ to denote the Hadamard product (also known as the element-wise product), that is, for any two matrices $A, B \in \mathbb{R}^{d_1\times d_2}$ of the same size, $A \circ B \in \mathbb{R}^{d_1\times d_2}$ and $(A \circ B)_{ij} = A_{ij}B_{ij}$. For any matrix $A \in \mathbb{R}^{d_1\times d_2}$, let $\mathbf{I}_{A=0}$ be the indicator matrix, such that, $(\mathbf{I}_{A=0})_{ij} = 1$ if $A_{ij} = 0$, and $(\mathbf{I}_{A=0})_{ij} = 0$ otherwise. Similarly, let $\mathbf{I}_{A>0}$ be that, $(\mathbf{I}_{A>0})_{ij} = 1$ if $A_{ij} > 0$, and $(\mathbf{I}_{A>0})_{ij} = 0$ if $A_{ij} \leq 0$. We use $\|X\|_F, \|X\|_{op}, \|X\|_*$, and $\|X\|_\infty$ to denote the Frobenius norm, the operator norm, the nuclear norm, and the max norm of matrix X respectively.

2.2.2 The Latent Space Poisson Model

To model the co-occurrence matrix, the latent space Poisson model assumes the co-occurrence time follows a Poisson distribution whose parameter is connected to the latent positions of the codes by the inner-product model. Specifically, each code i is represented by a latent vector $w_i \in \mathbb{R}^q$ in a low-dimensional latent space and a degree heterogeneity parameter $\beta_i \in \mathbb{R}$. Let $W = [w_1, w_2, ..., w_n]^T \in \mathbb{R}^{n\times q}$. We consider the following inner-product latent

space model (Hoff, 2003; Ma et al., 2020), i.e., for any $i < j$,

$$X_{ij} = X_{ji} \sim \text{Poisson}(\lambda_{ij}), \tag{2.1}$$

where

$$\log(\lambda_{ij}) = w_i^T w_j + \beta_i + \beta_j = \Phi_{ij}. \tag{2.2}$$

Note that the probability mass function of the Poisson distribution with parameter λ is

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!} \text{ for } k = 0, 1, 2, ...$$

We rewrite the formula (2.2) in matrix form:

$$\log(\Lambda) = WW^T + \beta 1_n^T + 1_n^T \beta = \Phi, \tag{2.3}$$

where $\beta = (\beta_1, \beta_2, ..., \beta_n)^T$, and $\Lambda = (\lambda_{ij})_{n \times n}$.

2.2.3 The Latent Space Zero-inflated Poisson Model

We will first introduce the model structure of the latent space zero-inflated Poisson model before we talk about the models and parameters in more detail.

To account for the abundant zeros in a co-occurrence matrix, we propose the latent space zero-inflated Poisson model by combining the zero-inflated Poisson model developed by Lambert (1992) with the latent space model. Unlike the previous model, each node i is now represented by two latent vectors $v_i \in \mathbb{R}^{q_1}$ and $w_i \in \mathbb{R}^{q_2}$ in low-dimensional latent spaces. Let $V = [v_1, v_2, ..., v_n]^T \in \mathbb{R}^{n \times q_1}$ and $W = [w_1, w_2, ..., w_n]^T \in \mathbb{R}^{n \times q_2}$. We consider the

following inner-product latent space model (Hoff, 2003; Ma et al., 2020), i.e., for any $i < j$,

$$X_{ij} = X_{ji} \sim \begin{cases} 0 & \text{with probability } 1 - \pi_{ij}, \\ \text{Poisson}(\lambda_{ij}) & \text{with probability } \pi_{ij}, \end{cases} \tag{2.4}$$

where,

$$\text{logit}(\pi_{ij}) = v_i^T v_j + \alpha_i + \alpha_j = \Theta_{ij}, \tag{2.5}$$

$$\log(\lambda_{ij}) = w_i^T w_j + \beta_i + \beta_j = \Phi_{ij}. \tag{2.6}$$

The parameters α_i's and β_i's are used to model node degree heterogeneity. Specifically, a larger α_i indicates the i-th code is more likely to co-occur with other codes, while a larger β_i suggests that the co-occurrence times of the i-th code with other codes are more likely to be larger.

We rewrite the formula (2.5) and (2.6) in matrix form:

$$\text{logit}(\Pi) = VV^T + \alpha 1_n^T + 1_n^T \alpha = \Theta, \tag{2.7}$$

$$\log(\Lambda) = WW^T + \beta 1_n^T + 1_n^T \beta = \Phi, \tag{2.8}$$

where $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n)^T$, $\beta = (\beta_1, \beta_2, ..., \beta_n)^T$, $\Pi = (\pi_{ij})_{n \times n}$, and $\Lambda = (\lambda_{ij})_{n \times n}$.

The parameters (V, α) in (2.7) model the existence of co-occurred ICD code pairs, while the parameters (W, β) in (2.8) model the co-occurrence times. By including the co-occurrence times, these models are able to capture more information than models that consider only the binary co-occurrence matrix (or binary network), thereby providing a deeper understanding of the underlying relationships between the ICD codes and diseases.

The latent positions V and W can be considered as the vector representations of the ICD codes. In Section 2.5, we demonstrate that these vector representations can be used as features in downstream tasks, showing the practical value of our proposed model. One of

the advantages of our model is the flexibility to use different dimensions for V and W, which allows for more fine-grained control over the representation of the ICD codes.

Identifiability. To ensure the identifiability of the parameters, we make the same assumptions as in Ma et al. (2020). Specifically, we assume the latent vectors are centered, i.e., $J_n V = V$ and $J_n W = W$, where $J_n = I_n - \frac{1}{n} 1_n 1_n^T$. This assumption ensures that the parameters are identifiable up to orthogonal transformations. Furthermore, under this assumption, VV^T and WW^T are directly identifiable.

Relationship to related models. The parameters α's and β's are crucial components of the latent space zero-inflated Poisson model and connect it with other related models. For example, when α's are large, the corresponding π's increase, and the latent space zero-inflated Poisson model becomes similar to the latent space Poisson model. On the other hand, larger values of β's would make the model similar to the latent space Bernoulli model presented in Ma et al. (2020). As such, the latent space Poisson model and the model in Ma et al. (2020) are both special cases of the latent space zero-inflated Poisson model, which shows the flexibility of our proposed model. We will provide further details and comparisons between these models in our simulation studies and real-world examples.

2.2.4 Model Fitting

In this section, we introduce two methods for fitting the proposed models: the projected gradient descent algorithm and the EM algorithm. The projected gradient descent algorithm can be used to fit both the latent space Poisson model and the latent space zero-inflated Poisson model. On the other hand, the EM algorithm is adapted to fit the latent space zero-inflated Poisson model to handle the mixture distribution. Both approaches minimize the negative log-likelihood function. The projected gradient descent algorithm minimizes the objective function directly under the model restriction, while the EM algorithm itera-

tively infers which part of the model generates the zeros and updates the model parameters accordingly.

In the following section, we will take the latent space zero-inflated Poisson model as an example to present the fitting algorithms. The latent space Poisson model can be fit by the projected gradient descent algorithm similarly.

2.2.4.1 The Projected Gradient Descent Algorithm

We consider the negative log-likelihood function of the latent space zero-inflated Poisson model as the objective function:

$$\begin{aligned} l(\alpha, V, \beta, W) &= -\sum_{i<j=1}^{n} \log P(X_{ij}|\alpha, V, \beta, W) \\ &= -\sum_{i<j=1}^{n} (\mathbb{1}_{X_{ij}=0} \log(1 + e^{\Theta_{ij} - \exp(\Phi_{ij})}) + \mathbb{1}_{X_{ij}>0}(\Theta_{ij} + X_{ij}\Phi_{ij} - e^{\Phi_{ij}}) - \log(1 + e^{\Theta_{ij}})) \end{aligned}$$

where $\mathbb{1}_{X_{ij}=0}$ is an indicator, its value is 1 if and only if $X_{ij} = 0$. We define $\mathbb{1}_{X_{ij}>0}$ similarly.

We use the following optimization to estimate the parameters:

$$(\hat{\alpha}, \hat{V}, \hat{\beta}, \hat{W}) = \underset{\alpha,\beta \in \mathbb{R}^n, V \in \mathbb{R}^{n \times q_1}, W \in \mathbb{R}^{n \times q_2}}{\arg\min} l(\alpha, V, \beta, W),$$

where $VV^T + \alpha 1_n^T + 1_n^T \alpha = \Theta$ and $WW^T + \beta 1_n^T + 1_n^T \beta = \Phi$.

We adapt the projected gradient descent algorithm in Ma et al. (2020) to solve the optimization. In each iteration, we update the parameters along the direction that decreases the loss function. The parameters are centered after each iteration to ensure identifiability. A detailed summary of the estimation method is shown in Algorithm 1.

Algorithm 1: The projected gradient descent algorithm for estimation

Input: a count-weighted network $X_{n\times n}$, step length $(\eta_V, \eta_W, \eta_\alpha, \eta_\beta)$, initialization $(V_0, W_0, \alpha_0, \beta_0)$,number of iterations T;

Output: $(\hat{V}, \hat{W}, \hat{\alpha}, \hat{\beta})$;

for $t = 0, 1, ..., T-1$ **do**

$\tilde{V}_{t+1} = V_t - \eta_V \nabla_V l(\alpha, V, \beta, W)$;

$\tilde{W}_{t+1} = W_t - \eta_W \nabla_W l(\alpha, V, \beta, W)$;

$\alpha_{t+1} = \alpha_t - \eta_\alpha \nabla_\alpha l(\alpha, V, \beta, W)$;

$\beta_{t+1} = \beta_t - \eta_\beta \nabla_\beta l(\alpha, V, \beta, W)$;

$V_{t+1} = J_n \tilde{V}_{t+1}$ $W_{t+1} = J_n \tilde{W}_{t+1}$

end

$\hat{V}, \hat{W}, \hat{\alpha}, \hat{\beta} = V_T, W_T, \alpha_T, \beta_T$;

We adapt the choices of step size proposed in Ma et al. (2020):

$$\eta_V = \eta/\|V_0\|_F^2, \eta_\alpha = \eta/(2n),$$
$$\eta_W = \eta/\|W_0\|_F^2, \eta_\beta = \eta/(2n).$$

2.2.4.2 The EM Algorithm

Alternatively, the EM algorithm can also be used to fit the model, which is a commonly used approach in zero-inflated Poisson regression literature (Lambert, 1992; Min and Agresti, 2005; Wang et al., 2014). In our case, we adapted the EM algorithm to the latent space zero-inflated Poisson model to handle the mixture distribution better.

The mixture of Φ and Θ complicates the optimization of $l(\alpha, V, \beta, W)$. To address this issue, we need to account for the uncertainty of whether an element $X_{ij} = 0$ comes from the Poisson distribution or the zero-inflation part. To tackle this, we introduce a binary variable Z_{ij} that takes the value 1 if $X_{ij} = 0$ and it comes from the Poisson distribution or $X_{ij} > 0$. Otherwise, Z_{ij} takes the value 0. This way, we can reformulate X_{ij} as the product

of two independent random variables $Z_{ij} \sim \text{Bernoulli}(\pi_{ij})$ and $\tilde{X}_{ij} \sim \text{Poisson}(\lambda_{ij})$. This reformulation enables us to apply the EM algorithm to estimate the model parameters.

The distribution of the unobserved latent variable Z_{ij} with respect to the observed data X_{ij} and the current estimates of the parameters $(V^{(t)}, W^{(t)}, \alpha^{(t)}, \beta^{(t)})$ is:

$$\mathbf{P}(Z_{ij} = 1 | X_{ij}, V^{(t)}, W^{(t)}, \alpha^{(t)}, \beta^{(t)}) = \begin{cases} 1 & \text{if } X_{ij} > 0, \\ \frac{\pi_{ij}^{(t)} \exp(-\lambda_{ij}^{(t)})}{1 - \pi_{ij}^{(t)} + \pi_{ij}^{(t)} \exp(-\lambda_{ij}^{(t)})} & \text{if } X_{ij} = 0, \end{cases} \tag{2.9}$$

where

$$\begin{aligned} \text{logit}(\pi_{ij}^{(t)}) &= v_i^{(t)\prime} v_j^{(t)} + \alpha_i^{(t)} + \alpha_j^{(t)} = \Theta_{ij}^{(t)}, \\ \log(\lambda_{ij}^{(t)}) &= w_i^{(t)\prime} w_j^{(t)} + \beta_i^{(t)} + \beta_j^{(t)} = \Phi_{ij}^{(t)}. \end{aligned}$$

To simplify the notation, we use $\Psi = (V, W, \alpha, \beta)$ to represent all parameters in the rest of this section.

E Step. When $X_{ij} > 0$, Z_{ij} must be equal to 1, so the expected value of the log-likelihood function of the ijth element is

$$Q_{ij}(\Psi | \Psi^{(t)}) = \mathbf{E}_{Z_{ij}|X_{ij}, \Psi^{(t)}}[l_{ij}(\Psi)] = \log(\pi_{ij} \frac{\exp(-\lambda_{ij}) \lambda_{ij}^{X_{ij}}}{X_{ij}!}),$$

and when $X_{ij} = 0$, its expected value is

$$\begin{aligned} Q_{ij}(\Psi | \Psi^{(t)}) =& \mathbf{E}_{Z_{ij}|X_{ij}, \Psi^{(t)}}[l_{ij}(\Psi)] \\ =& \mathbf{P}(Z_{ij} = 0 | X_{ij}, \Psi^{(t)}) \log(1 - \pi_{ij}) + \mathbf{P}(Z_{ij} = 1 | X_{ij}, \Psi^{(t)}) \log(\pi_{ij} \exp(-\lambda_{ij})), \end{aligned}$$

where $\mathbf{P}(Z_{ij} = 0 | X_{ij}, \Psi^{(t)})$ and $\mathbf{P}(Z_{ij} = 1 | X_{ij}, \Psi^{(t)})$ are given in equation (2.9). Therefore,

the total expected log-likelihood function is:

$$\begin{aligned} Q_{ij}(\Psi|\Psi^{(t)}) =& \mathbf{E}_{Z_{ij}|X_{ij},\Psi^{(t)}}[l_{ij}(\Psi)] \\ =& \mathbb{1}_{X_{ij}=0} \cdot \mathbf{P}(Z_{ij}=0|X_{ij},\Psi^{(t)}) \log(1-\pi_{ij}) \\ &+ \mathbb{1}_{X_{ij}=0} \cdot \mathbf{P}(Z_{ij}=1|X_{ij},\Psi^{(t)}) \log(\pi_{ij}\exp(-\lambda_{ij})) \\ &+ \mathbb{1}_{X_{ij}>0} \cdot \log(\pi_{ij}\frac{\exp(-\lambda_{ij})\lambda_{ij}^{X_{ij}}}{X_{ij}!}). \end{aligned}$$

M Step. We update the parameters by maximizing the expected log-likelihood obtained in the E Step by using the same projected gradient descent algorithm as shown in Section 2.2.4.1:

$$\Psi^{(t+1)} = (V^{(t+1)}, W^{(t+1)}, \alpha^{(t+1)}, \beta^{(t+1)}) = \underset{\alpha,\beta\in\mathbb{R}^n, V\in\mathbb{R}^{n\times q_1}, W\in\mathbb{R}^{n\times q_2}}{\arg\max} Q(\Psi|\Psi^{(t)})$$

We apply both estimation methods to the simulation study and real-world data analysis. The two methods have comparable performances in both simulation and real-world data experiments. Compared with the EM algorithm, the projected gradient descent algorithm is easier to tune.

2.3 Theoretical Results

In this section, we present some theoretical results on the estimation of the parameters. Existing work (Ma et al., 2020) has established error bounds for the Bernoulli model, but those techniques highly rely on the excellent properties of the exponential family. While the same techniques work well with the Poisson model and we show error bounds for the Poisson model in Section 2.3.1, the zero-inflated Poisson model does not belong to the exponential family, and the existing approaches do not work. Additionally, the mixture of Φ and Θ in the negative log-likelihood of the zero-inflated Poisson model leads to further difficulties in

developing theoretical results.

To overcome the difficulties, we developed error bounds for the latent space zero-inflated Poisson model by using the Peano remainder of the Taylor Theorem instead of the commonly used Lagrange remainder as in Ma et al. (2020) and Zhang et al. (2022). We present the result for the latent space zero-inflated Poisson model in Section 2.3.2.

2.3.1 Result for the Poisson Model

We consider the feasible parameter space of the latent space Poisson model:

$$\mathcal{F}(n, q, M) = \{\Phi | \Phi = WW^T + \beta 1_n^T + 1_n \beta^T, |\Phi_{ij}| < M, J_n W = W\},$$

where $\Phi \in \mathbb{R}^{n \times n}, W \in \mathbb{R}^{n \times q}, \beta \in \mathbb{R}^n$, and $J_n = I_n - \frac{1}{n} 1_n 1_n^T$.

Let Φ^* be the true parameters of the model that were used in the data generation, and $\hat{\Phi}$ be the estimated parameters obtained by minimizing the loss function:

$$\begin{aligned}(\hat{\beta}, \hat{W}) &= \underset{(W,\beta) \in \mathcal{F}(n,q,M)}{\arg\min} \; l(\beta, W) \\ &= \underset{(W,\beta) \in \mathcal{F}(n,q,M)}{\arg\min} \; -\log P(X|\beta, W).\end{aligned}$$

In this section, we constrain the true parameters and the estimated parameters in the feasible parameter space $\mathcal{F}(n, q, M)$ for the purpose of deriving theoretical results. We do not put these constraints in simulation studies or real data examples. We apply similar constraints in Section 2.3.2 as well.

The following theorem provides a bound on the estimation error of $\hat{\Phi}$.

Theorem 2.3.1 *When Φ^* and $\hat{\Phi}$ belong to the feasible parameter space $\mathcal{F}(n, q, M)$, there exists constant C, such that,*

$$\mathbf{E}\|\hat{\Phi} - \Phi^*\|_F \leq C\sqrt{nq} \tag{2.10}$$

where the constant C does not depend on n, and q.

2.3.2 Results for the Zero-inflated Poisson Model

We consider the following feasible parameter space:

$$\mathcal{F}(n, q_1, q_2, M_\Theta, M_\Phi) = \{\Theta, \Phi | \Theta = VV^T + \alpha 1_n^T + 1_n^T \alpha, \Phi = WW^T + \beta 1_n^T + 1_n \beta^T, \\ |\Theta_{ij}| < M_\Theta, |\Phi_{ij}| < M_\Phi, J_n V = V, J_n W = W\},$$

where $\Theta, \Phi \in \mathbb{R}^{n \times n}, V \in \mathbb{R}^{n \times q_1}, W \in \mathbb{R}^{n \times q_2}, \alpha, \beta \in \mathbb{R}^n$, and $J_n = I_n - \frac{1}{n} 1_n 1_n^T$.

We define a C-neighbourhood of a pair of parameters (Θ^*, Φ^*) as

$$\mathcal{F}_C(\Theta^*, \Phi^*) = \{\Theta, \Phi | (\Theta, \Phi) \in \mathcal{F} \text{ such that } |\Theta_{ij} - \Theta^*_{ij}| < C \text{ and } |\Phi_{ij} - \Phi^*_{ij}| < C\}.$$

Assume we generate a network X with true parameters $(\Theta^*, \Phi^*) \in \mathcal{F}$, and $(\hat{\Theta}, \hat{\Phi})$ be the estimated parameters obtained by minimizing the loss function:

$$(\hat{\alpha}, \hat{V}, \hat{\beta}, \hat{W}) = \underset{(\alpha, V, \beta, W)}{\arg\min}\, l(\alpha, V, \beta, W),$$

subject to $\hat{\Theta} = \hat{V}\hat{V}^T + \hat{\alpha} 1_n^T + 1_n^T \hat{\alpha}, \hat{\Phi} = \hat{W}\hat{W}^T + \hat{\beta} 1_n^T + 1_n \hat{\beta}^T$ and $(\hat{\Theta}, \hat{\Phi}) \in \mathcal{F}_C(\Theta^*, \Phi^*)$ for a pre-specified constant C. The following theorem provides bounds on the estimation error of $\hat{\Theta}$ and $\hat{\Phi}$.

Theorem 2.3.2 *When true parameters $(\Theta^*, \Phi^*) \in \mathcal{F}$ and $(\hat{\Theta}, \hat{\Phi})$ are estimated by the above procedure, there exists constant C such that*

$$\mathbf{E}\|\hat{\Theta} - \Theta^*\|_F, \mathbf{E}\|\hat{\Phi} - \Phi^*\|_F \leq C\sqrt{n \max\{q_1, q_2\}} \tag{2.11}$$

where the constant C does not depend on n, q_1, and q_2.

Remark 1. The error bound derived in Section 2.3 aligns with the simulation studies in Section 2.4, where the estimation error $\|\hat{\Phi} - \Phi^*\|_F$ and $\|\hat{\Theta} - \Theta^*\|_F$ were found to be of order $O(\sqrt{nq})$ ($q = q_1 = q_2$ in the simulation). This is consistent with existing works in Ma et al. (2020) as well.

The proof is given in the Appendix A.

2.4 Simulation Studies

In this section, we present the results of our simulation study and investigate how the estimation of the latent space zero-inflated Poisson model is affected by the number of nodes and the dimensions of the latent spaces. To specify the model parameters, we follow these steps:

- Generate the node degree heterogeneity parameters: $\alpha_i = -\tilde{\alpha}_i / \sum_{j=1}^{n} \tilde{\alpha}_j$, where $\tilde{\alpha}_i \overset{\text{i.i.d.}}{\sim} U[1, 3]$. Generate β_i with the same procedure.
- Generate the latent positions: $v_i = \mu_v + N(0, I_k)$, where $\mu_v \overset{\text{i.i.d.}}{\sim} U[-1, 1]$. Generate w_i with the same procedure.
- Generate the observed co-occurrence matrix: generate X_{ij} with the parameters and model (2.4) - (2.6).

To simplify the experimental setup, we let v_i and w_i have the same dimension, denoted by $q = q_1 = q_2$. For each pair of values $(n, q) \in \{250, 500, 750, 1000\} \times \{2, 4, 6, 8\}$, we generate 20 independent copies of the co-occurrence matrix according to the above procedure.

2.4.1 Effect of the Model Configuration

In this section, we show the effect of model configurations on the performance. Specifically, we vary the number of codes and the latent space dimensions and show their effect on the

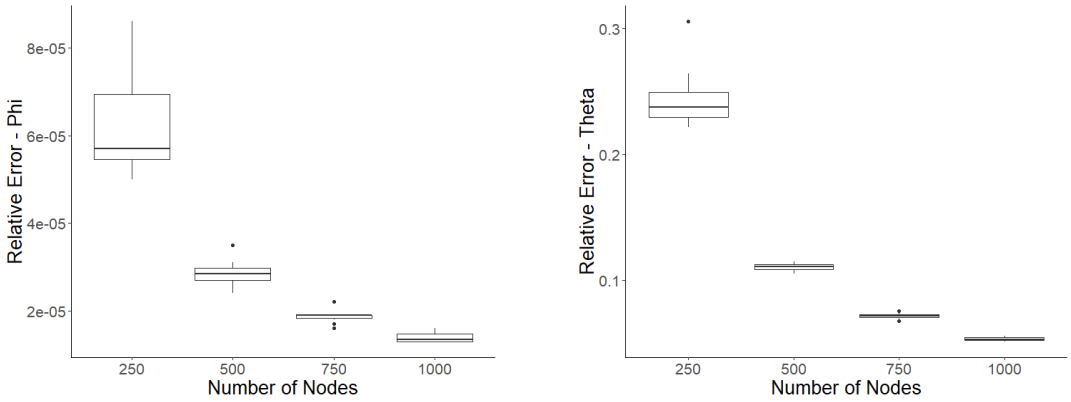


Figure 2.1: Boxplot for relative errors with varying numbers of codes (fix $q = 4$). Left: the relative of Φ; Right: the relative of Θ.

performance.

Effect of the Number of Codes. Figure 2.1 shows the estimation error for varying numbers of codes when we fix $q = 4$, where "Relative Error - Phi" is defined as $\|\hat{\Phi} - \Phi^*\|_F^2/\|\Phi^*\|_F^2$ and "Relative Error - Theta" is defined as $\|\hat{\Theta} - \Theta^*\|_F^2/\|\Theta^*\|_F^2$. The relative estimation errors of both Φ and Θ scale at the order of $1/n$, which agrees well with the theoretical results in Section 2.3. As the number of nodes increases, the relative error becomes smaller. We observe similar results for $q \in \{2, 6, 8\}$.

Effect of the Latent Space Dimension. Figure 2.2 shows the estimation error for varying latent space dimensions when we fix $n = 500$. The relative estimation errors of both Φ and Θ scale at the order of q, which also agree with the theoretical results. Similar results are observed for $n \in \{250, 750, 1000\}$.

2.4.2 Comparison with Related Models.

Then, we compare our proposed models (the Poisson model and the zero-inflated Poisson model) with the model (Bernoulli) in Ma et al. (2020) by some simulations.

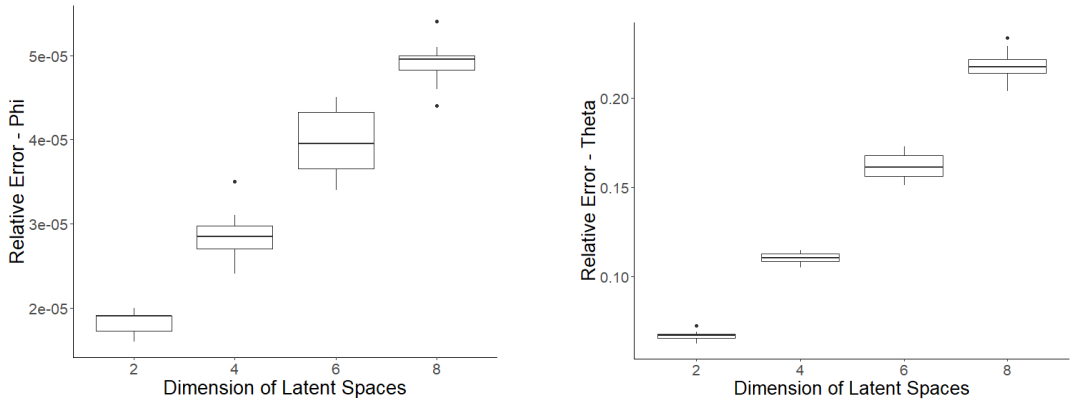


Figure 2.2: Boxplot for relative errors with varying latent space dimensions (fix $n = 500$). Left: the relative of Φ; Right: the relative of Θ.

2.4.2.1 Zero-Inflated Poisson Model vs. Bernoulli Model

We use the same settings as the previous simulations, but we generate β_i by $\beta_i \sim U(\beta_0 - 0.5, \beta_0 + 0.5)\}$. When β_0 increases, the probability of $X_{ij} = 0$ will become closer to $1 - \pi_{ij}$, which is essentially a Bernoulli distribution. Therefore, the zero-inflated Poisson model will become closer to the Bernoulli model in Ma et al. (2020).

We vary β_0 in $\{-2, -1.5, ..., 1.5\}$ and generate matrix X with the zero-inflated Poisson model. Then, we fit the zero-inflated Poisson model and the Bernoulli model on the generated matrix and compare their abilities to detect positive-weighted edges. Let p_{ij} be the true probability of $X_{ij} > 0$, and $\hat{p}_{ij}$ be the probability estimated by the models. We compare our proposed zero-inflated Poisson model and the Bernoulli model by comparing the following two measures:

- Rank Correlation: the rank correlation between the true probabilities $\{p_{ij}\}$ and the estimated probabilities $\{\hat{p}_{ij}\}$;
- Relative Error: the relative error of the estimated probabilities

$$\text{Relative Error} = \text{mean}(\{(\hat{p}_{ij} - p_{ij})^2/(p_{ij}(1 - p_{ij}))\}).$$

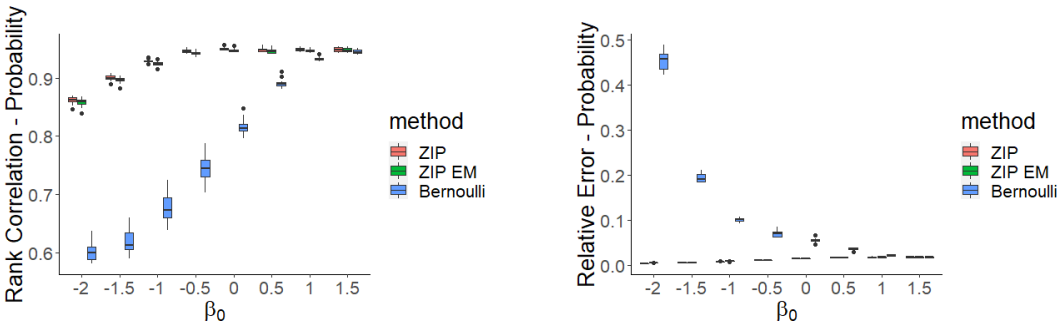


Figure 2.3: The performance of the two models with varying β_0. Left: the rank correlation between true probabilities and the estimated probabilities; Right: the relative error of the estimated probabilities. We show the performances of three methods: the latent space zero-inflated model fitted with projected gradient descent (labeled as "ZIP"), the latent space zero-inflated model fitted with EM algorithm (labeled as "ZIP EM"), and the Bernoulli model.

Figure 2.3 shows the performance of the zero-inflated Poisson model (labeled as "ZIP" and "ZIP EM") and the Bernoulli model. We can see the zero-inflated Poisson model always outperforms the Bernoulli model within the range of β_0. The difference in performance becomes smaller when β_0 increases, and the difference becomes negligible when the true model is almost the same as the Bernoulli model ($\beta_0 = 1.5$).

2.4.2.2 Zero-Inflated Poisson Model vs. Poisson Model

Similarly, we can compare the zero-inflated Poisson model with the ordinary Poisson model by changing α's. We generate α_i by $\alpha_i \sim U(\alpha_0 - 0.5, \alpha_0 + 0.5)\}$. When α_0 increases, π_{ij} will also increase, which means the corresponding zero-inflated Poisson model will become closer to the ordinary Poisson model.

We vary α_0 in $\{-2, -1.5, ..., 2\}$ and generate matrix X with the zero-inflated Poisson model. Then, we fit the zero-inflated Poisson model and the Poisson model on the generated matrix and compare their abilities to predict the weights (X_{ij}). Let X_{ij} be the co-occurrence time of the i-th code and the j-th code, and $\hat{X}_{ij}$ be the weight estimated by the models. The following two measures are used to compare our proposed zero-inflated Poisson model and the Poisson model:

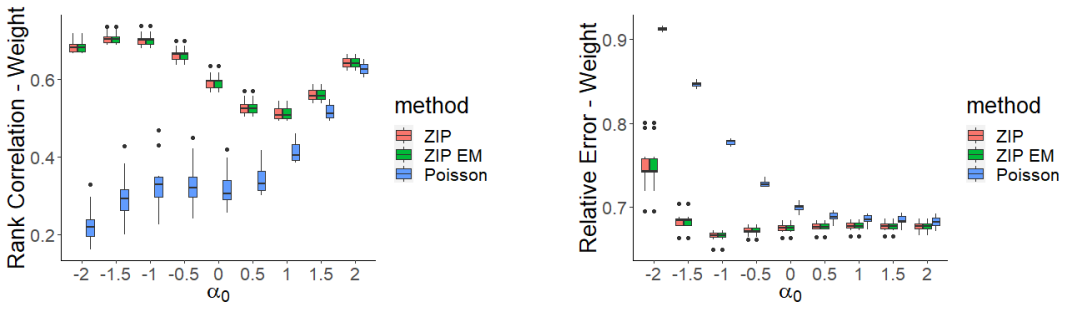


Figure 2.4: The performance of the two models with varying α_0. Left: the rank correlation between true weights and the estimated weights; Right: the relative error of the estimated weights. The labels of methods are the same as in Figure 2.4.

- Rank Correlation: the rank correlation between the true weights $\{X_{ij}\}$ and the estimated weights $\{\hat{X}_{ij}\}$;
- Relative Error: the relative error of the estimated weights

$$\text{Relative Error} = \text{mean}(\{((\hat{EX}_{ij} - EX_{ij})/\hat{EX}_{ij})^2\}).$$

Figure 2.4 shows the performance of the Poisson model and the zero-inflated Poisson model(labeled as "ZIP" and "ZIP EM"). We can see the zero-inflated Poisson model always outperforms the Poisson model. The difference in performance becomes smaller when α_0 increases, i.e., when the zero-inflated Poisson model becomes closer to the ordinary Poisson model.

The result in Section 2.4.2 indicates that both the Bernoulli model and the Poisson model are special cases of our proposed zero-inflated Poisson model, which demonstrates the flexibility of our proposed model.

2.5 Real-world Data Examples

In this section, we evaluate the effectiveness of our proposed methods in modeling and analyzing real-world electronic health record (EHR) data by applying them to the MIMIC-

III database (Johnson et al., 2019).

2.5.1 The MIMIC-III Database

The MIMIC-III database collects health-related data of over 40,000 patients who had been admitted to intensive care units (ICU) of the Beth Isreal Deaconess Medical Center between 2001 and 2012 (Johnson et al., 2016, 2019; Goldberger et al., 2000). It includes a broad range of information, such as laboratory test results, medical orders, billing information, demographics, as well as medical notes and reports for 53,423 hospital admissions. The database is publicly available and has been commonly used in healthcare research (Che et al., 2018; Shickel et al., 2017; Suresh et al., 2017).

In the database, each admission is annotated with a set of International Classification of Disease version 9 (ICD-9) codes. We define the co-occurrence times of a pair of ICD-9 codes as the number of times that they appear in the same record. We construct a co-occurrence matrix of pairwise co-occurrences for all ICD-9 codes. We apply our proposed models to this co-occurrence matrix to learn vector representations of each code. The learned representations are then used to predict whether a patient will be readmitted to an ICU within 30 days.

2.5.2 Data Preprocessing

Because of the substantial differences between adult and pediatric physiology, we only use the admission records of patients aged 18 years or older. For each admission, we collect the corresponding ICD-9 codes and a set of nine variables related to the patients (e.g., age, gender, ethnicity, number of ICU stays).

We define a binary outcome variable to indicate whether a patient is readmitted to an ICU within 30 days of their current discharge. The 30-day readmission rate for adult patients in the MIMIC-III database is 5.32%.

We divide the dataset into a training set, a validation set, and a testing set. The training

set contains about 80% of the admissions, while the rest two sets each contain about 10% of the admissions. For an ICD-9 code that rarely occurred, we put at least one admission record that contains it into the training set.

2.5.3 Experiment Design

We use our proposed models to learn vector representations for each ICD-9 code by applying them to the co-occurrence matrix of ICD-9 codes. We then aggregate the vector representations by averaging them across all corresponding ICD-9 codes in each admission. These vector representations, along with other variables such as age, gender, ethnicity, and number of ICU stays, are used as predictors for predicting the outcome. We use Random Forest as the prediction model due to its superior performance in our experiments compared to other commonly used models such as Logistic Regression and XGBoost.

In literature, a popular way to generate vector representations of ICD-9 codes is using word embedding models in natural language processing (NLP). For instance, several studies such as Shi et al. (2021); Nguyen et al. (2018); Choi et al. (2017b), and Feng et al. (2017) used the Skip-gram model, Choi et al. (2017a) applied the GloVe model, and Cai et al. (2018) utilized the CBOW model. Thus, to evaluate the performance of our proposed methods, we compare them with the above-mentioned models. Additionally, we compare our method with a prediction model that does not use any ICD code information (labeled as "None") to highlight the benefits of using ICD code representations.

The optimal latent space dimensions for all models are selected from $\{10, 25, 50, 100\}$ by cross-validation (Browne, 2000).

2.5.4 Results

We show the average AUC score and standard error of each prediction model in Table 2.1. We can see that both our proposed latent space zero-inflated Poisson model (labeled as "ZIP") and the latent space Poisson model (labeled as "Poisson") outperform all other

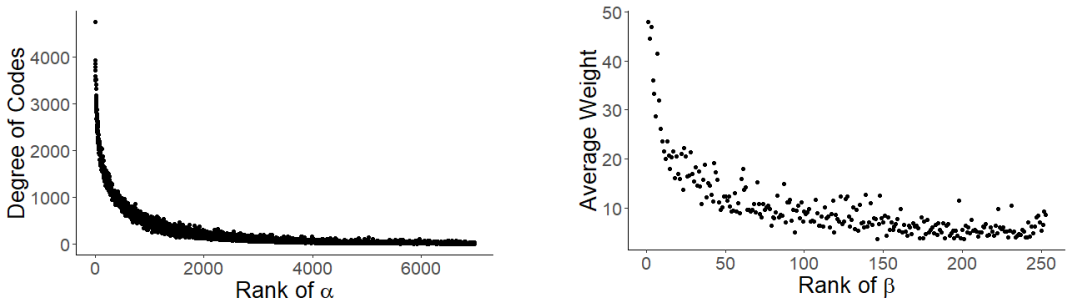


Figure 2.5: The change of learned α and β with the characteristics of the ICD codes. Left: the degree of codes vs. the rank of estimated α; Right: the average weight vs. the rank of estimated β. only codes with positive estimated β are shown.

methods. Also, using ICD code representations in the prediction could significantly improve the prediction performance (compared with "None").

Embedding Method	None	GloVe	CBOW	Skip-Gram	Poisson	ZIP
Test AUC	0.651	0.712	0.713	0.720	0.723	0.726
(standard error)	(0.004)	(0.003)	(0.003)	(0.004)	(0.003)	(0.004)

Table 2.1: The test AUC scores and corresponding standard error of different methods. ZIP: the proposed latent space zero-inflated Poisson model; Poisson: the latent space Poisson model; None: does not use any ICD code information.

Figure 2.5 shows how the learned α and β in the zero-inflate Poisson model are changing with the characteristics of the ICD codes. In the left figure, each point represents an ICD code. The vertical axis shows the degree of the code, which represents the number of other ICD codes that have co-occurred with it. The horizontal axis shows the rank of its learned parameter α, ordered in descending order.

In the right figure, the vertical axis represents the average co-occurrence number of the code, which is the average number of positive co-occurrences with other ICD codes. The horizontal axis shows the rank of its learned parameter β, ordered in descending order. Only ICD codes with positive parameter β are displayed in the figure.

The decreasing pattern observed in both figures indicates a strong correlation between the learned α and β parameters and the characteristics of ICD codes. Specifically, ICD

codes that co-occur with more other codes have a larger α, while ICD codes that co-occur more frequently with other codes have a larger β. This finding is consistent with the model definition of α and β.

2.5.5 Case Study

In this section, we provide examples to demonstrate that the ICD code representations we learned accurately capture the clinical meanings of the codes. We present the top five most frequent ICD-9 codes in the MIMIC-III database in Table 2.2.

ICD-9 code	Count	Code Description
401.9	226,978	Unspecified essential hypertension
428.0	183,892	Congestive heart failure unspecified
427.31	171,022	Atrial fibrillation
414.01	141,513	Coronary atherosclerosis of native coronary artery
584.9	136,181	Acute kidney failure, unspecified

Table 2.2: The top five frequent ICD-9 codes in MIMIC-III

To further illustrate our point, we found the ICD-9 codes whose embeddings are closest to the top two most frequent ICD-9 codes (401.9 and 428.0). The zero-inflated Poisson model with latent space dimensions 25 is used to learn ICD code embedding.

The two ICD-9 codes that are closest to 401.9 (Unspecified essential hypertension) are 272.0 (Pure hypercholesterolemia) and 250.00 (Diabetes mellitus without mention of complication, type ii or unspecified type, not stated as uncontrolled) in our learned space. The three diseases, hypertension, hypercholesterolemia, and diabetes, are common concurrent diseases (Song et al., 2016).

Similarly, the ICD-9 code 428.0 (Congestive heart failure unspecified) is closest to 410.71 (Subendocardial infarction, initial episode of care) and 425.4 (Dilated cardiomyopathy), which are also related to heart diseases.

As cancer is a leading cause of death, we also looked at an example of a cancer-related code. The ICD-9 code 198.3 (Secondary malignant neoplasm of the brain and spinal cord)

frequently occurs in the MIMIC-III database. In our learned space, the three ICD-9 codes closest to it are 197.0 (Secondary malignant neoplasm of the lung), 198.5 (Secondary malignant neoplasm of bone and bone marrow), and 198.4 (Secondary malignant neoplasm of other parts of the nervous system). These codes are all related to secondary malignant neoplasms.

It is worth noting that none of the closest code pairs in the above examples co-occur the most frequently. This indicates that our model captures more complex and detailed relationships between the codes beyond just their co-occurrence information.

2.6 Discussion

In this chapter, we propose a novel approach for ICD code embedding - the latent space zero-inflated Poisson model. Unlike traditional latent space models for binary networks, our proposed models not only model the existence of co-occurrence but also the number of co-occurrence times. Both simulation studies and real-world data examples demonstrate that our proposed method can achieve better performance by also modeling the co-occurrence times.

The zero-inflated Poisson distribution in the proposed model plays an important role in handling abundant zeros in the co-occurrence matrix. The Poisson part of the zero-inflated Poisson distribution characterizes the positive co-occurrence times. On the other hand, we would not lose the information contained in the inflated zeros, as the Bernoulli part of the distribution captures it.

As discussed in Section 2.4, both the Bernoulli model (Ma et al., 2020) and the latent space Poisson model are special cases of our proposed latent space zero-inflated Poisson model. This shows the high flexibility of our proposed model and its ability to fit different data.

Our proposed method automatically learns the vector representations, which can serve as

features in downstream tasks. The usefulness of these vector representations is demonstrated in Section 2.5. Because there are a large amount of ICD codes (over 13,000 different ICD-9 codes and more than 68,000 ICD-10 codes) and the sample size of healthcare-related research is usually relatively small, it is infeasible to use ICD codes directly in many statistical models. The vector representations provided by our model can substantially reduce the data dimension and make it possible to utilize modern statistical models.

CHAPTER 3

Joint Latent Space Zero-Inflated Poisson Model for ICD Code Translation

3.1 Introduction

The amount of Electronic Health Record (EHR) data has rapidly increased across healthcare systems, offering invaluable insights into patient care and healthcare research. A typical EHR dataset includes various variables such as patient demographics, lab results, diagnoses, medications, orders, and notes. For each health record, a group of International Classification of Diseases (ICD) codes are assigned by doctors to provide important clinic information. These codes serve as precise descriptors of patients' medical conditions, procedures received, and other clinical details. The ICD codes in EHRs are well-structured with almost no missing data points and are utilized worldwide, making them perfect candidates for healthcare research.

Much work has been undertaken to apply ICD codes in clinical applications. Due to the large number of distinct ICD codes, researchers have to apply embedding methods to map ICD codes into lower-dimensional vectors. There is a group of well-studied word embedding models in the field of natural language processing (NLP) that can be used to map ICD codes, such as Skip-gram (Mikolov et al., 2013) and GloVe (Pennington et al., 2014). Therefore, researchers typically borrow word embedding methods to model ICD codes, such as the work in Shi et al. (2021); Nguyen et al. (2018); Choi et al. (2017b); Feng et al. (2017);

Choi et al. (2017a); Cai et al. (2018). However, as mentioned in Chapter 2, ICD codes and natural language words have significantly different characteristics. Instead of utilizing word embedding methods, Chapter 2 proposed a latent space zero-inflated Poisson model for ICD codes specifically. The above works on ICD codes all show the value of ICD codes in clinical research.

The Ninth revision of the International Classification of Diseases (ICD-9), designed in the late 1970s, was used in the United States for several decades until 2015. In October 2015, hospitals in the United States transitioned from ICD-9 to ICD-10 codes because the former no longer met the medical needs of healthcare providers and patients. This transition resulted in many debates and discussions (Wollman, 2011; Topaz et al., 2013; Khera et al., 2018; Hamedani et al., 2021; Kusnoor et al., 2020). The transition introduced challenges in converting and merging EHR data and applications to the newer system. The General Equivalence Mappings (GEM) (Butler, 2007) are mappings between ICD-9 codes and ICD-10 codes created by medical experts. However, the relationship among ICD codes is too complicated to be captured in just a simple mapping. The subtle differences between similar ICD codes and the fundamental differences in the ICD-9 and ICD-10 structures lead to difficulties in finding a precise mapping between the two versions of ICD codes. As a result, within the GEM, only some of the codes have a one-to-one mapping to the other code system. Many codes are mapped to multiple codes in the GEM. For instance, 255 ICD-9 codes are mapped to more than 50 ICD-10 codes, and more than 7,200 ICD-10 codes are mapped to multiple ICD-9 codes (Wollman, 2011). Research also showed that the transition has affected the usage of ICD codes. For example, Hamedani et al. (2021) shows that the monthly prevalence of the ICD code for subarachnoid hemorrhage was stable before the transition. However, it increased after the transition, even though the actual disease prevalence remained stable. Consequently, there were recommendations for studies to limit themselves to just one ICD system, potentially resulting in the loss of a large number of samples. Therefore, the ability to map historical data with ICD-9 codes to the newer system is crucial, especially

given that sample sizes are often relatively small in clinical studies. Being able to map ICD-9 codes and ICD-10 codes is also important for personalized medicine because a patient could have medical records with both ICD code versions.

To address the above challenge, we introduce a joint latent space zero-inflated Poisson model that learns embeddings for both ICD-9 and ICD-10 codes simultaneously, as well as a transformation connecting the two ICD code versions. Specifically, Our proposed model employs two latent space zero-inflated Poisson models, as described in Chapter 2, one for ICD-9 codes and one for ICD-10 codes. A linear transformation T is learned to align the latent spaces of the two latent space zero-inflated Poisson models with the help of a high-quality code dictionary containing a small amount of matched code pairs. Each code pair has an ICD-9 code and an ICD-10 code with identical medical meaning. The code dictionary is obtained from the ICD General Equivalence Mappings (GEM) (Butler, 2007). While the latent space zero-inflated Poisson models optimize the vector representations for both ICD code systems, the transformation T tends to align the vector representations of matched code pairs.

The efficacy of our model for ICD code mapping relies on the consistency of the diseases' internal relationships and interactions, which would not be influenced by the ICD code system. For instance, hypertension and hypotension should never co-occur on the same medical record. Based on the latent space zero-inflated Poisson model, the latent vectors v for hypertension and hypotension should be approximately in opposite directions to each other in both ICD systems. Therefore, a transformation trained by aligning latent vectors for hypertension will also align the latent vectors for hypotension. This internal structure of disease makes the transformation learned from the dictionary provide a generalized mapping that can be extrapolated to map other ICD codes. The main contributions of this chapter are the following.

- We introduce a novel approach, the joint latent space zero-inflated Poisson model, which learns embeddings for ICD-9 and ICD-10 codes simultaneously, as well as a

transformation that maps the ICD codes to the other system. The mapping can be used in aggregating EHR datasets and healthcare applications with ICD-9 codes with the newer ICD-10 system, which addresses one of the main challenges raised by the ICD code system transition.

- We provided error bounds for the estimation of vector representation. The joint latent space zero-inflated Poisson model maintains the same error bounds as the model proposed in Chapter 2.

- In a real data experiment, we processed the Nationwide Readmissions Database (NRD) to generate a dataset with both ICD-9 and ICD-10 codes. We designed an ICD code translation task with that dataset and compared our proposed method with existing approaches. Our proposed model outperforms all existing approaches. The good translation performance shows the practical value of our proposed model.

The rest of the chapter is organized as follows. In Section 3.2, we first review the latent space zero-inflated Poisson model, and then propose the joint latent space zero-inflated Poisson model and introduce the fitting method. In Section 3.3, we provide error bounds for the estimation errors. In Section 3.4, using synthetic data, we show the efficacy of our model. We then investigate the effects of model configurations and the number of matched code pairs on the model performance. We deploy the proposed method on real-world datasets and demonstrate its practical applications in Section 3.5. In Section 3.6, we conclude this chapter and provide some discussions.

3.2 Model

In this section, we first introduce necessary notations and review the latent space zero-inflated Poisson model proposed in Chapter 2, which is a crucial component of the proposed

framework. Then we present the joint latent space zero-inflated Poisson model and its identifiability conditions. Additionally, we introduce the fitting algorithm.

3.2.1 Notation

Assume there are n ICD-9 codes and m ICD-10 codes, and we observed the co-occurrence matrices $X^{(9)} \in \mathbb{N}^{n\times n}$ and $X^{(10)} \in \mathbb{N}^{m\times m}$ of the two versions of ICD codes respectively, where $X^{(9)}_{ij} = X^{(9)}_{ji} \in \mathbb{N}$ is the number of co-occurrence times of the i-th ICD-9 code and the j-th ICD-9 code. Similarly, $X^{(10)}_{ij} = X^{(10)}_{ji} \in \mathbb{N}$ is the number of co-occurrence times of the i-th ICD-10 code and the j-th ICD-10 code. Let $\{c_1, c_2, ..., c_n\}$ be the set of all ICD-9 codes and $\{d_1, d_2, ..., d_m\}$ be the set of all ICD-10 codes.

Assume we have access to a mapping or a dictionary of L pairs of codes. Without loss of generality, let the L pairs of codes be $(c_1, d_1), (c_2, d_2), ..., (c_L, d_L)$. The ICD-9 code and the ICD-10 code in each pair have the same medical meaning.

3.2.2 The Latent Space Zero-Inflated Poisson Model

In this section, we review the latent space zero-inflated Poisson model proposed in Chapter 2, which is a key component of our model. The latent space zero-inflated Poisson model learns latent positions of a single version of ICD codes (e.g., the ICD-9 codes) based on the co-occurrence matrix. The model assumes the co-occurrence time of a pair of ICD codes follows a zero-inflated Poisson distribution whose parameters are connected to the latent positions of the corresponding codes by the inner-product model. Specifically, code i is represented by two latent vectors $v_i, w_i \in \mathbb{R}^q$ in a low-dimension latent space of q degree and two heterogeneity parameters α_i and β_i. Let $V = [v_1, v_2, ..., v_n]^T \in \mathbb{R}^{n\times q}$ and $W = [w_1, w_2, ..., w_n]^T \in \mathbb{R}^{n\times q}$ be the matrices consisting of the latent positions of all codes, where n is the total number of codes.

The latent space zero-inflated Poisson model considers the following inner-product latent space model (Hoff, 2003; Ma et al., 2020), for any $i < j$,

$$X_{ij} = X_{ji} \sim \begin{cases} 0 & \text{with probability } 1 - \pi_{ij}, \\ \text{Poisson}(\lambda_{ij}) & \text{with probability } \pi_{ij}, \end{cases} \tag{3.1}$$

where,

$$\text{logit}(\pi_{ij}) = v_i \cdot v_j + \alpha_i + \alpha_j = \Theta_{ij}, \tag{3.2}$$

$$\log(\lambda_{ij}) = w_i \cdot w_j + \beta_i + \beta_j = \Phi_{ij}. \tag{3.3}$$

The parameters α_i's and β_i's are node degree heterogeneity parameters. The parameters (V, α) in (3.2) model the existence of Poisson-distributed co-occurrence time, while the parameters (W, β) in (3.3) model the number of co-occurrences. The latent positions V and W can be considered vector representations of the nodes.

3.2.3 The Joint Latent Space Zero-Inflated Poisson Model

To overcome the problem of lacking an ICD code mapping model, we propose a novel approach: the joint latent space zero-inflated Poisson model. The proposed model consists of two latent space zero-inflated Poisson models and a linear transformation T. The two latent space zero-inflated Poisson models, as proposed in Chapter 2, are used to learn vector representations of the two versions of ICD codes respectively, while the transformation T is a linear transformation learned based on a high-quality dictionary with L matched pairs of ICD codes $\{(c_1, d_1), (c_2, d_2), ..., (c_L, d_L)\}$. The two codes (one ICD-9 code and one ICD-10 code) in a matched pair (c_i, d_i) have the identical medical meaning. The transformation T is trained to align the vector representations of each pair of the matched codes. By aligning the matched codes, the transformation gains the ability to also align other ICD codes with identical or similar medical meanings.

Specifically, ICD-9 code c_i is represented by two latent vectors $v_i^{(9)}, w_i^{(9)} \in \mathbb{R}^{q_9}$ in a low-dimension latent space of q_9 degree and two heterogeneity parameters $\alpha_i^{(9)}$ and $\beta_i^{(9)}$, while

ICD-10 code d_i is associated with $v_i^{(10)}, w_i^{(10)} \in \mathbb{R}^{q_{10}}$ in a latent space of q_{10} degree and two heterogeneity parameters $\alpha_i^{(10)}$ and $\beta_i^{(10)}$. Our model assumes the co-occurrence matrices $X^{(9)}$ and $X^{(10)}$ follow the zero-inflated Poisson distribution determined by the above parameters:

$$X_{ij}^{(9)} = X_{ji}^{(9)} \sim \begin{cases} 0 & \text{with probability } 1 - \pi_{ij}^{(9)}, \\ \text{Poisson}(\lambda_{ij}^{(9)}) & \text{with probability } \pi_{ij}^{(9)}, \end{cases} \tag{3.4}$$

$$X_{ij}^{(10)} = X_{ji}^{(10)} \sim \begin{cases} 0 & \text{with probability } 1 - \pi_{ij}^{(10)}, \\ \text{Poisson}(\lambda_{ij}^{(10)}) & \text{with probability } \pi_{ij}^{(10)}, \end{cases} \tag{3.5}$$

where,

$$\text{logit}(\pi_{ij}^{(9)}) = v_i^{(9)} \cdot v_j^{(9)} + \alpha_i^{(9)} + \alpha_j^{(9)} = \Theta_{ij}^{(9)}, \tag{3.6}$$

$$\log(\lambda_{ij}^{(9)}) = w_i^{(9)} \cdot w_j^{(9)} + \beta_i^{(9)} + \beta_j^{(9)} = \Phi_{ij}^{(9)}, \tag{3.7}$$

$$\text{logit}(\pi_{ij}^{(10)}) = v_i^{(10)} \cdot v_j^{(10)} + \alpha_i^{(10)} + \alpha_j^{(10)} = \Theta_{ij}^{(10)}, \tag{3.8}$$

$$\log(\lambda_{ij}^{(10)}) = w_i^{(10)} \cdot w_j^{(10)} + \beta_i^{(10)} + \beta_j^{(10)} = \Phi_{ij}^{(10)}. \tag{3.9}$$

The transformation T consists of two linear transformations, each transformation maps a vector of q_{10} degrees to a vector of q_9 degrees. We use $T = (T_V, T_W)$ to represent it, where $T_V, T_W \in \mathbb{R}^{q_9 \times q_{10}}$ are the matrix forms of the transformation. The transformation maps vector representations of an ICD-10 code into latent spaces of ICD-9 codes. Assume we have a high-quality dictionary with L matched pair of ICD codes $\{(c_1, d_1), (c_2, d_2), ..., (c_L, d_L)\}$. Let the latent vectors of c_i be $(v_i^{(9)}, w_i^{(9)})$ and the latent vectors of d_i be $(v_i^{(10)}, w_i^{(10)})$. The transformation $T = (T_V, T_W)$ maps $(v_i^{(10)}, w_i^{(10)})$ to $(v_i', w_i') \triangleq (T_V(v_i^{(10)}), T_W(w_i^{(10)}))$. Because the medical meanings of c_i and d_i are identical, ideally, we want v_i' being close to $v_i^{(9)}$ and w_i' being close to $w_i^{(9)}$. To achieve that, the transformation T is learned to minimize the distance between (v_i', w_i'), and $(v_i^{(9)}, w_i^{(9)})$. We will discuss more about this in the fitting method part.

The latent position $(V^{(9)}, W^{(9)})$ and $(V^{(10)}, W^{(10)})$ are considered the vector representa-

tion of the ICD codes. It can be used to explain the relationships and interactions between different diseases and used in downstream tasks. Also, by using the transformation, we can map the latent positions of different versions of ICD codes into the same space, which has various potential uses. For example, it can help us translate between different versions of ICD codes. Also, in real-world medical applications, it could help us aggregate datasets with different versions of ICD codes.

Identifiability To ensure the identifiability of the parameters, we apply the same assumptions as in Ma et al. (2020). We assume the latent vectors are centered, i.e., $J_n V^{(9)} = V^{(9)}$, $J_n W^{(9)} = W^{(9)}$, $J_m V^{(10)} = V^{(10)}$, $J_m W^{(10)} = W^{(10)}$, where $J_n = I_n - \frac{1}{n} 1_n 1_n^T$ and $J_m = I_m - \frac{1}{m} 1_m 1_m^T$. This identifiability assumption makes sure that the parameters are identifiable up to orthogonal transformations. Also, VV^T and WW^T are directly identifiable under this assumption ($V \in \{V^{(9)}, V^{(10)}\}$ and $W \in \{W^{(9)}, W^{(10)}\}$).

3.2.4 Fitting Method

To fit the model, we apply the Adam optimizer (Kingma and Ba, 2014) to minimize the loss function (i.e., the negative log-likelihood function).

The loss function of the joint model consists of three parts: the loss functions for the two latent space zero-inflated Poisson models ($l^{(9)}$ and $l^{(10)}$), and a loss function l_t for the transformation (defined in (3.11)). We have the total loss function l be

$$l = l^{(9)} + \gamma \cdot l_t + l^{(10)}, \tag{3.10}$$

where γ is an adjustable weight for the transformation part of loss.

The loss functions for the two latent space zero-inflated Poisson models are the negative

log-likelihood as defined in Chapter 2:

$$
\begin{aligned}
l^{(k)}(\alpha^{(k)}, V^{(k)}, \beta^{(k)}, W^{(k)}) &= -\sum_{i<j=1}^{n} \log P(X_{ij}^{(k)} | \alpha^{(k)}, V^{(k)}, \beta^{(k)}, W^{(k)}) \\
&= -\sum_{i<j=1}^{n} (\mathbb{1}_{X_{ij}^{(k)}=0} \log(1 + e^{\Theta_{ij}^{(k)} - \exp(\Phi_{ij}^{(k)})}) \\
&\qquad + \mathbb{1}_{X_{ij}^{(k)}>0}(\Theta_{ij}^{(k)} + X_{ij}^{(k)}\Phi_{ij}^{(k)} - e^{\Phi_{ij}^{(k)}}) - \log(1 + e^{\Theta_{ij}^{(k)}}))
\end{aligned}
$$

where $k = 9, 10$ is the ICD code version number.

We use the mean square distance between the vector representations of the matched codes as the loss function for the transformation:

$$
l_t = \frac{1}{L} \sum_{i=1}^{L} \left(\text{MSE}(v_i^{(9)}, T_V(v_i^{(10)})) + \text{MSE}(w_i^{(9)}, T_W(w_i^{(10)}) \right). \tag{3.11}
$$

By minimizing the loss function l_t, the transformation T gains the ability to align the vector representations of the matched codes in $\{(c_1, d_1), (c_2, d_2), ..., (c_L, d_L)\}$

Then, all parameters $\mathcal{P} = \{\alpha^{(k)}, V^{(k)}, \beta^{(k)}, W^{(k)}, T_v, T_w | k = 9, 10\}$ are fitted using the Adam optimizer by minimizing the loss function (3.10).

3.3 Theoretical Results

In this section, we show some theoretical results on the model estimation. Existing work Ma et al. (2020) presented estimation error bounds for the latent space Bernoulli model. The work in Chapter 2 provided similar theoretical guarantees for the latent space Poisson model and the latent space zero-inflated Poisson model.

We extend the result in Chapter 2 to the joint latent space zero-inflated Poisson Model. We show theoretical guarantees for the estimation of parameters.

3.3.1 Result for the Parameters Estimation

Let the feasible parameter space of the joint latent space zero-inflated Poisson model be

$$\begin{aligned}\mathcal{F}(n_k, q_k, M_\Theta, M_\Phi, k \in \{9,10\}, T_V, T_W) = \\ \{\Theta^{(k)}, \Phi^{(k)}, k \in \{9,10\} | \Theta^{(k)} = V^{(k)} V^{(k)T} + \alpha^{(k)} 1_{n_k}^T + 1_{n_k} \alpha^{(k)T}, \\ \Phi^{(k)} = W^{(k)} W^{(k)T} + \beta^{(k)} 1_{n_k}^T + 1_{n_k} \beta^{(k)T}, \\ J_{n_k} V^{(k)} = V^{(k)}, J_{n_k} W^{(k)} = W^{(k)}, \\ |\Theta_{ij}^{(k)}| < M_\Theta, |\Phi_{ij}^{(k)}| < M_\Phi, k \in \{9,10\}, \\ T_V, T_W \in \mathbb{R}^{q_9 \times q_{10}}\},\end{aligned}$$

where $\Theta^{(k)}, \Phi^{(k)} \in \mathbb{R}^{n_k \times n_k}, V^{(k)} \in \mathbb{R}^{n_k \times q_k}, W^{(k)} \in \mathbb{R}^{n_k \times q_k}, \alpha^{(k)}, \beta^{(k)} \in \mathbb{R}^{n_k}$, and $J_{n_k} = I_{n_k} - \frac{1}{n_k} 1_{n_k} 1_{n_k}^T$. M_Θ and M_Φ are pre-defined upper bounds.

We define a C-neighbourhood of a pair of parameters (Θ^*, Φ^*) as

$$\mathcal{F}_C(\Theta^*, \Phi^*) = \{\Theta, \Phi | (\Theta, \Phi) \in \mathcal{F} \text{ such that } |\Theta_{ij} - \Theta_{ij}^*| < C \text{ and } |\Phi_{ij} - \Phi_{ij}^*| < C\}.$$

Assume we generate a network $(X^{(9)}, X^{(10)})$ with true parameters $(\Theta^{(k)^*}, \Phi^{(k)^*}) \in \mathcal{F}$, and the latent positions of the first L pairs of codes are matched by a pair of transformation $T = (T_V, T_W)$. Specifically, $v_i^{(9)} = T_V(v_i^{(10)})$ and $w_i^{(9)} = T_W(w_i^{(10)})$ for any $i \in \{1, 2, ..., L\}$. Let $(\hat{\Theta}^{(k)}, \hat{\Phi}^{(k)})$ be the estimated parameters obtained by minimizing the loss function l using the method presented in Section 3.2, subject to $\hat{\Theta}^{(k)} = \hat{V}^{(k)} \hat{V}^{(k)T} + \hat{\alpha}^{(k)} 1_n^T + 1_n \hat{\alpha}^{(k)T}, \hat{\Phi}^{(k)} = \hat{W}^{(k)} \hat{W}^{(k)T} + \hat{\beta}^{(k)} 1_n^T + 1_n \hat{\beta}^{(k)T}$ and $(\hat{\Theta}^{(k)}, \hat{\Phi}^{(k)}) \in \mathcal{F}_C(\Theta^{(k)^*}, \Phi^{(k)^*})$ for a pre-specified constant C.

The following theorem provides bounds on the estimation error of $\hat{\Theta}^{(k)}$ and $\hat{\Phi}^{(k)}$.

Theorem 3.3.1 *Let $(\Theta^{(k)}, \Phi^{(k)})$ be the true parameters and $(\hat{\Theta}^{(k)}, \hat{\Phi}^{(k)})$ are estimated by the*

proposed procedure, there exists constant C_1 *such that*

$$\mathbf{E}\|\hat{\Theta}^{(k)} - \Theta^{(k)}\|_F, \mathbf{E}\|\hat{\Phi}^{(k)} - \Phi^{(k)}\|_F \leq C_1\sqrt{\max\{n_9 q_9, n_{10} q_{10}\}} \tag{3.12}$$

where $k \in \{9, 10\}$ *and the constant* C_1 *does not depend on* $n_9, n_{10}, q_9,$ *and* q_{10}.

We can see that the joint latent space zero-inflated Poisson model maintains the same error bound as the latent space zero-inflated Poisson model, even though it has a much more complicated structure. The proof is given in the Appendix B.

3.4 Simulation Studies

In this section, we present the simulation results and investigate how the number of model dimensions and the amount of paired codes would influence the model performance. The model parameters are generated following these steps:

- Generate 2,000 source codes and 2,000 target codes, each code has latent space positions in one of the corresponding latent space zero-inflated Poisson models. Specifically, there are two latent space zero-inflated Poisson models, one for the source codes, and one for the target codes. Let the latent positions for source codes be $v_i^{(s)}$ and $w_i^{(s)}$, while the latent positions target codes be $v_i^{(t)}$ and $w_i^{(t)}$.

- Without loss of generality, let the first L source codes and first L target codes be the L pairs of matched codes (i.e., $v_i^{(t)} = T_v(v_i^{(s)})$ and $w_i^{(t)} = T_w(w_i^{(s)})$ for any $1 \leq i \leq L$). We very L from 50 to 300. The L pairs of matched codes are used for learning the transformation.

- To test the model, the latent positions of the last 1,000 target codes and source codes are also matched with the same transformation (i.e., $v_i^{(t)} = T_v(v_i^{(s)})$ and $w_i^{(t)} = T_w(w_i^{(s)})$ for any $1001 \leq i \leq 2000$). These 1,000 pairs of codes are used as left-out samples to test the model.

- The latent parameters of source codes are generated following the same steps as in Chapter 2. We summarize the steps here:
  - Generate the node degree heterogeneity parameters: $\alpha_i^{(t)} = -\tilde{\alpha}_i / \sum_{j=1}^{n} \tilde{\alpha}_j$, where $\tilde{\alpha}_i \overset{\text{i.i.d.}}{\sim} U[1,3]$. Generate $\beta_i^{(t)}$ with the same procedure.
  - Generate the latent positions: $v_i^{(s)} = \mu_v + N(0, I_k)$, where $\mu_v \overset{\text{i.i.d.}}{\sim} U[-1,1]$. Generate $w_i^{(s)}$ with the same procedure.
  - Generate the observed network: generate $X_{ij}^{(s)}$ with the parameters using the latent space zero-inflated Poisson model.
- The latent parameters $v_i^{(t)}$ and $w_i^{(t)}$ of the first L and last 1,000 target codes are set to be the transformation of the latent parameters of the corresponding source codes. The latent parameters $v_i^{(t)}$ and $w_i^{(t)}$ of other target codes and other undecided parameters (e.g., $\alpha_i^{(t)}$ and $\beta_i^{(t)}$) of the target codes are generated using the same generation procedure described above.

To simplify the setting, we let all latent positions have the same dimension $q = q_9 = q_{10} = 10$. For each experiment setting, we generate 20 independent copies of the network and then fit and evaluate the model on them.

Evaluation. The main goal of this joint latent space zero-inflated Poisson model is to align the latent positions of the source codes and the target codes, or in other words, to find a translation between them. Therefore, we use the precision@k measurement on the left-out samples to evaluate the models. The precision@k is the proportion of pairs for which the corresponding target code is in the k-th nearest neighbors of the source code. We vary k form $\{1, 3, 5, 10\}$.

General Performance. We evaluated the performance of our joint latent space zero-inflated Poisson model for a specific configuration of parameters: $L = 200, q = 10$, and

$\gamma = 0.1$. The true dimension q was used for the model. This configuration was chosen to show the general performance of our proposed model.

Under this setting, our model exhibited promising results in terms of the precision@k score, which measures how accurately the model can align the source codes and the target codes. The Table 3.1 shows the detailed results:

k	Precision@k	Standard Error
1	0.8857	0.0474
3	0.9218	0.0325
5	0.9326	0.0282
10	0.9499	0.0212

Table 3.1: Precision@k scores and standard errors

The results demonstrate the efficacy of our model in mapping the source codes and target codes. With a high precision of 0.9499 for $k = 10$ and consistently low standard errors, we are confident in the model's potential for real-world applications, such as ICD code translation. The model can also be useful in combining medical records and data with different versions of ICD codes.

Having established the baseline performance of our joint latent space zero-inflated Poisson model under a specific configuration, it's important to understand how the choice of model dimensions and the weight parameter γ can affect its efficacy. A suitable choice is necessary to ensure the model's precision, and robustness. In the subsequent sections, we investigate these parameters' impact on the model's performance and offer insights that can guide optimal parameter selection for real-world applications.

3.4.1 Effect of the Model Configuration

In this section, we show the effect of model configurations on the performance. Specifically, we vary the latent space dimensions and the weight parameter γ and show their effect on the performance.

Effect of the Latent Space Dimensions. While the true dimension is 10, in real-world applications, the actual latent space dimension is unknown. As such, we explore the model's performance under various dimension settings to simulate realistic scenarios and ensure the robustness of our model. We use the same procedure as in the previous section to generate model parameters. We show the results of the precision@k score on the left-out testing set for various dimension settings in Table 3.2 (the numbers in the brackets are corresponding standard errors).

Dimension	p@1 (standard error)	p@3 (standard error)	p@5 (standard error)	p@10 (standard error)
6	0.4927 (0.0527)	0.6783 (0.0523)	0.7459 (0.0476)	0.8229 (0.0404)
8	0.8039 (0.0220)	0.923 (0.0129)	0.9498 (0.0093)	0.973 (0.0057)
10	0.8857 (0.0474)	0.9218 (0.0325)	0.9326 (0.0282)	0.9499 (0.0212)
12	0.9995 (3.07e-4)	1 (0)	1 (0)	1 (0)
14	0.9985 (4.77e-4)	0.9999 (1e-4)	0.9999 (1e-4)	1 (0)

Table 3.2: Precision@k and standard error for different latent space dimensions

The results show the relationship between the number of dimensions and the model's precision. For dimensions below the true value (e.g., 6), the model appears to have insufficient capacity to capture all relationships, leading to worse performance. As we approach the true dimension (10), the precision improves significantly, indicating that the model is sufficient to represent and align the codes. This shows the necessity to have a relatively large number of dimensions so that the model could capture the complexity of the data.

Interestingly, increasing the dimension beyond the true value (12 and 14) leads to near-perfect precision scores. A larger latent space provided the model with additional capacity to represent the relationships and patterns between the source codes and the target codes. Moreover, a higher-dimensional latent space might be seen as a form of redundancy, which could potentially make our model more robust to variances and noise in the data. However, higher dimensions require more computational resources and could lead to overfitting issues.

In summary, choosing the right number of dimensions for the model is crucial. Based on the result, we would recommend selecting a relatively large dimension for better performance. However, the computational cost and overfitting issues also need to be taken into account.

Effect of the Weight γ. The weight γ also plays a crucial role in the joint latent space zero-inflated Poisson model, particularly in balancing the individual code characteristics and the relationship between the two versions of codes.

To study the impact of γ, we kept the number of matched codes L constant at 200 and used the true number of dimensions (10) for the latent spaces. We perform experiments with different values of γ, ranging from 0.001 to 10,000. Table 3.3 shows the performance (as measured by precision@k and the numbers in the brackets are corresponding standard errors) of the models with different values of γ.

γ	p@1 (standard error)	p@3 (standard error)	p@5 (standard error)	p@10 (standard error)
0.001	0.6972 (0.0619)	0.7734 (0.0501)	0.802 (0.0456)	0.8383 (0.0388)
0.01	0.8713 (0.0682)	0.9043 (0.0521)	0.9168 (0.0464)	0.934 (0.0382)
0.1	0.8857 (0.0474)	0.9218 (0.0325)	0.9326 (0.0282)	0.9499 (0.0212)
10	0.8251 (0.0887)	0.8509 (0.0846)	0.8616 (0.0832)	0.8756 (0.0811)
100	0.6977 (0.0901)	0.7588 (0.0854)	0.7798 (0.0835)	0.8112 (0.0815)
1000	0.4838 (0.1119)	0.6282 (0.1045)	0.6837 (0.0929)	0.7668 (0.0702)
10000	0 (0)	0 (0)	0 (0)	0 (0)

Table 3.3: Precision@k and standard error for different weights (γ)

The result shows a non-monotonic relationship between the performance and γ. While moderate values of γ yield high precision, low or high values lead to worse performance. In essence, γ allows the model to be more or less sensitive to the differences between the source and target codes, influencing the model's ability to align the two spaces. Careful adjustment of γ is needed to achieve optimal performance.

3.4.2 Effect of the Number of Matched Code Pairs L

The match code pairs are an important part of our model, as they provide valuable information on the relationship between source codes and target codes. It is necessary to

figure out how the number of matched code pairs would influence the model performance because high-quality matched pairs can be difficult and expensive to obtain in practice. In this section, we study the effect of the number of matched code pairs, L, on the model performance. We fix the true dimension and the model dimension at 10. Based on the result in the previous section, we fix γ at 0.1. We vary the number of matched code pairs L in $\{10, 50, 100, 150, 200, 250, 300\}$. Table 3.4 shows the results of the precision@k scores (the number in the brackets are corresponding standard errors) for different numbers of L.

L	p@1 (standard error)	p@3 (standard error)	p@5 (standard error)	p@10 (standard error)
10	0.2037 (0.0371)	0.2793 (0.0438)	0.3174 (0.0451)	0.3698 (0.0476)
50	0.9026 (0.0504)	0.9314 (0.0355)	0.9451 (0.0283)	0.9596 (0.0208)
100	0.8741 (0.0514)	0.9108 (0.0367)	0.9226 (0.0318)	0.9394 (0.0249)
150	0.9013 (0.0502)	0.9291 (0.0364)	0.9388 (0.0314)	0.9491 (0.0261)
200	0.8857 (0.0474)	0.9218 (0.0325)	0.9326 (0.0282)	0.9499 (0.0212)
250	0.8667 (0.0601)	0.9061 (0.0445)	0.9201 (0.0385)	0.9391 (0.0304)
300	0.8884 (0.0456)	0.9202 (0.0328)	0.9326 (0.0277)	0.9455 (0.0225)

Table 3.4: Precision@k for different numbers of matched code pairs (L)

When L is small, at 10, the model performs badly because of the sparse data. This indicates the model's need for a more comprehensive set of matched code pairs to learn

alignment representations robustly. However, as L increases to the range of 50 to 200, the model exhibits peak performance. Interestingly, as we go beyond an L of 200, the performance remains satisfactory but doesn't show significant improvement. This result shows the value of the right dataset size for efficient code alignment. After the model has enough data, increasing the sample size does not provide better performance. Therefore, it is more important to obtain a high-quality group of matched pairs instead of having a large number of matched pairs, which further guides our usage of data in the real data experiment.

3.5 Real Data Examples

In this section, we demonstrate the proposed method's ability to model and translate different versions of ICD codes using a real-world dataset, the Nationwide Readmissions Database (NRD). For each year, the dataset collects information for patients discharged from a hospital or died in a hospital. A part of the patients may have repeat visits in that year. The dataset addresses the lack of information on hospital readmission. We use the NRD dataset of the year 2015 (NRD 2015) because the hospitals in the US transited from ICD-9 to ICD-10 on October 1, 2015. We select patients who have visits both before and after the transition to ICD-10 and use the ICD codes in their medical records to fit the models. In total, we get 947,053 matched patients with 24,065 distinct ICD-10 codes and 10,137 ICD-9 codes. By matching the patients and constricting the data to the year 2015, we can minimize the bias and variation in patient population, disease prevalence, and ICD code usage. The NRD dataset is publicly available and has been used in various healthcare research (Kolte et al., 2017; Jacobs et al., 2018; Agrawal et al., 2018).

3.5.1 The General Equivalence Mappings

The General Equivalence Mappings (GEM) (Butler, 2007) provide a practical mapping between ICD-9 and ICD-10. The GEM was created by the National Center for Health Statistics

(NCHS), the Centers for Medicare and Medicaid Services (CMS), American Health Information Management Association (AHIMA), the American Hospital Association, and 3M Health Information Systems. Each ICD-9 code is provided with a translation in ICD-10 in the GEM, and vice versa. However, the map is not one-to-one. Only a part of the codes have a one-to-one mapping. Many codes are mapped to multiple codes in the GEM. For example, 255 ICD-9 codes are mapped to more than 50 ICD-10 codes, while there are more than 7,200 ICD-10 codes mapped to multiple ICD-9 codes (Wollman, 2011). Furthermore, most of the translations are marked as *approximate*, which means the translations are not exact. Only 3,533 translations are exact in the GEM version we currently use. As shown in the simulation, a small amount of high-quality translation is enough for a satisfactory performance. We expect the 3,533 exact matches would be enough for the proposed model to provide a good translation between ICD-9 and ICD-10 codes. We use the 3,533 exact matches for both the supervision and evaluation. More specifically, we put 70% of the exact matches in the supervision set, and the rest 30% are used for evaluation.

3.5.2 Data Preprocessing

In the database, each visit is notated by a set of ICD codes. The ICD-9 codes were used for visits before the transition on October 1, 2015, and the ICD-10 codes were used for visits after the transition. We define the co-occurrence times of a pair of ICD codes as the number of times that they appear in the same visit in the NRD 2015. We construct two co-occurrence matrices by counting the pairwise co-occurrence times of all ICD codes, one for the ICD-9 and one for the ICD-10.

3.5.3 Experiment Design

We apply our proposed model to the co-occurrence matrices of the two versions of ICD codes. 70% of the exact matches in GEM are used for supervision. The 30% left out matches are used for evaluation, and we report the precision@10 measurement.

Hyper-parameters	Values
Dimension d	10, 16, 24, 32, 64, 128
Weight γ	0.001, 0.01, ..., 1000

Table 3.5: The selection of hyper-parameters d and γ

A popular way to generate numerical representations of ICD codes in literature is by borrowing word embedding models from natural language processing (NLP). For instance, the work in Choi et al. (2017a) utilized the GloVe (Pennington et al., 2014) model, while Nguyen et al. (2018); Choi et al. (2017b), and Feng et al. (2017) applied the Skip-gram (Mikolov et al., 2013) model. Therefore, we also apply the GloVe model and the Skip-gram model for this task and compare our proposed model with them. A similar transformation matrix is learned for the word embedding models.

A special case of our proposed method is also compared. We replaced the latent space zero-inflated Poisson model in our model with the ordinary latent space Poisson model. This special version is simpler and trains faster.

Table 3.5 shows the selection of hyper-parameters for the models. The best combination of hyper-parameters is determined by cross-validation.

3.5.4 Results

We show the precision@10 score, selected hyper-parameters, and the training time until convergence for each model in Table 3.6.

Model	p@10	Dim	Weight γ	Training Time (until convergence)
Skip-Gram	0.446	16	10.0	382 mins
Glove	0.458	16	100.0	411 mins
Joint Poisson	0.490	32	10.0	**31 mins**
Joint zero-inflated Poisson	**0.518**	24	100.0	**44 mins**

Table 3.6: The Model Performance (evaluated by precision@10), training time, and hyper-parameters selected by cross-validation for each model.

We can see that our proposed model (labeled as "Joint zero-inflated Poisson") has the highest precision@10 score and a short training time. The simplified version, the joint Poisson model, outperforms the word embedding models as well.

3.5.5 Summary

By outperforming existing models, our approach successfully demonstrates its proficiency in learning vector representations of both ICD-9 and ICD-10 and mapping ICD code to the other code system.

Firstly, the successful translation between ICD-9 and ICD-10 codes, achieved by our model, addresses a critical need in the healthcare community. Because of the transition in ICD code systems, institutions, and professionals are often faced with the task of mapping historical data and applications (coded in ICD-9) to newer systems (coded in ICD-10) (Topaz et al., 2013; Khera et al., 2018; Wollman, 2011). Our model can be easily applied to this process, reducing the effort and potential inaccuracies associated with manual or rule-based translations.

Besides a typical one-to-one translation, our proposed model has its unique ability to represent ICD codes in the latent spaces of the other ICD code version. This is particularly important given that exact translations between ICD-9 and ICD-10 don't always exist due to

the differences and updates in the ICD code systems. Instead of direct translations, which can lead to potential ambiguities or inaccuracies, our approach provides a more flexible representation. By mapping the codes into latent spaces, we allow for a deeper understanding and comparison of the codes.

In practice, the mapped latent vectors can be used in downstream tasks directly without being restricted to finding a one-to-one transition. the vector representations learned by our model can serve multiple secondary purposes. Potentially applications include clustering, anomaly detection, and predictive modeling where ICD codes serve as inputs. Our proposed model is helpful in these applications by providing high-quality, consistent, and interpretable input data.

3.6 Discussion

In this chapter, we propose a joint latent space zero-inflated Poisson approach. The model is built on the latent space zero-inflated Poisson model and addresses the lack of a method that can map different versions of ICD codes to each other.

The proposed model consists of two latent space zero-inflated Poisson models and a linear transformation. It learns embeddings for both ICD-9 codes and ICD-10 codes simultaneously. The transformation is trained to map ICD codes into the latent space of the other ICD system. Guided by a high-quality dictionary of ICD code pairs, the transformation has the ability to map ICD codes with similar medical meanings to close latent positions.

The vector representations and transformations learned by our model are useful in many healthcare applications. For instance, the vector representations can be used as features in downstream tasks, such as risk prediction, just like the examples in Chapter 2 and existing works (Shi et al., 2021; Nguyen et al., 2018; Choi et al., 2017b). Furthermore, the learned transformation is useful for ICD code translation and understanding the relationship and interactions among diseases. The ability to map ICD-9 codes to the ICD-10 system could

help healthcare researchers merge historical medical data with ICD-9 codes to align with the up-to-date data with ICD-10 codes. This helps with overcoming the common challenge of the lack of enough samples in healthcare research. Also, clinical practices and applications built with ICD-9 codes can be transferred to the newer system easily with our transformation.

CHAPTER 4

A Novel Estimation Method for the Latent Space DiPH Model Using Mixed Likelihood

4.1 Introduction

Although the latent space zero-inflated Poisson model achieves good performance in modeling ICD codes and demonstrated its application value in predictive tasks and the ICD code mapping task, it focuses solely on pairwise co-occurrences of ICD codes. Similarly, the global vectors for word representation (GloVe) (Pennington et al., 2014), a popular word embedding model that has been applied to ICD codes in literature, is also trained on the co-occurrence matrix. Using a co-occurrence matrix simplifies network datasets and provides satisfactory results. However, it also loses an essential kind of information, the higher-order relationship among multiple ICD codes. Literates demonstrate that the higher-order relationship plays a crucial role in network analysis (Greening Jr et al., 2015; Karwa and Petrović, 2016; Chodrow, 2020). In addition, patients with the same disease might present different risks of developing secondary diseases based on other medical conditions (Sánchez-Valle et al. (2020)). The work in Goh et al. (2007) shows more than five different diseases can be associated with the same gene, indicating the existence of higher-order associations. Therefore, an approach that takes the high-order interactions into account is needed.

Hypergraph (Berge, 1970), an extension of the dyadic network allowing edges to connect more than two nodes, has attracted lots of research interest. Like a dyadic network, a

hypergraph also consists of a node set $\mathcal{V}$ and a set of edges $\mathcal{E}$, where each edge in $\mathcal{E}$ is a hyperedge. Each hyperedge contains a set of nodes in $\mathcal{V}$. The hyperedges in a hypergraph can have any number of nodes. Many models have been developed to learn hypergraph embedding (Zhou et al., 2006; Tu et al., 2018; Zhen and Wang, 2023; Yu and Zhu, 2023). This chapter focuses on one of these models, the Diversity and Popularity Hypergraph (DiPH) proposed in Yu and Zhu (2023), and proposes a new fitting algorithm for it.

The DiPH model is a latent space model based on the determinantal point processes (Kulesza et al., 2012), which is designed to capture the diversity among nodes in hyperedges and the node's individual popularity simultaneously. In fact, it is common to observe a high degree of diversity in many real-world networks. For example, individuals from diverse backgrounds collaborate on tasks that require a broad skill set; stores offer a variety of goods to meet the varied needs of diverse customer groups. Therefore, in contrast to many other models that are driven by similarity, the DiPH model considers diversity within hyperedges.

The paper Yu and Zhu (2023) estimates the DiPH model by maximum likelihood estimation (MLE). We observe that MLE is numerically unstable for the DiPH model. Specifically, the estimation obtained through MLE exhibits high variance, particularly when the number of hyperedges is relatively small. In addition, the estimation error increases significantly as the number of hyperedges decreases. To address this, we introduce the mixed likelihood function, the combination of the ordinary likelihood function and pseudo-likelihood function. We define the pseudo-likelihood function based on conditional probabilities. Specifically, it is defined by the probability of observing a particular hyperedge given the presence of a certain subset of nodes within that hyperedge. Essentially, each term in the pseudo-likelihood function considers a sub-graph of the hypergraph. This approach is shown to be more stable and provides more accurate estimations. The main contributions of this chapter are the following.

- We propose a new fitting algorithm using the mixed likelihood for the Diversity and Popularity Hypergraph (DiPH) model. The mixed likelihood combines the ordinary

log-likelihood with pseudo-log-likelihood. This approach provides more stable and accurate estimations compared to using the ordinary likelihood alone.

- We establish the theoretical guarantees for the consistency of the mixed likelihood estimation, indicating that the introduction of pseudo-likelihood would not affect the consistency of the model.

- In simulation studies, compared with the original estimation method using MLE, the proposed approach achieves smaller estimation error and stabler performance, especially in high-dimensional and sparse data scenarios.

- Utilizing the MIMIC-III database, we conduct a real-world data experiment that applies the DiPH model to a hypergraph of ICD codes constructed from EHR data. The learned latent positions are used as predictors in a predictive task. The proposed method achieves the best prediction performance compared with MLE estimation and other embedding methods, demonstrating its practical value.

- To the best of our knowledge, this is the first approach to fit a model with the combination of ordinary likelihood and pseudo-likelihood. Because of its high degree of adaptability, it has potential application to a variety of models beyond the DiPH model.

The rest of the chapter is organized as follows. In Section 4.2, we review the DiPH model and then propose the mixed likelihood function, and introduce the fitting method. In Section 4.3, we provide a theoretical guarantee for the consistency of the estimation. In Section 4.4, we conduct a series of simulation studies that validate the effectiveness of the mixed likelihood method, examining its performance under various data conditions and model configurations and demonstrating its improvement over the original estimation method. Section 4.5 applies the proposed method to the MIMIC-III dataset and shows its practical utility. The chapter concludes with Section 4.6, where we summarize our findings and suggest directions for future research.

4.2 Model

In this section, we first introduce necessary notations and review the Diversity and Popularity Hypergraph (DiPH) model (Yu and Zhu, 2023). Then we propose the mixed likelihood function for the DiPH model. The mixed likelihood function consists of the ordinary likelihood function and the pseudo-likelihood function. A weight parameter w is introduced to balance the two likelihood functions. We then present the conditions for identifiability and the fitting method.

4.2.1 Notation

We use $[n]$ to denote the set $\{1, 2, ..., n\}$ for any positive integer n. For a hypergraph with n_v nodes, we index the nodes using positive integers and let $\mathcal{V} = [n_v]$ be the set of nodes. Let $\mathcal{E} = \{e_1, e_2, ..., e_{n_e}\}$ be the set of observed hyperedges, where n_e is the total number of hyperedges and $e_l \subset \mathcal{V}$ for any $l \in [n_e]$.

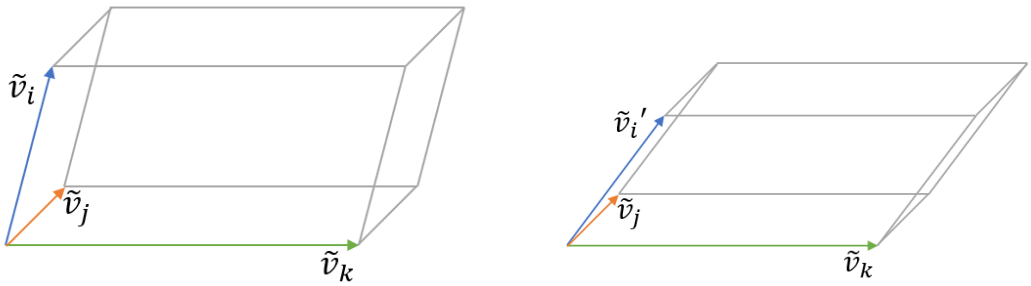
4.2.2 The Diversity and Popularity Hypergraph (DiPH) Model

In this section, we review the Diversity and Popularity Hypergraph (DiPH) model proposed in Yu and Zhu (2023), which is driven by the diversity within hyperedges and heterogeneous popularity among individual nodes.

The DiPH model associates each node i ($i \in \mathcal{V} = [n_v]$) with a latent position $v_i \in \mathbb{R}^d$ ($0 < d < n_v$) and a popularity parameter $\alpha_i > 0$. The model assumes each hyperedge of a hypergraph independently follows a distribution $\mathcal{P}$ over all subsets of the set of nodes $[n_v]$. Specifically, let $\{e_1, e_2, ..., e_{n_e}\}$ be the set of hyperedges of a hypergraph with n_v nodes, the DiPH model assumes

$$e_k \overset{i.i.d.}{\sim} \mathcal{P} \text{ for } k = 1, 2, ..., n_e. \tag{4.1}$$

The distribution $\mathcal{P}$ is generated by the parameters $\{v_i, \alpha_i | i \in \mathcal{V}\}$ with a determinantal point



(a) (b)

Figure 4.1: (a) The parallelotope formed with $\tilde{v_i}$, $\tilde{v_j}$, and $\tilde{v_k}$. (b) The parallelotope formed with $\tilde{v_i}'$, $\tilde{v_j}$, and $\tilde{v_k}$. Note $\tilde{v_i}$ and $\tilde{v_i}'$ have the same length, but the volume of the parallelotope in (a) is larger than (b). Therefore, the volume of the parallelotope is determined by both the length and spread of the vectors.

process (DPP) (Kulesza et al., 2012) as follows. Let w_i be the $i-$th standard basis vector of $\mathbb{R}^{n_v}$, and define $\tilde{v_i} = (v_i, \sqrt{\alpha_i} w_i) \in \mathbb{R}^{d+n_v}$. We can see the structure of $\tilde{v_i}$'s better in the matrix form:

$$\begin{pmatrix} \tilde{v_1} \\ \tilde{v_2} \\ \dots \\ \tilde{v_{n_v}} \end{pmatrix} = \begin{pmatrix} v_1 & \sqrt{\alpha_1} & 0 & 0 & \dots & 0 \\ v_2 & 0 & \sqrt{\alpha_2} & 0 & \dots & 0 \\ & & \dots & & & \\ v_{n_v} & 0 & 0 & 0 & \dots & \sqrt{\alpha_{n_v}} \end{pmatrix}. \tag{4.2}$$

For any hyperedge $e \subset [n_v]$, the probability of observing it is related to the parallelotope formed with vectors $\{\tilde{v_i} | i \in e\}$ (see Figure 4.1 for more detail). Specifically, the distribution $\mathcal{P}$ is given by

$$P(E = e) \propto \text{vol}^2(\{\tilde{v_i} | i \in e\}), \tag{4.3}$$

where $\text{vol}(\{\tilde{v_i} | i \in e\})$ is the volume of the parallelotope. In order to compute the volume

more easily, define

$$L_{n_v \times n_v} = (\tilde{v_i}^T \tilde{v_j})_{i,j=1}^{n_v}. \tag{4.4}$$

Then the right hand side of (4.3) is given by

$$\text{vol}^2(\{\tilde{v_i} | i \in e\}) = \det(L_e),$$

where L_e is the submatrix of L with rows and columns with indexes in e and $\det(\cdot)$ is the determinant of a matrix (Anderson, 1958). Using mathematical induction, we can show

$$\sum_{e \subset [n_v]} \det(L_e) = \det(L + I).$$

Therefore, we have

$$P(E = e) = \frac{\det(L_e)}{\det(L + I)}, \tag{4.5}$$

for any hyperedge $e \subset [n_v]$. The DiPH model is defined by (4.1), (4.2), (4.4), and (4.5). The model is written as $H(v_1, v_2, ..., v_{n_v}, \alpha)$, or $H(L)$.

In order to ensure identifiability, the DiPH model requires that all v_i's share the same (unknown) norm, i.e.

$$\|v_i\|_2 = \|v_{i'}\|_2 > 0 \text{ for any } i, i' \in [n_v].$$

For a given hypergraph with an observed set of hyperedges $\mathcal{E} = \{e_1, e_2, ..., e_{n_e}\}$, the

log-likelihood function of the DiPH model is

$$\begin{aligned}\ell =& \frac{1}{n_e}\sum_{k=1}^{n_e}\log(P(E=e_k))\\ =& \frac{1}{n_e}\sum_{k=1}^{n_e}\log\left(\frac{\det(L_{e_k})}{\det(L+I)}\right)\\ =& \frac{1}{n_e}\sum_{k=1}^{n_e}\operatorname{logdet}(L_{e_k})-\operatorname{logdet}(L+I).\end{aligned} \tag{4.6}$$

Therefore, one can apply maximum likelihood estimation (MLE) to estimate the parameters by solving

$$\underset{\substack{v_i\in\mathbb{R}^d,\alpha_i>0,\\ \|v_i\|_2 \text{ is a constant over } i}}{\arg\max} \frac{1}{n_e}\sum_{k=1}^{n_e}\operatorname{logdet}(L_{e_k})-\operatorname{logdet}(L+I). \tag{4.7}$$

The optimization problem (4.7) can be solved by the projected gradient descent algorithm (Lange and Lange, 2013). We refer readers to Yu and Zhu (2023) for more details on the identifiability, estimation, and other properties of the DiPH model.

4.2.3 The Mixed Likelihood

In this section, we define the pseudo-log-likelihood for the DiPH model and the mixed likelihood, which combines the ordinary log-likelihood and the pseudo-log-likelihood.

To define the pseudo-log-likelihood, we first present the conditional probability for the DiPH model. For any hyperedges $e_1 \subset e_2 \subset [n_v]$ and a random hyperedge E that follows the distribution $\mathcal{P}$ as defined in (4.3), we have

$$P(E=e_2|e_1\subset E)=\frac{\det(L_{e_2})}{\det(L+I-I_{e_1})},$$

where I_{e_1} is an n_v-by-n_v diagonal matrix whose diagonal elements are either one or zero. The i-th diagonal element of I_{e_1} is one if and only if $i \in e_1$.

We define the pseudo-log-likelihood as

$$\begin{aligned}\ell_{\mathrm{p}} &= \frac{1}{n_e}\sum_{k=1}^{n_e}\sum_{e\subset e_k}\log(P(E=e_k|e\subset E)) \\ &= \frac{1}{n_e}\sum_{k=1}^{n_e}\sum_{e\subset e_k}\left(\mathrm{logdet}(L_{e_k})-\mathrm{logdet}(L+I-I_e)\right). \end{aligned} \tag{4.8}$$

For an observed hyperedge e_k, the inner summation is taken over all subset $e \subset e_k$. It considers all possible subsets of a hyperedge, capturing the diverse relationships among the nodes in the hyperedge. The pseudo-log-likelihood function measures the likelihood of observing a hyperedge given partial information about the hyperedge (i.e., a part of nodes in the hyperedge). It is important to note that conditioning on $e_1 \subset E$, the distribution of other nodes in the hyperedge $(E \bigcap e_1^c)$ also follows a DiPH model on the node set e_1^c, the complement set of e_1 in $\mathcal{V}$. Essentially, each term in the pseudo-log-likelihood function measures a sub-graph of the original hypergraph, capturing information from a different perspective.

Recall the ordinary log-likelihood is

$$\ell = \frac{1}{n_e}\sum_{k=1}^{n_e}\mathrm{logdet}(L_{e_k})-\mathrm{logdet}(L+I).$$

We define the mixed likelihood as

$$\ell_{\mathrm{mixed}} = w\times\ell+\ell_{\mathrm{p}}, \tag{4.9}$$

where $w > 0$ is a weight parameter. The parameter w balances the weight of the ordinary log-likelihood and the pseudo-log-likelihood and introduces flexibility to our approach. By introducing the pseudo-log-likelihood and combining it with the ordinary log-likelihood, the mixed likelihood captures a more comprehensive and detailed understanding of hypergraphs.

4.2.4 Estimation Method

In this section, we propose a new estimation method for the DiPH model using the mixed likelihood ℓ_{mixed} defined in (4.9). We employ the maximum likelihood estimation (MLE) to the mixed likelihood, analogous to the estimation method in the original DiPH paper (Yu and Zhu, 2023). We reparameterize v_i's using $V \in \mathbb{R}_{n_v \times d}$ and β by

$$V_{i\cdot} := \frac{v_i}{\|v_i\|_2}, \text{and } \beta := \|v_i\|_2^2.$$

Recall that

$$L = (\tilde{v_i}^T \tilde{v_j})_{i,j=1}^{n_v} = (v_i^T v_j)_{i,j=1}^{n_v} + \text{diag}(\alpha) = \beta V V^T + \text{diag}(\alpha).$$

Then the mixed likelihood becomes

$$\begin{aligned}
\ell_{\text{mixed}} &= \frac{w}{n_e}\left(\sum_{k=1}^{n_e} \text{logdet}(L_{e_k}) - \text{logdet}(L+I)\right) \\
&\quad + \frac{1}{n_e}\sum_{k=1}^{n_e}\sum_{e\subset e_k}\left(\text{logdet}(L_{e_k}) - \text{logdet}(L+I-I_e)\right) \\
&= \frac{w}{n_e}\left(\sum_{k=1}^{n_e} \text{logdet}((\beta V V^T)_{e_k} + \text{diag}(\alpha)_{e_k}) - \text{logdet}(\beta V V^T + \text{diag}(\alpha) + I)\right) \\
&\quad + \frac{1}{n_e}\sum_{k=1}^{n_e}\sum_{e\subset e_k}\left(\text{logdet}((\beta V V^T)_{e_k} + \text{diag}(\alpha)_{e_k}) - \text{logdet}(\beta V V^T + \text{diag}(\alpha) + I - I_e)\right).
\end{aligned}$$

We estimate the parameters by solving the optimization problem

$$\underset{\substack{V\in\mathbb{R}_{n_v\times d}, \|V_{i\cdot}\|_2=1, \\ \alpha_i>0, \beta>0}}{\arg\max} \quad \ell_{\text{mixed}} \tag{4.10}$$

using the projected gradient descent algorithm (Lange and Lange, 2013). In each iteration, the algorithm first performs a gradient update for the estimation of the parameters and then projects the parameters into constraint sets to satisfy the model constraints. Specifically, the algorithm projects $V_{i\cdot}$ into the unit sphere and projects α_i into $\mathbb{R}^+$. To make the estimation

stabler and converge faster, we recommend fixing β and tuning it as a hyperparameter.

4.3 Theoretical Results

In this section, we present the consistency for the mixed likelihood estimation of the DiPH model.

Consider a DiPH model $H(v_1^*, v_2^*, ..., v_{n_v}^*, \alpha^*)$ with $v_i^* \in \mathbb{R}^d, \alpha_i > 0$, and a set of hyperedges $\mathcal{E} = \{e_1, e_2, ..., e_{n_e}\}$ generated from the DiPH model. Let $\hat{v}_1, \hat{v}_2, ..., \hat{v}_{n_v}, \hat{\alpha}$ be the parameters estimated by the optimization problem (4.10). Moreover, let $\hat{L}$ be the MLE for $L^* = (v_i^{*T} v_j^*)_{i,j=1}^{n_v} + \text{diag}(\alpha^*)$ defined as

$$\hat{L} := \underset{L \in \mathcal{L}}{\arg\max}\ \ell_{\text{mixed}}, \tag{4.11}$$

where $\mathcal{L}$ is the feasible space of L:

$$\mathcal{L} = \{L = (v_i^T v_j)_{i,j=1}^{n_v} + \text{diag}(\alpha) | v_i^* \in \mathbb{R}^d, \alpha_i > 0, \text{ and } \|v_i\|_2 > 0 \text{ is constant}\}.$$

4.3.1 Consistency

We use the same metrics as in Yu and Zhu (2023) to measure the error of estimation. Because the latent positions v_i's are identifiable up to an orthogonal transformation and individual sign flips, we use $\min_{O \in \mathbb{O}_d, s_i = \pm 1} \sum_{i=1}^{n_v} \|\hat{v}_i - s_i O v_i^*\|_2$ to measure the error of $\hat{v}_i$'s, where $\mathbb{O}_d$ is the set of d-dimension orthogonal matrices. Similarly, because L is identifiable up to row and column sign flips, we use $\min_{S \in \mathbb{D}_{n_v}} \|\hat{L} - SL^*S\|_F$ to measure the error of $\hat{L}$, where $\mathbb{D}_{n_v}$ is the set of n_v-dimension matrices with all diagonal entries being 1 or -1. Theorem 4.3.1 gives the consistency of the proposed estimation method using the mixed likelihood.

Theorem 4.3.1 *Consider a DiPH model $H(v_1^*, v_2^*, ..., v_{n_v}^*, \alpha^*)$, with the set of hyperedges $\mathcal{E} = \{e_1, e_2, ..., e_{n_e}\}$ independently generated from the DiPH model. Let $\hat{v}_1, \hat{v}_2, ..., \hat{v}_{n_v}, \hat{\alpha}, \hat{L}$ be*

the estimated parameter using the optimization problem (4.10) and (4.11). When $n_v > 2d$ *and* $\{v_1^*, v_2^*, ..., v_{n_v}^*\}$ *span* $\mathbb{R}^d$*, we have*

$$\min_{O \in \mathbb{O}_d, s_i = \pm 1} \sum_{i=1}^{n_v} \|\hat{v}_i - s_i O v_i^*\|_2 \xrightarrow{p} 0,$$

$$\|\hat{\alpha} - \alpha^*\|_2 \xrightarrow{p} 0,$$

$$\min_{S \in \mathbb{D}_{n_v}} \|\hat{L} - SL^*S\|_F \xrightarrow{p} 0,$$

as $n_e \to \infty$*.*

Despite introducing the pseudo-log-likelihood, our proposed estimation method with mixed likelihood keeps the same consistency property as the original DiPH estimation method. As mentioned in section 4.2.3, each term in the pseudo-log-likelihood is actually a term in the log-likelihood function of a sub-graph. Therefore, it is expected that the new fitting algorithm using the combination of ordinary likelihood and pseudo-log-likelihood would maintain the consistency that the original fitting algorithm holds. In addition, the following Theorem 4.3.2 provides the same consistency guarantee for estimation generated by solely maximizing the pseudo-log-likelihood ℓ_p.

Theorem 4.3.2 *Consider a DiPH model* $H(v_1^*, v_2^*, ..., v_{n_v}^*, \alpha^*)$*, with the set of hyperedges* $\mathcal{E} = \{e_1, e_2, ..., e_{n_e}\}$ *independently generated from the DiPH model. Let* $\hat{v}_1', \hat{v}_2', ..., \hat{v}_{n_v}', \hat{\alpha}', \hat{L}'$ *be the estimated parameters by maximizing the pseudo-log-likelihood* ℓ_p*. Under the same assumption as in Theorem 4.3.1, we have*

$$\min_{O \in \mathbb{O}_d, s_i = \pm 1} \sum_{i=1}^{n_v} \|\hat{v}_i' - s_i O v_i^*\|_2 \xrightarrow{p} 0,$$

$$\|\hat{\alpha}' - \alpha^*\|_2 \xrightarrow{p} 0,$$

$$\min_{S \in \mathbb{D}_{n_v}} \|\hat{L}' - SL^*S\|_F \xrightarrow{p} 0,$$

as $n_e \to \infty$*.*

Theorem 4.3.1 and Theorem 4.3.2 provide confidence in the estimation yielded by the proposed method. The proof of Theorem 4.3.1 and Theorem 4.3.2 is established utilizing the result in Van der Vaart (2000) and is given in the Appendix.

4.4 Simulation Studies

In this section, we present some simulation studies and evaluate the performance of the estimation method using the mixed likelihood function. The simulation studies are designed to compare the proposed method against the original estimation method proposed in the DiPH paper (Yu and Zhu, 2023). We summarize our simulation settings as follows:

- **Nodes and hyperedges**: We set the number of nodes (n_v) to 100 and vary the number of hyperedges (n_e) from 100 to 3,000.
- **Latent space dimensions**: The dimension of the latent space (d) is varied among $\{2, 3, 4, 6, 8\}$.
- **Parameter generation**: latent positions $v_1^*, ..., v_{n_v}^*$ are uniformly and independently selected from the unit sphere. The popularity parameter α_i^*'s are generated by $\sqrt{\alpha_i^*} = 0.15\gamma_i + 0.05$, where γ_i's independently follow the Beta$(1, 4)$ distribution. The density function of α is given in Figure 4.2. Most nodes will have a relatively small popularity parameter, while a few nodes will have a large popularity parameter.
- **Hypergraph generation**: we independently generate n_e hyperedges $\mathcal{E} = \{e_1, e_2, ..., e_{n_e}\}$ from the model $H(v_1^*, v_2^*, ..., v_{n_v}^*, \alpha^*)$.
- **Number of simulations**: For each different setting, we generated 10 independent hypergraphs to ensure the robustness of our results.
- **Subset selection in pseudo-log-likelihood**: due to the high computational complexity involved in traversing all subsets within a hyperedge (totaling 2^t different subsets, where t is the number of nodes in a hyperedge), we introduce a sampling approach

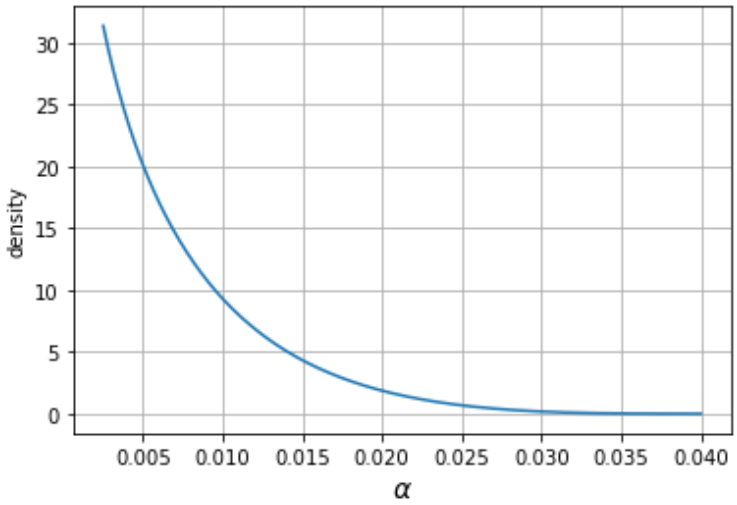


Figure 4.2: The density function of α

to address the problem. Instead of considering all subsets, we randomly sample a part of them for the calculation. We vary the number of subsets sampled from {t, 2t, 4t, 8t}. This method significantly reduced the computational load while maintaining the integrity of the estimation process.

We then estimate the parameters v_i^*'s and α_i^*'s using both the proposed estimation method and the original estimation method for the DiPH model. In simulation studies, we assume the length of latent vectors ($\beta = 1$) is known and use the true value of β in both approaches. To compare the performance of the two approaches, we use the relative errors of v_i's and α_i's. The relative error of α_i's is defined as

$$l(\hat{\alpha}, \alpha) = \text{mean}_i(|\frac{\hat{\alpha}_i - \alpha_i}{\alpha_i}|).$$

And the relative error of v_i's is

$$l(\hat{V}, V) = \frac{\min_{O \in \mathbb{O}_d, S \in \mathbb{D}_{n_v}} \|\hat{V} - SVO\|_F}{\|V\|_F},$$

where O is a d-by-d orthogonal matrix and S is an n_v-by-n_v diagonal matrix with diagonal

entries in $\{-1, 1\}$.

4.4.1 Performance in the estimation of V

Figure 4.3 shows the relative error of V across different dimensions (3,4, and 6) and numbers of hyperedges m for the DiPH model using both our proposed estimation method and the original approach (labeled as DiPH). For our proposed method, we vary the subset sampling sizes from $\{t, 2t, 4t, 8t\}$. The results are labeled as DiPH_mix_x, where x is the sample size.

In all dimensions, the proposed estimation method with mixed likelihood consistently outperforms the original approach. The mixed likelihood estimation method also shows less variability, suggesting it is more consistent and stable.

For both approaches, increasing the number of hyperedges m tends to decrease the relative error. Therefore, having more hyperedges benefits both methods. Notably, the mixed likelihood is more stable across different numbers of hyperedges, indicating the mixed likelihood method can already achieve good performance even with a small number of hyperedges. Also, as the dimension becomes larger, where the complexity of the model increases, the mixed likelihood method shows increasing advantages over the original method.

These findings indicate that the mixed likelihood method is more reliable in conditions of limited information (e.g., a small number of hyperedges or a complex model with a high dimension). The improved performance in these challenging scenarios demonstrates the practical applicability of the mixed likelihood method, making it a valuable tool for analyzing hypergraphs in real-world situations where data might be complex or sparse.

When the subset sampling size increases from t to $8t$, there is no significant change in the estimation performance, indicating the proposed method is not sensitive to the sampling size.

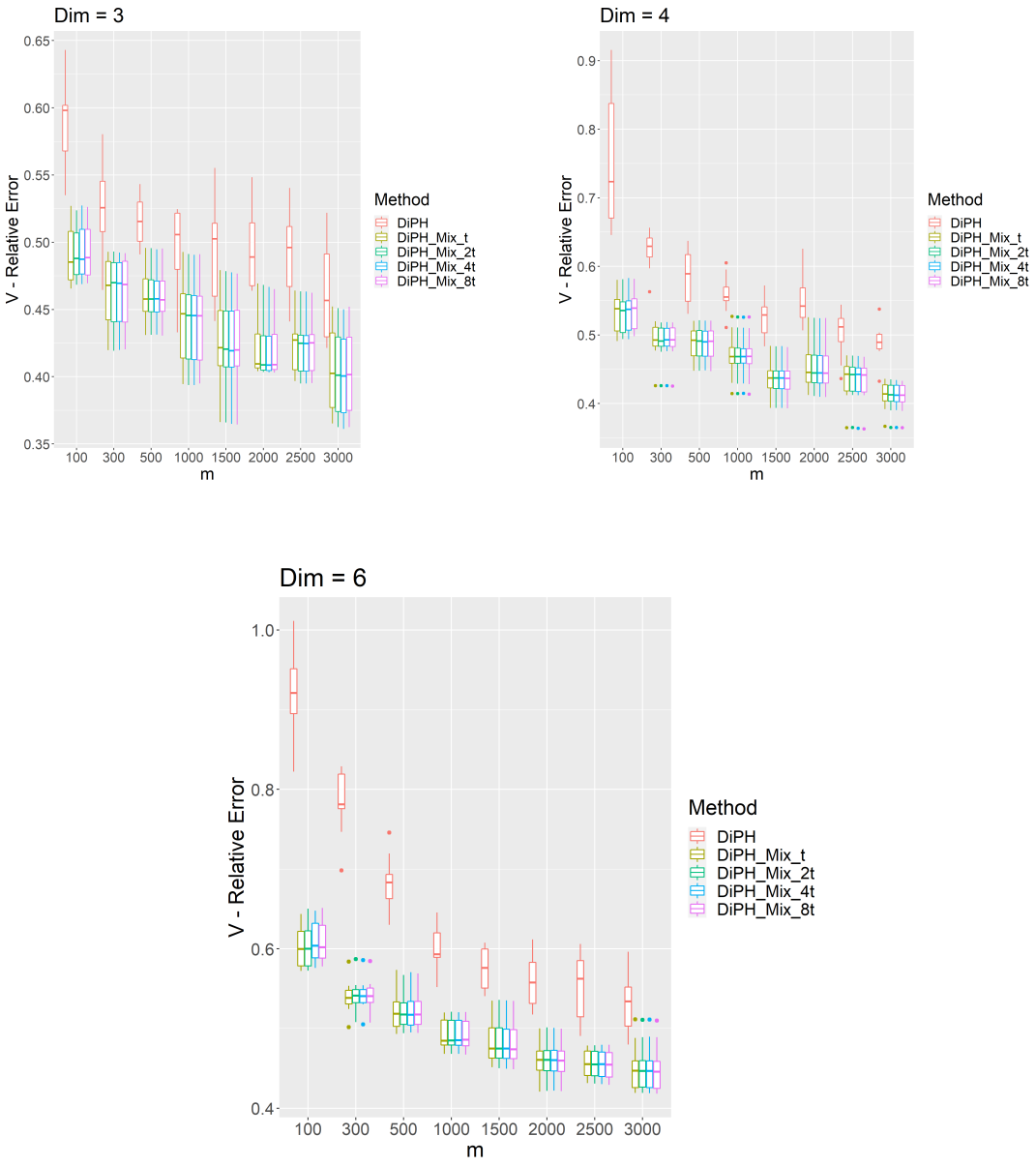


Figure 4.3: Relative error of V. Left (Row 1): the relative error of V for model dimension $d = 3$; Right (Row 1): the relative error of V for model dimension $d = 4$; Middle (Row 2): the relative error of V for model dimension $d = 6$.

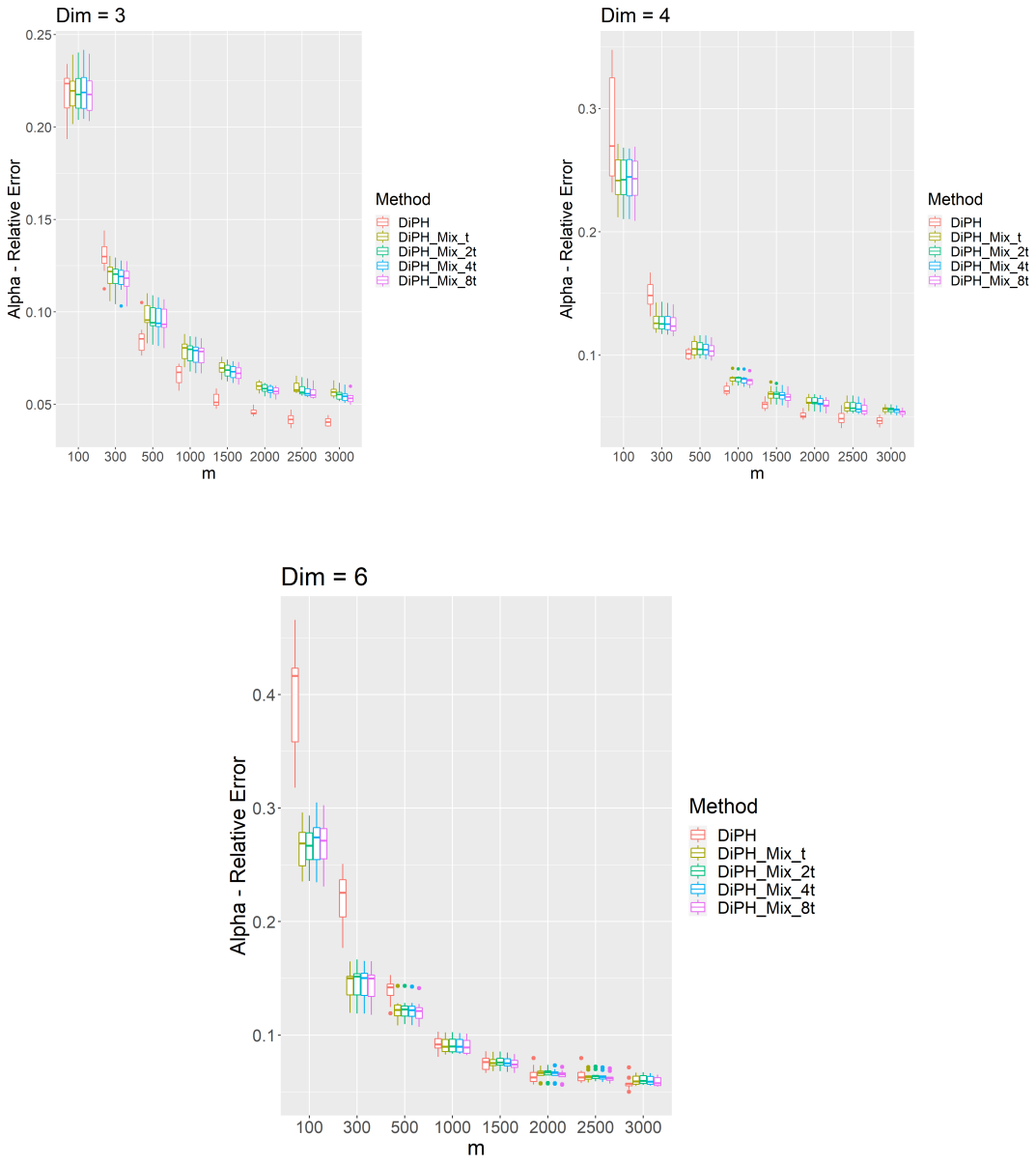


Figure 4.4: Relative error of α_i's. Left (Row 1): the relative error of α_i's for model dimension $d = 3$; Right (Row 1): the relative error of α_i's for model dimension $d = 4$; Middle (Row 2): the relative error of α_i's for model dimension $d = 6$.

4.4.2 Performance in the estimation of α's

Figure 4.4 shows the estimation error of α's. As the number of hyperedges m increases, the relative error for both methods decreases, which indicates that more data provides a more accurate estimation of the popularity parameter and matches the observation of the estimation of V.

In the case of smaller numbers of hyperedges (m), there is a significant advantage of the mixed likelihood estimation over the original estimation method, especially in higher dimensions. This aligns with the idea that the mixed likelihood approach is particularly beneficial in scenarios where the available information is limited.

In conclusion, the mixed likelihood estimation method, with the subset sampling strategy, is a robust and efficient alternative to the original DiPH model estimation. It offers improved accuracy and consistency, particularly in data-sparse and high-dimensional scenarios. It is expected the mixed likelihood estimation method has the potential for widespread application in complex hypergraph modeling.

4.5 Real Data Examples

In this section, we illustrate the practical utility of the mixed likelihood estimation method by applying it to a real-world electronic health record (EHR) dataset, the MIMIC-III database (Johnson et al., 2019).

4.5.1 The MIMIC-III database

The MIMIC-III database is a comprehensive collection of EHR data from over 50 thousand ICU admissions at the Beth Isreal Deaconess Medical Center between 2001 and 2012 (Johnson et al., 2016, 2019; Goldberger et al., 2000). It provides detailed information about the admissions, such as demographics, laboratory test results, and detailed treatment records.

Each admission is annotated with a group of ICD-9 (International Classification of Diseases, the 9th revision) codes. These codes capture important information about patient diagnoses and treatments, making them invaluable for healthcare data analysis.

When using the ICD codes in healthcare research, the conventional approach is to borrow word embedding techniques from natural language processing (Shi et al., 2021; Nguyen et al., 2018; Choi et al., 2017b; Feng et al., 2017; Choi et al., 2017a; Cai et al., 2018). Some of those models rely on pairwise co-occurrences of ICD codes. However, these methods have significant drawbacks.

Word embedding models from natural language processing are primarily designed for the human language, which differs substantially from ICD codes. For example, words in a human language sentence have order, while ICD codes are typically treated as a set. Moreover, the meaning of a word can be ambiguous, but ICD codes are designed to be precise and specific. Also, ICD codes have a well-structured tree hierarchy, which words do not have. Therefore, those word embedding models may not be the most suitable model to capture the full clinical context and the complicated relationships that exist between different diagnoses.

On the other hand, focusing solely on pairwise co-occurrences of ICD codes results in the loss of substantial information. Frequently, more than two medical conditions interact with each other. Considering just pairwise information is not enough to capture these higher-order interactions. In contrast, the DiPH model treats ICD codes in the EHR database as a hypergraph. The ability to learn higher-order information makes it possible for us to develop more accurate predictive models and tools that reflect the complexity of real-world clinical scenarios.

In the following real data example, we apply the DiPH model to a similar predictive task as the one in Chapter 2. We demonstrate the practical value of the proposed estimation method by comparing it with other approaches in the predictive task.

4.5.2 Data Processing

We follow the way in Chapter 2 to process the MIMIC-III dataset. Specifically, the group of diagnosis ICD codes and nine variables related to the patients (e.g., age, gender, ethnicity, number of ICU stays) are collected and used as predictors. We remove admissions of patients aged less than 18 years to maintain consistency in physiological profiles.

We define the 30-day readmission rate and use it as the outcome of the prediction task. The readmission rate in the MIMIC-III database is 5.32% among adults. We split the dataset into a training set (80% of samples), a validation set (10% of samples), and a testing set (10% of samples). To ensure an unbiased assessment, we use the validation set to tune model parameters and the testing set to evaluate the final model performance.

We create a hypergraph of ICD codes using the MIMIC-III database. For each admission, the associated group of diagnosis ICD codes form a hyperedge of the hypergraph. As the DiPH model encourages diversity (Yu and Zhu, 2023) and it is common that there are similar ICD codes in one admission, we trim all ICD codes into their category level (i.e., we treat 250.00 and 250.01 as the same code 250). Duplicate codes in a hyperedge are removed. ICD codes with less than 5 appearances are also removed.

After the processing, there are 823 ICD codes (nodes) and 36,656 admissions (hyperedges) in the hypergraph.

4.5.3 Experiment Design

We then fit the DiPH model on the hypergraph using both the mixed likelihood and the ordinary log-likelihood. The learned latent positions of the ICD codes, derived from the fitting process, serve as features in the downstream predictive task.

Specifically, for each admission, we aggregate the latent positions of the associated ICD codes. The aggregated vectors, along with other variables, are then used as predictors for Random Forest. In this real data example, we aggregate the latent positions simply by taking the average over all codes.

Hyper-parameters	Values
Dimension d	5, 12, 20, 32, 48
Learning Rate	0.01, 0.001, ..., 10^{-6}
Weight w (Mixed likelihood only)	1, 10, 25, 50, 100

Table 4.1: The selection of hyper-parameters

Table 4.1 shows the selection of hyper-parameters of the estimation methods of DiPH. The best combination of hyper-parameters is selected by cross-validation for both methods. We present the average performance on the test set in the next section. We also show the average performance of the latent space zero-inflated Poisson model (labeled "ZIP") in Chapter 2 for comparison. The experiment setting for the latent space zero-inflated Poisson model is the same as in Section 2.5.

4.5.4 Results

Table 4.2 shows the average test AUC and standard error of each method. Without any embedding method, the Random Forest achieves an AUC score of 0.651, which serves as a baseline for evaluating the amount of information learned from ICD codes by different methods. The standard errors associated with these AUC scores are relatively small for all approaches, suggesting a high level of precision in the AUC estimates and model robustness.

Employing the latent space zero-inflated Poisson model for embedding leads to a higher AUC score of 0.713, indicating that the additional information provided by latent positions learned by the model benefits the prediction model. The ordinary DiPH model also improves upon the baseline with an AUC score of 0.708, though it falls slightly short of the performance achieved by the latent space zero-inflated Poisson model.

Notably, the DiPH model estimated by the mixed likelihood achieves an AUC score of 0.719, surpassing all other methods. This improvement suggests that the mixed likelihood estimation method, which combines both ordinary log-likelihood and pseudo-log-likelihood, is particularly effective in capturing useful information from the ICD codes. The results

demonstrated again the ability of the mixed likelihood estimation method to improve the performance of the DiPH model.

Embedding Method	None	ZIP	DiPH	DiPH (Mixed likelihood)
Test AUC (standard error)	0.651 (0.004)	0.713 (5.33e-3)	0.708 (4.91e-3)	0.719 (4.97e-3)

Table 4.2: The test AUC scores and standard error of each model

4.6 Discussion and Future Work

In this chapter, we introduced the mixed likelihood function for the DiPH model and proposed a new estimation method using it.

The consistency theorem provided for the mixed likelihood estimation confirms that the method maintains the great theoretical properties of the DiPH model. It is guaranteed that as the amount of hyperedges increases, the estimations converge to the true underlying parameters, providing a solid theoretical foundation for the application of this method.

In the simulation studies, the proposed approach demonstrated a significant advantage over the original approach, especially when dealing with high-dimensional and sparse hypergraphs. The ability to produce reliable, robust, and accurate estimations even with limited information is especially valuable in fields like healthcare research, where the databases are often relatively small.

In conclusion, the mixed likelihood function presents a better estimation performance for the DiPH model. Its advantages are demonstrated in both simulation studies and real data examples.

As far as we know, this is the first work to estimate a model with the combination of ordinary likelihood and pseudo-likelihood. By introducing pseudo-log-likelihood, which essentially considers sub-models, the method provides a panoramic view of the data from different perspectives. This approach significantly enhances our ability to capture and inter-

pret complex patterns and relationships that can hardly captured by traditional methods, thereby improving the accuracy and robustness of our model performance.

Given the observed success of the mixed likelihood methodology with the DiPH model, it's plausible to conjecture that this approach could be beneficially applied to a range of other models, particularly those involving multi-way interactions. Therefore, one potential extension of the study is to consider a more general framework for an estimation method using mixed likelihood.

In this chapter, we demonstrate that our proposed approach is more stable and achieves better performance. We also establish consistency for the proposed model. However, it is still unclear how the mixed likelihood helps with the estimation. Therefore, another direction of future work is developing a deeper understanding of the mechanism through which the mixed likelihood function enhances the estimation process.

APPENDIX A

Appendix For Chapter 2

A.1 Proofs of main theorems

A.1.1 Proof of Theorem 2.3.2

By the definition of (Θ^*, Φ^*) and $(\hat{\Theta}, \hat{\Phi})$, we have

$$l(\Theta^*, \Phi^*) - l(\hat{\Theta}, \hat{\Phi}) \geq 0 \tag{A.1}$$

where,

$$\begin{aligned} l(\Theta, \Phi) =& l(\alpha, V, \beta, W), \\ \Theta =& VV^T + \alpha 1_n^T + 1_n^T \alpha, \\ \Phi =& WW^T + \beta 1_n^T + 1_n^T \beta. \end{aligned}$$

Let $f(\Theta) = \log(1 + \exp(\theta))$, $g(\Phi) = \exp(\Phi)$, and $h(\Theta, \Phi) = \log(1 + e^{\Theta - e^{\Phi}})$ then

$$
\begin{aligned}
\text{LHS of (A.1)} = & \sum_{X_{ij}=0} \left[\log(1 + e^{\hat{\Theta}_{ij} - e^{\hat{\Phi}_{ij}}}) - \log(1 + e^{\Theta^*_{ij} - e^{\Phi^*_{ij}}}) \right] \\
& + \sum_{X_{ij}>0} \left[(\hat{\Theta}_{ij} - \Theta^*_{ij}) + X_{ij}(\hat{\Phi}_{ij} - \Phi^*_{ij}) - (\exp(\hat{\Phi}_{ij}) - \exp(\Phi^*_{ij})) \right] \\
& - \sum_{i,j=1}^{n} \left[\log(1 + \exp(\hat{\theta_{ij}})) - \log(1 + \exp(\theta^*_{ij})) \right] \\
& = \sum_{X_{ij}=0} \left[h(\hat{\Theta}_{ij}, \hat{\Phi}_{ij}) - h(\Theta^*_{ij}, \Phi^*_{ij}) \right] \\
& + \sum_{X_{ij}>0} \left[(\hat{\Theta}_{ij} - \Theta^*_{ij}) + X_{ij}(\hat{\Phi}_{ij} - \Phi^*_{ij}) - (g(\hat{\Phi}_{ij}) - g(\Phi^*_{ij})) \right] \\
& - \sum_{i,j=1}^{n} \left[f(\hat{\Theta}_{ij}) - f(\Theta^*_{ij}) \right] \\
& \geq 0
\end{aligned}
$$

To use the Taylor Theorem, we rewrite the above formula as:

$$
\begin{aligned}
0 \leq & \sum_{X_{ij}=0} \left[h(\hat{\Theta}_{ij}, \hat{\Phi}_{ij}) - h(\Theta^*_{ij}, \Phi^*_{ij}) - h_{\Theta}(\Theta^*_{ij}, \Phi^*_{ij})(\hat{\Theta}_{ij} - \Theta^*_{ij}) - h_{\Phi}(\Theta^*_{ij}, \Phi^*_{ij})(\hat{\Phi}_{ij} - \Phi^*_{ij}) \right] \\
& - \sum_{i,j=1}^{n} \left[f(\hat{\Theta}_{ij}) - f(\Theta^*_{ij}) - f'(\Theta^*_{ij})(\hat{\Theta}_{ij} - \Theta^*_{ij}) \right] \\
& - \sum_{X_{ij}>0} \left[g(\hat{\Phi}_{ij}) - g(\Phi^*_{ij}) - g'(\Phi^*_{ij})(\hat{\Phi}_{ij} - \Phi^*_{ij}) \right] \\
& + \left[\sum_{X_{ij}>0} 1 + \sum_{X_{ij}=0} h_{\Theta}(\Theta^*_{ij}, \Phi^*_{ij}) - \sum_{i,j=1}^{n} f'(\Theta^*_{ij}) \right] (\hat{\Theta}_{ij} - \Theta^*_{ij}) \\
& + \left[\sum_{X_{ij}>0} (X_{ij} - g'(\Phi^*_{ij})) + \sum_{X_{ij}=0} h_{\Phi}(\Theta^*_{ij}, \Phi^*_{ij}) \right] (\hat{\Phi}_{ij} - \Phi^*_{ij})
\end{aligned}
$$

To simplify the formula, let

$$\Delta^{(1)} = \sum_{X_{ij}>0} 1 + \sum_{X_{ij}=0} h_\Theta(\Theta^*_{ij}, \Phi^*_{ij}) - \sum_{i,j=1}^{n} f'(\Theta^*_{ij}),$$

and

$$\Delta^{(2)} = \sum_{X_{ij}>0} (X_{ij} - g'(\Phi^*_{ij})) + \sum_{X_{ij}=0} h_\Phi(\Theta^*_{ij}, \Phi^*_{ij}).$$

By using the Taylor Theorem with Peano remainder, we have:

$$\begin{aligned}
0 \le \frac{1}{2} \sum_{X_{ij}=0} [&h_{\Theta\Theta}(\Theta^*_{ij}, \Phi^*_{ij})(\hat{\Theta}_{ij} - \Theta^*_{ij})^2 + h_{\Theta\Phi}(\Theta^*_{ij}, \Phi^*_{ij})(\hat{\Theta}_{ij} - \Theta^*_{ij})(\hat{\Phi}_{ij} - \Phi^*_{ij}) + \\
&h_{\Phi\Phi}(\Theta^*_{ij}, \Phi^*_{ij})(\hat{\Phi}_{ij} - \Phi^*_{ij})^2 + 2 \sum_{|\alpha|=2} R^h_\alpha(\hat{\Theta}_{ij}, \hat{\Phi}_{ij})(\hat{\Theta}_{ij} - \Theta^*_{ij}, \hat{\Phi}_{ij} - \Phi^*_{ij})^\alpha] \\
-\frac{1}{2} \sum_{i,j=1}^{n} &[f''(\Theta^*_{ij})(\hat{\Theta}_{ij} - \Theta^*_{ij})^2 + 2R^f(\hat{\Theta}_{ij})(\hat{\Theta}_{ij} - \Theta^*_{ij})^2] \qquad \text{(A.2)} \\
-\frac{1}{2} \sum_{X_{ij}>0} &[g''(\Phi^*_{ij})(\hat{\Phi}_{ij} - \Phi^*_{ij})^2 + 2R^g(\hat{\Phi}_{ij})(\hat{\Phi}_{ij} - \Phi^*_{ij})^2] \\
&+ \left\langle \Delta^{(1)}, \hat{\Theta} - \Theta^* \right\rangle + \left\langle \Delta^{(2)}, \hat{\Phi} - \Phi^* \right\rangle,
\end{aligned}$$

where R^h_α, R^f, R^g are Peano remainders s.t.

$$\lim_{\hat{\Theta}_{ij} \to \Theta^*_{ij}} R^f(\hat{\Theta}_{ij}) \to 0, \lim_{\hat{\Phi}_{ij} \to \Phi^*_{ij}} R^g(\hat{\Phi}_{ij}) \to 0, \text{and} \lim_{(\hat{\Theta}_{ij}, \hat{\Phi}_{ij}) \to (\Theta^*_{ij}, \Phi^*_{ij})} R^h_\alpha(\hat{\Theta}_{ij}, \hat{\Phi}_{ij}) \to 0, \qquad \text{(A.3)}$$

for any multi-index notation $|\alpha| = 2$.

We present some lemmas first before we proceed to the proof of the main theorem.

Lemma A.1.1 *For any $1 \le i, j \le n$, $\mathbf{E}\Delta^{(1)}_{i,j} = 0$, and there exists constant c_1 such that $\mathbf{E}\|\Delta^{(1)}\|_{op} \le c_1\sqrt{n}$.*

Lemma A.1.2 *Similar to Lemma A.1.1, we have for any $1 \le i, j \le n$, $\mathbf{E}\Delta^{(2)}_{i,j} = 0$, and there exists constant c_2 such that $\mathbf{E}\|\Delta^{(2)}\|_{op} \le c_2\sqrt{n}$.*

Lemma A.1.3 *(Theorem 2 in Latała (2005)). For any finite matrix $Z = \{Z_{ij}\} \in \mathbb{R}^{n\times m}$ of independent mean zero random variables, there exists a universal constant κ_0, which does not depend on n and m, such that*

$$\mathbf{E}\|Z\|_{op} \leq \kappa_0 \left[\max_i \left(\sum_j \mathbf{E}(Z_{ij}^2) \right)^{1/2} + \left(\max_j \sum_i \mathbf{E}(Z_{ij}^2) \right)^{1/2} + \left(\sum_{ij} \mathbf{E}(Z_{ij}^2) \right)^{1/4} \right].$$

Lemma A.1.4 *There exist constants $C_\Theta, C_\Phi > 0$, such that,*

$$\begin{aligned}
&\mathbf{E}\Big(\sum_{i,j=1}^n f''(\Theta^*_{ij})(\Theta_{ij}-\Theta^*_{ij})^2 + \sum_{X_{ij}>0} g''(\Phi^*_{ij})(\Phi_{ij}-\Phi^*_{ij})^2 \\
&- \sum_{X_{ij}=0} [h_{\Theta\Theta}(\Theta^*_{ij},\Phi^*_{ij})(\Theta_{ij}-\Theta^*_{ij})^2 + h_{\Theta\Phi}(\Theta^*_{ij},\Phi^*_{ij})(\Theta_{ij}-\Theta^*_{ij})(\Phi_{ij}-\Phi^*_{ij}) + h_{\Phi\Phi}(\Theta^*_{ij},\Phi^*_{ij})(\Phi_{ij}-\Phi^*_{ij})^2]\Big) \\
\geq & C_\Theta \sum_{i,j=1}^n (\Theta_{ij}-\Theta^*_{ij})^2 + C_\Phi \sum_{i,j=1}^n (\Phi_{ij}-\Phi^*_{ij})^2.
\end{aligned}$$

Lemma A.1.5 *There exists constant C, such that, for any $|\Theta_{ij}-\Theta^*_{ij}| < C$ and $|\Phi_{ij}-\Phi^*_{ij}| < C$, we have $R^f(\Theta_{ij}) < \frac{C_\Theta}{8}$, $R^g(\Phi_{ij}) < \frac{C_\Phi}{8}$, $R^h_{\Theta\Theta}(\Theta_{ij},\Phi_{ij}) < \frac{C_\Theta}{8}$, $R^h_{\Phi\Phi}(\Theta_{ij},\Phi_{ij}) < \frac{C_\Phi}{8}$, and $R^h_{\Theta\Phi}(\Theta_{ij},\Phi_{ij}) < \frac{\min(C_\Phi,C_\Theta)}{8}$.*

Combining Lemma A.1.4, Lemma A.1.5, and (A.2), we have,

$$
\begin{aligned}
&\frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}-\Theta^*\|_F^2+\frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}-\Phi^*\|_F^2\\
\leq&\frac{1}{2}\mathbf{E}\sum_{i,j=1}^{n}[f''(\Theta_{ij}^*)(\hat{\Theta}_{ij}-\Theta_{ij}^*)^2+2R^f(\hat{\Theta}_{ij})(\hat{\Theta}_{ij}-\Theta_{ij}^*)^2]\\
&+\frac{1}{2}\mathbf{E}\sum_{X_{ij}>0}[g''(\Phi_{ij}^*)(\hat{\Phi}_{ij}-\Phi_{ij}^*)^2+2R^g(\hat{\Phi}_{ij})(\hat{\Phi}_{ij}-\Phi_{ij}^*)^2]\\
&-\frac{1}{2}\mathbf{E}\sum_{X_{ij}=0}[h_{\Theta\Theta}(\Theta_{ij}^*,\Phi_{ij}^*)(\hat{\Theta}_{ij}-\Theta_{ij}^*)^2+h_{\Theta\Phi}(\Theta_{ij}^*,\Phi_{ij}^*)(\hat{\Theta}_{ij}-\Theta_{ij}^*)(\hat{\Phi}_{ij}-\Phi_{ij}^*)\\
&+h_{\Phi\Phi}(\Theta_{ij}^*,\Phi_{ij}^*)(\hat{\Phi}_{ij}-\Phi_{ij}^*)^2+2\sum_{|\alpha|=2}R_\alpha^h(\hat{\Theta}_{ij},\hat{\Phi}_{ij})(\hat{\Theta}_{ij}-\Theta_{ij}^*,\hat{\Phi}_{ij}-\Phi_{ij}^*)^\alpha]\\
\leq&\mathbf{E}\left\langle\Delta^{(1)},\hat{\Theta}-\Theta^*\right\rangle+\mathbf{E}\left\langle\Delta^{(2)},\hat{\Phi}-\Phi^*\right\rangle\\
\leq&\mathbf{E}(\|\Delta^{(1)}\|_{op}\sqrt{\mathrm{rank}(\hat{\Theta}-\Theta^*)}\|\hat{\Theta}-\Theta^*\|_F)+\mathbf{E}(\|\Delta^{(2)}\|_{op}\sqrt{\mathrm{rank}(\hat{\Phi}-\Phi^*)}\|\hat{\Phi}-\Phi^*\|_F)
\end{aligned}
$$

By Lemma A.1.1, and Lemma A.1.2, we have,

$$
\frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}-\Theta^*\|_F^2+\frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}-\Phi^*\|_F^2\leq c_1\sqrt{nq_1}\mathbf{E}\|\hat{\Theta}-\Theta^*\|_F+c_2\sqrt{nq_2}\mathbf{E}\|\hat{\Phi}-\Phi^*\|_F.
$$

Therefore,

$$
\begin{aligned}
&\max\{\mathbf{E}\|\hat{\Theta}-\Theta^*\|_F^2,\mathbf{E}\|\hat{\Phi}-\Phi^*\|_F^2\}\\
\leq&\frac{8\max\{c_1\sqrt{nq_1},c_2\sqrt{nq_2}\}}{\min\{C_\Theta,C_\Phi\}}(\mathbf{E}\|\hat{\Theta}-\Theta^*\|_F+\mathbf{E}\|\hat{\Theta}-\Theta^*\|_F)\\
\leq&\frac{16\max\{c_1\sqrt{nq_1},c_2\sqrt{nq_2}\}}{\min\{C_\Theta,C_\Phi\}}\max\{\mathbf{E}\|\hat{\Theta}-\Theta^*\|_F,\mathbf{E}\|\hat{\Phi}-\Phi^*\|_F\}.
\end{aligned}
$$

So, we have,

$$
\mathbf{E}\|\hat{\Theta}-\Theta^*\|_F\leq\frac{16\max\{c_1\sqrt{nq_1},c_2\sqrt{nq_2}\}}{\min\{C_\Theta,C_\Phi\}},
$$

and

$$
\mathbf{E}\|\hat{\Phi}-\Phi^*\|_F\leq\frac{16\max\{c_1\sqrt{nq_1},c_2\sqrt{nq_2}\}}{\min\{C_\Theta,C_\Phi\}},
$$

which concludes our proof.

A.1.1.1 Proof of Lemma A.1.1

To simplify the proof, we omit all the star marks in Φ^*, Φ^*_{ij}, Θ^*, and Θij^* in this proof. Because

$$\Delta^{(1)} = \sum_{X_{ij}>0} 1 + \sum_{X_{ij}=0} h_\Theta(\Theta_{ij}, \Phi_{ij}) - \sum_{i,j=1}^{n} f'(\Theta_{ij}),$$

we have,

$$\begin{aligned}
\mathbf{E}\Delta^{(1)}_{i,j} &= \mathbf{P}(X_{ij} > 0) + \mathbf{P}(X_{ij} = 0) \times h_\Theta(\Theta_{ij}, \Phi_{ij}) - f'(\Theta_{ij}) \\
&= (1 - \mathbf{P}(X_{ij} = 0)) + \mathbf{P}(X_{ij} = 0) \times \frac{e^{\Theta_{ij} - \exp(\Phi_{ij})}}{1 + e^{\Theta_{ij} - \exp(\Phi_{ij})}} - \frac{e^{\Theta_{ij}}}{1 + e^{\Theta_{ij}}}. \qquad \text{(A.4)}
\end{aligned}$$

Plug

$$\begin{aligned}
\mathbf{P}(X_{ij} = 0) &= \frac{1}{1 + e^{\Theta_{ij}}} + \frac{e^{\Theta_{ij}}}{1 + e^{\Theta_{ij}}} \cdot e^{-\exp(\Phi_{ij})} \\
&= \frac{1 + e^{\Theta_{ij} - \exp(\Phi_{ij})}}{1 + e^{\Theta_{ij}}}
\end{aligned}$$

in (A.4), we have

$$\begin{aligned}
\mathbf{E}\Delta^{(1)}_{i,j} &= \frac{e^{\Theta_{ij}} - e^{\Theta_{ij} - \exp(\Phi_{ij})}}{1 + e^{\Theta_{ij}}} + \frac{1 + e^{\Theta_{ij} - \exp(\Phi_{ij})}}{1 + e^{\Theta_{ij}}} \times \frac{e^{\Theta_{ij} - \exp(\Phi_{ij})}}{1 + e^{\Theta_{ij} - \exp(\Phi_{ij})}} - \frac{e^{\Theta_{ij}}}{1 + e^{\Theta_{ij}}}. \\
&= 0
\end{aligned}$$

Because Θ_{ij} and Φ_{ij} are bounded, and

$$\Delta^{(1)}_{i,j} = \mathbb{1}_{X_{ij}>0} + \mathbb{1}_{X_{ij}=0} \times \frac{e^{\Theta_{ij} - \exp(\Phi_{ij})}}{1 + e^{\Theta_{ij} - \exp(\Phi_{ij})}} - \frac{e^{\Theta_{ij}}}{1 + e^{\Theta_{ij}}},$$

one can show that $\mathbf{E}((\Delta^{(1)}_{i,j})^2)$ is also bounded. Therefore, by Lemma A.1.3, we can prove the second part of Leamm A.1.1, i.e., there exists constant c_1 such that $\mathbf{E}\|\Delta^{(1)}\|_{op} \leq c_1\sqrt{n}$.

This concludes our proof of Lemma A.1.1.

A.1.1.2 Proof of Lemma A.1.2

To simplify the proof, we also omit all the star marks in Φ^*, Φ^*_{ij}, Θ^*, and Θij^* in this proof. Because

$$\Delta^{(2)} = \sum_{X_{ij}>0} (X_{ij} - g'(\Phi_{ij})) + \sum_{X_{ij}=0} h_\Phi(\Theta_{ij}, \Phi_{ij})$$

we have

$$\begin{aligned}
\mathbf{E}\Delta^{(2)}_{ij} &= \mathbf{E}(\mathbb{1}_{X_{ij}>0} \cdot X_{ij}) - \mathbf{E}(\mathbb{1}_{X_{ij}>0} \cdot g'(\Phi_{ij})) + \mathbf{E}(\mathbb{1}_{X_{ij}=0} \cdot h_\Phi(\Theta_{ij}, \Phi_{ij})) \\
&= \mathbf{E}(X_{ij}) - \mathbf{P}(X_{ij} > 0) g'(\Phi_{ij}) + \mathbf{P}(X_{ij} = 0) h_\Phi(\Theta_{ij}, \Phi_{ij})) \\
&= \frac{e^{\Theta_{ij}}}{1+e^{\Theta_{ij}}} \cdot e^{\Phi_{ij}} - \frac{e^{\Theta_{ij}} - e^{\Theta_{ij} - \exp(\Phi_{ij})}}{1+e^{\Theta_{ij}}} \cdot e^{\Phi_{ij}} + \frac{1+e^{\Theta_{ij}-\exp(\Phi_{ij})}}{1+e^{\Theta_{ij}}} \cdot \frac{-e^{\Theta_{ij}-\exp(\Phi_{ij})+\Phi_{ij}}}{1+e^{\Theta_{ij}-\exp(\Phi_{ij})}} \\
&= 0
\end{aligned}$$

Because Θ_{ij} and Φ_{ij} are bounded, one can show that $\mathbf{E}((\Delta^{(2)}_{i,j})^2)$ is also bounded. Therefore, by Lemma A.1.3, we can prove the second part of Leamm A.1.2, i.e., there exists constant c_2 such that $\mathbf{E}\|\Delta^{(2)}\|_{op} \leq c_2\sqrt{n}$. This concludes our proof of Lemma A.1.2.

A.1.1.3 Proof of Lemma A.1.4

We first provide the formulas for all the derivatives in the Lemma:

$$
\begin{aligned}
f''(\Theta^*_{ij}) &= \frac{e^{\Theta^*_{ij}}}{(1+e^{\Theta^*_{ij}})^2} \\
g''(\Phi^*_{ij}) &= \exp(\Phi^*_{ij}) \\
h_{\Theta\Theta}(\Theta^*_{ij}, \Phi^*_{ij}) &= \frac{e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})}}{(1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})})^2} \\
h_{\Phi\Phi}(\Theta^*_{ij}, \Phi^*_{ij}) &= \frac{e^{\exp(\Phi^*_{ij})+\Theta^*_{ij}+2\Phi^*_{ij}} - e^{\exp(\Phi^*_{ij})+\Theta^*_{ij}+\Phi^*_{ij}} - e^{2\Theta^*_{ij}+\Phi^*_{ij}}}{(e^{\exp(\Phi^*_{ij})}+e^{\Theta^*_{ij}})^2} \\
h_{\Theta\Phi}(\Theta^*_{ij}, \Phi^*_{ij}) &= -\frac{e^{-\exp(\Phi^*_{ij})+\Theta^*_{ij}+\Phi^*_{ij}}}{(1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})})^2}.
\end{aligned}
$$

We have

$$
\begin{aligned}
&\mathbf{E}(\sum_{i,j=1}^{n} f''(\Theta^*_{ij}) - \sum_{X_{ij}=0} h_{\Theta\Theta}(\Theta^*_{ij}, \Phi^*_{ij}))_{ij} \\
=&\frac{e^{\Theta^*_{ij}}}{(1+e^{\Theta^*_{ij}})^2} - \frac{1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})}}{1+e^{\Theta^*_{ij}}} \cdot \frac{e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})}}{(1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})})^2} \\
=&\frac{e^{\Theta^*_{ij}} - e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})}}{(1+e^{\Theta^*_{ij}})^2(1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})})}.
\end{aligned} \tag{A.5}
$$

Let

$$
A = \frac{e^{\Theta^*_{ij}} - e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})}}{(1+e^{\Theta^*_{ij}})^2(1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})})}. \tag{A.6}
$$

Similarly, we have

$$\begin{aligned}
&\mathbf{E}(\sum_{X_{ij}>0} g''(\Phi^*_{ij}) - \sum_{X_{ij}=0} h_{\Phi\Phi}(\Theta^*_{ij}, \Phi^*_{ij})) \\
=&\frac{e^{\Theta^*_{ij}} - e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})}}{1+e^{\Theta^*_{ij}}} \cdot \exp(\Phi^*_{ij}) \\
&- \frac{1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})}}{1+e^{\Theta^*_{ij}}} \cdot \frac{e^{\exp(\Phi^*_{ij})+\Theta^*_{ij}+2\Phi^*_{ij}} - e^{\exp(\Phi^*_{ij})+\Theta^*_{ij}+\Phi^*_{ij}} - e^{2\Theta^*_{ij}+\Phi^*_{ij}}}{(e^{\exp(\Phi^*_{ij})}+e^{\Theta^*_{ij}})^2} \\
=&\frac{e^{\Theta^*_{ij}+\Phi^*_{ij}} + e^{2\Theta^*_{ij}+\Phi^*_{ij}-\exp(\Phi^*_{ij})} - e^{\Theta^*_{ij}+2\Phi^*_{ij}-\exp(\Phi^*_{ij})}}{(1+e^{\Theta^*_{ij}})(1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})})}.
\end{aligned} \tag{A.7}$$

Let

$$B = \frac{e^{\Theta^*_{ij}+\Phi^*_{ij}} + e^{2\Theta^*_{ij}+\Phi^*_{ij}-\exp(\Phi^*_{ij})} - e^{\Theta^*_{ij}+2\Phi^*_{ij}-\exp(\Phi^*_{ij})}}{(1+e^{\Theta^*_{ij}})(1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})})}. \tag{A.8}$$

Also, let

$$\begin{aligned}
C =& -\mathbf{E}\sum_{X_{ij}=0} h_{\Theta\Phi}(\Theta^*_{ij}, \Phi^*_{ij}) \\
=&\frac{e^{-\exp(\Phi^*_{ij})+\Theta^*_{ij}+\Phi^*_{ij}}}{(1+e^{\Theta^*_{ij}})(1+e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})})}.
\end{aligned} \tag{A.9}$$

Having (A.5)-(A.9), we only need to prove that there exist constants $C_\Theta, C_\Phi > 0$, such that,

$$\begin{aligned}
&A(\Theta_{ij}-\Theta^*_{ij})^2 + B(\Phi_{ij}-\Phi^*_{ij})^2 + C(\Theta_{ij}-\Theta^*_{ij})(\Phi_{ij}-\Phi^*_{ij}) \\
&\qquad \geq C_\Theta(\Theta_{ij}-\Theta^*_{ij})^2 + C_\Phi(\Phi_{ij}-\Phi^*_{ij})^2
\end{aligned} \tag{A.10}$$

We claim that $A > \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}}$ and $B > \frac{1}{2}(1+e^{\Theta^*_{ij}})C$. Note that

$$|C(\Theta_{ij}-\Theta^*_{ij})(\Phi_{ij}-\Phi^*_{ij})| \leq \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}}(\Theta_{ij}-\Theta^*_{ij})^2 + \frac{1}{2}(1+e^{\Theta^*_{ij}})C(\Phi_{ij}-\Phi^*_{ij})^2.$$

The left-hand-side of (A.10)

$$\begin{aligned}\text{LHS} \geq & A(\Theta_{ij}-\Theta^*_{ij})^2 + B(\Phi_{ij}-\Phi^*_{ij})^2 - \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}}(\Theta_{ij}-\Theta^*_{ij})^2 - \frac{1}{2}(1+e^{\Theta^*_{ij}})C(\Phi_{ij}-\Phi^*_{ij})^2 \\ \geq & \min_{i,j}(A - \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}})(\Theta_{ij}-\Theta^*_{ij})^2 + \min_{i,j}(B - \frac{1}{2}(1+e^{\Theta^*_{ij}})C)(\Phi_{ij}-\Phi^*_{ij})^2 \qquad \text{(A.11)}\end{aligned}$$

Let $C_\Theta = \inf_{i,j,\Theta^*,\Phi^*}(A - \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}})$ and $C_\Phi = \inf_{i,j,\Theta^*,\Phi^*}(B - \frac{1}{2}(1+e^{\Theta^*_{ij}})C)$, we have (A.10) holds. Therefore, to prove the Lemma, we only need to show that $A > \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}}$, $B > \frac{1}{2}(1+e^{\Theta^*_{ij}})C$ and $C_\Theta, C_\Phi > 0$.

We first show that $A > \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}}$ and $C_\Theta > 0$. Note

$$\begin{aligned} & A > \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}} \\ \iff & e^{\Theta^*_{ij}} - e^{\Theta^*_{ij}-\exp(\Phi^*_{ij})} - \frac{1}{2}e^{-\exp(\Phi^*_{ij})+\Theta^*_{ij}+\Phi^*_{ij}} > 0 \\ \iff & 1 > e^{-\exp(\Phi^*_{ij})} + \frac{1}{2}e^{-\exp(\Phi^*_{ij})+\Phi^*_{ij}}. \qquad \text{(A.12)}\end{aligned}$$

Let $x = e^{\Phi^*_{ij}} > 0$, then the right-hand-side of (A.12) becomes $y(x) = e^{-x} + \frac{1}{2}xe^{-x}$. Note that

$$\begin{aligned} y'(x) &= -e^{-x} + \frac{1}{2}e^{-x} - \frac{1}{2}xe^{-x} \\ &= -\frac{1}{2}e^{-x} - \frac{1}{2}xe^{-x} \\ &< 0, \end{aligned}$$

so the right-hand-side of (A.12) is smaller than $y(0) = 1$. Therefore, $A > \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}}$. Also, because Θ^*_{ij} is bounded, x cannot approach zero infinitely and $A - \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}}$ cannot approach zero infinitely. As a result $C_\Theta = \inf_{i,j,\Theta^*,\Phi^*}(A - \frac{1}{2}\frac{C}{1+e^{\Theta^*_{ij}}}) > 0$.

We then show $B > \frac{1}{2}(1+e^{\Theta^*_{ij}})C$ and $C_\Phi > 0$. Note

$$
\begin{aligned}
& B > \frac{1}{2}(1+e^{\Theta^*_{ij}})C \\
\iff & e^{\Theta^*_{ij}+\Phi^*_{ij}} + e^{2\Theta^*_{ij}+\Phi^*_{ij}-\exp(\Phi^*_{ij})} - e^{\Theta^*_{ij}+2\Phi^*_{ij}-\exp(\Phi^*_{ij})} > \frac{1}{2}(1+e^{\Theta^*_{ij}})(e^{-\exp(\Phi^*_{ij})+\Theta^*_{ij}+\Phi^*_{ij}}) \\
\iff & e^{\Theta^*_{ij}+\Phi^*_{ij}} + e^{2\Theta^*_{ij}+\Phi^*_{ij}-\exp(\Phi^*_{ij})} - e^{\Theta^*_{ij}+2\Phi^*_{ij}-\exp(\Phi^*_{ij})} > \frac{1}{2}e^{-\exp(\Phi^*_{ij})+\Theta^*_{ij}+\Phi^*_{ij}} + \frac{1}{2}e^{-\exp(\Phi^*_{ij})+2\Theta^*_{ij}+\Phi^*_{ij}} \\
\impliedby & e^{\Theta^*_{ij}+\Phi^*_{ij}} - e^{\Theta^*_{ij}+2\Phi^*_{ij}-\exp(\Phi^*_{ij})} > \frac{1}{2}e^{-\exp(\Phi^*_{ij})+\Theta^*_{ij}+\Phi^*_{ij}} \\
\iff & 1 - e^{\Phi^*_{ij}-\exp(\Phi^*_{ij})} > \frac{1}{2}e^{-\exp(\Phi^*_{ij})}
\end{aligned}
\tag{A.13}
$$

Let $x = e^{\Phi^*_{ij}} > 0$ and $y(x) = xe^{-x} + \frac{1}{2}e^{-x}$, then

$$
\begin{aligned}
y'(x) &= e^{-x} - xe^{-x} - \frac{1}{2}e^{-x} \\
&= \frac{1}{2}e^{-x} - xe^{-x}.
\end{aligned}
$$

Note that $y'(x) \geq 0$ when $x \leq \frac{1}{2}$ and $y'(x) \leq 0$ when $x \geq \frac{1}{2}$, so $y(x) \leq y(\frac{1}{2}) = e^{-\frac{1}{2}} < 1$. Therefore, the last inequation of (A.13) holds. As a result $B > \frac{1}{2}(1+e^{\Theta^*_{ij}})C$, and $C_\Phi = \inf_{i,j,\Theta^*,\Phi^*}(B - \frac{1}{2}(1+e^{\Theta^*_{ij}})C) > 0$. We have proved Lemma A.1.4.

A.1.1.4 Proof of Lemma A.1.5

Because $C_\Theta, C_\Phi > 0$ and

$$
\lim_{\hat{\Theta}_{ij}\to\Theta^*_{ij}} R^f(\hat{\Theta}_{ij}) \to 0, \lim_{\hat{\Phi}_{ij}\to\Phi^*_{ij}} R^g(\hat{\Phi}_{ij}) \to 0, \text{and} \lim_{(\hat{\Theta}_{ij},\hat{\Phi}_{ij})\to(\Theta^*_{ij},\Phi^*_{ij})} R^h_\alpha(\hat{\Theta}_{ij},\hat{\Phi}_{ij}) \to 0,
$$

for any multi-index notation $|\alpha| = 2$, we can find such $C > 0$ satisfying $R^f(\Theta_{ij}) < \frac{C_\Theta}{8}$, $R^g(\Phi_{ij}) < \frac{C_\Phi}{8}$, $R^h_{\Theta\Theta}(\Theta_{ij},\Phi_{ij}) < \frac{C_\Theta}{8}$, $R^h_{\Phi\Phi}(\Theta_{ij},\Phi_{ij}) < \frac{C_\Phi}{8}$, and $R^h_{\Theta\Phi}(\Theta_{ij},\Phi_{ij}) < \frac{\min(C_\Phi,C_\Theta)}{8}$ for any $|\Theta_{ij} - \Theta^*_{ij}| < C$ and $|\Phi_{ij} - \Phi^*_{ij}| < C$.

A.1.2 Proof of Theorem 2.3.1

The exact same strategy can be used to prove Theorem 2.3.2. And it is easier to prove Theorem 2.3.1 than Theorem 2.3.2, so we omit the proof of Theorem 2.3.1 here.

APPENDIX B

Appendix For Chapter 3

B.1 Proof of main theorems

By the definition of $(\Theta^{(k)^*}, \Phi^{(k)^*}, T_V^*, T_W^*)$ and $(\hat{\Theta}^{(k)}, \hat{\Phi}^{(k)}, \hat{T}_V, \hat{T}_W)$, we have

$$l(\Theta^{(9)^*}, \Phi^{(9)^*}, \Theta^{(10)^*}, \Phi^{(10)^*}, T_V^*, T_W^*) - l(\hat{\Theta}^{(9)}, \hat{\Phi}^{(9)}, \hat{\Theta}^{(10)}, \hat{\Phi}^{(10)}, \hat{T}_V, \hat{T}_W) \geq 0.$$

Note that

$$l = l^{(9)} + \gamma \cdot l_t + l^{(10)},$$

and

$$l_t(T_V^*, T_W^*) = 0,$$

we have

$$\begin{aligned} & l^{(9)}(\Theta^{(9)^*}, \Phi^{(9)^*}) + l^{(10)}(\Theta^{(10)^*}, \Phi^{(10)^*}) \\ & \quad - l^{(9)}(\hat{\Theta}^{(9)}, \hat{\Phi}^{(9)}) - \gamma \cdot l_t(\hat{\Theta}^{(9)}, \hat{\Phi}^{(9)}, \hat{\Theta}^{(10)}, \hat{\Phi}^{(10)}, \hat{T}_V, \hat{T}_W) - l^{(10)}(\hat{\Theta}^{(10)}, \hat{\Phi}^{(10)}) \geq 0. \end{aligned} \quad \text{(B.1)}$$

Follow the proof of Theorem 2.3.2 in Appendix A, we have

$$
\begin{aligned}
l^{(k)}(\Theta^{(k)^*},\Phi^{(k)^*}) - l^{(k)}(\hat{\Theta}^{(k)},\hat{\Phi}^{(k)}) \leq \\
c_1\sqrt{n_k q_k}\mathbf{E}\|\hat{\Theta}^{(k)} - \Theta^{(k)^*}\|_F + c_2\sqrt{n_k q_k}\mathbf{E}\|\hat{\Phi}^{(k)} - \Phi^{(k)^*}\|_F \\
- \frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}^{(k)} - \Theta^{(k)^*}\|_F^2 - \frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}^{(k)} - \Phi^{(k)^*}\|_F^2,
\end{aligned} \tag{B.2}
$$

for $k \in \{9, 10\}$. Plug (B.2) in (B.1), we have

$$
\begin{aligned}
& c_1\sqrt{n_9 q_9}\mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F + c_2\sqrt{n_9 q_9}\mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F \\
& - \frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F^2 - \frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F^2 \\
& + c_1\sqrt{n_{10} q_{10}}\mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F + c_2\sqrt{n_{10} q_{10}}\mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F \\
& - \frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F^2 - \frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F^2 \\
& - \gamma \cdot l_t(\hat{\Theta}^{(9)}, \hat{\Phi}^{(9)}, \hat{\Theta}^{(10)}, \hat{\Phi}^{(10)}, \hat{T}_V, \hat{T}_W) \\
\geq & l^{(9)}(\Theta^{(9)^*}, \Phi^{(9)^*}) + l^{(10)}(\Theta^{(10)^*}, \Phi^{(10)^*}) \\
& - l^{(9)}(\hat{\Theta}^{(9)}, \hat{\Phi}^{(9)}) - \gamma \cdot l_t(\hat{\Theta}^{(9)}, \hat{\Phi}^{(9)}, \hat{\Theta}^{(10)}, \hat{\Phi}^{(10)}, \hat{T}_V, \hat{T}_W) - l^{(10)}(\hat{\Theta}^{(10)}, \hat{\Phi}^{(10)}) \\
\geq & 0
\end{aligned} \tag{B.3}
$$

Therefore,

$$
\begin{aligned}
& c_1\sqrt{n_9 q_9}\mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F + c_2\sqrt{n_9 q_9}\mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F \\
& \quad + c_1\sqrt{n_{10} q_{10}}\mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F + c_2\sqrt{n_{10} q_{10}}\mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F \\
\geq & \frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F^2 + \frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F^2 \\
& \frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F^2 + \frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F^2 \\
& \quad + \gamma \cdot l_t(\hat{\Theta}^{(9)}, \hat{\Phi}^{(9)}, \hat{\Theta}^{(10)}, \hat{\Phi}^{(10)}, \hat{T}_V, \hat{T}_W)
\end{aligned} \tag{B.4}
$$

Note that

$$l_t(\hat{\Theta}^{(9)}, \hat{\Phi}^{(9)}, \hat{\Theta}^{(10)}, \hat{\Phi}^{(10)}, \hat{T}_V, \hat{T}_W) \geq 0,$$

we have

$$\begin{aligned}
&c_1\sqrt{n_9 q_9}\mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F + c_2\sqrt{n_9 q_9}\mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F \\
&\quad + c_1\sqrt{n_{10} q_{10}}\mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F + c_2\sqrt{n_{10} q_{10}}\mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F \\
\geq &\frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F^2 + \frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F^2 \\
&\frac{1}{8}C_\Theta \mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F^2 + \frac{1}{8}C_\Phi \mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F^2
\end{aligned} \tag{B.5}$$

Therefore,

$$\begin{aligned}
&\max\{\mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F^2, \mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F^2, \mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F^2, \mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F^2\} \\
\leq &\frac{8\max\{c_1\sqrt{n_9 q_9}, c_2\sqrt{n_9 q_9}, c_1\sqrt{n_{10} q_{10}}, c_2\sqrt{n_{10} q_{10}}\}}{\min\{C_\Theta, C_\Phi\}} \\
\times &(\mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F + \mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F + \mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F + \mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F) \\
\leq &\frac{32\max\{c_1\sqrt{n_9 q_9}, c_2\sqrt{n_9 q_9}, c_1\sqrt{n_{10} q_{10}}, c_2\sqrt{n_{10} q_{10}}\}}{\min\{C_\Theta, C_\Phi\}} \\
\times &(\mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F, \mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F, \mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F, \mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F)
\end{aligned}$$

Because there exists constant C_1, such that

$$\frac{32\max\{c_1\sqrt{n_9 q_9}, c_2\sqrt{n_9 q_9}, c_1\sqrt{n_{10} q_{10}}, c_2\sqrt{n_{10} q_{10}}\}}{\min\{C_\Theta, C_\Phi\}} \leq C_1\sqrt{\max(\{n_9 q_9, n_{10} q_{10}\})},$$

we have

$$\mathbf{E}\|\hat{\Theta}^{(9)} - \Theta^{(9)^*}\|_F \leq C_1\sqrt{\max(\{n_9 q_9, n_{10} q_{10}\})},$$

$$\mathbf{E}\|\hat{\Phi}^{(9)} - \Phi^{(9)^*}\|_F \leq C_1\sqrt{\max(\{n_9 q_9, n_{10} q_{10}\})},$$

$$\mathbf{E}\|\hat{\Theta}^{(10)} - \Theta^{(10)^*}\|_F \leq C_1\sqrt{\max(\{n_9 q_9, n_{10} q_{10}\})},$$

$$\mathbf{E}\|\hat{\Phi}^{(10)} - \Phi^{(10)^*}\|_F \leq C_1\sqrt{\max(\{n_9 q_9, n_{10} q_{10}\})},$$

which concludes our proof.

APPENDIX C

Appendix For Chapter 4

C.1 Proofs of main theorems

C.1.1 Proof of Theorem 4.3.1

Before we prove the Theorem 4.3.1, we first introduce Lemma C.1.1. Let Φ be the expected mixed likelihood function, i.e., $\Phi = \mathrm{E}(\ell_{\text{mixed}})$.

Lemma C.1.1 *It is true that*

$$\arg\max_{L \in \mathbb{S}_{n_v}^+} \Phi(L) = \{SL^*S | S \in \mathbb{D}\},$$

and

$$\arg\max_{\substack{V \in \mathbb{R}_{n_v \times d}, \|V_{i\cdot}\|_2 = 1, \\ \alpha_i \geq 0, \beta \geq 0}} \Phi(V, \alpha, \beta) = \{(SV^*O, \beta^*, \alpha^*) | S \in \mathbb{D} \text{ and } O \in \mathbb{O}_d\}.$$

Essentially, Lemma C.1.1 shows that the true parameters (up to an orthogonal rotation and sign flips) maximize the mixed likelihood function.

Proof of Lemma C.1.1. Let $\mathbb{S}_{n_v}^+$ be the set of n_v-by-n_v symmetric matrices with only

positive elements. For any $L \in \mathbb{S}^+_{n_v}$, we have

$$\begin{aligned}\Phi(L) &= \sum_{e\subset[n_v]} P_{L^*}(e)\left(w \times (\text{logdet}(L_{e_k}) - \text{logdet}(L+I)) + \sum_{e'\subset e}(\log P_L(E=e|e'\subset E))\right)\\ &= \sum_{e\subset[n_v]} P_{L^*}(e) \times w \times (\text{logdet}(L_{e_k}) - \text{logdet}(L+I)) + \sum_{e\subset[n_v]} P_{L^*}(e)\sum_{e'\subset e}(\log P_L(E=e|e'\subset E)).\end{aligned}$$

Let

$$\Phi_1(L) = \sum_{e\subset[n_v]} P_{L^*}(e) \times w \times (\text{logdet}(L_{e_k}) - \text{logdet}(L+I)),$$

and

$$\Phi_2(L) = \sum_{e\subset[n_v]} P_{L^*}(e)\sum_{e'\subset e}(\log P_L(E=e|e'\subset E)).$$

We divide $\Phi(L)$ into $\Phi(L) = \Phi_1(L) + \Phi_2(L)$. We have

$$\begin{aligned}\Phi_2(L) &= \sum_{e\subset[n_v]} P_{L^*}(e)\sum_{e'\subset e}(\log P_L(E=e|e'\subset E))\\ &= \sum_{e'\subset[n_v]}\sum_{e'\subset e\subset[n_v]} P_{L^*}(e)\log P_L(E=e|e'\subset E)\\ &= \sum_{e'\subset[n_v]}\sum_{e'\subset e\subset[n_v]} P_{L^*}(e'\subset E)\frac{P_{L^*}(e)}{P_{L^*}(e'\subset E)}\log P_L(E=e|e'\subset E)\\ &= \sum_{e'\subset[n_v]}\sum_{e'\subset e\subset[n_v]} P_{L^*}(e'\subset E)P_{L^*}(E=e|e'\subset E)\log P_L(E=e|e'\subset E)\\ &= \sum_{e'\subset[n_v]} P_{L^*}(e'\subset E)\sum_{e'\subset e\subset[n_v]} P_{L^*}(E=e|e'\subset E)\log P_L(E=e|e'\subset E). \qquad \text{(C.1)}\end{aligned}$$

Note that $P_{L^*}(E=e|e'\subset E)$ gives a DiPH model on the complement set $\mathcal{V}-e'$ with parameter matrix L^*, $P_L(E=e|e'\subset E)$ also gives a DiPH model on the node set with parameter L. Let $\text{DiPH}_{e'}(L^*)$ be the sub-DiPH model on the set $\mathcal{V}-e'$. Replace the L with L^* in (C.1), we have

$$\Phi_2(L^*) = \sum_{e'\subset[n_v]} P_{L^*}(e'\subset E)\sum_{e'\subset e\subset[n_v]} P_{L^*}(E=e|e'\subset E)\log P_{L^*}(E=e|e'\subset E).$$

Note that for any fixed L^*, $P_{L^*}(e' \subset E)$ is a constant. We know that

$$\begin{aligned}\Phi_2(L^*) - \Phi_2(L) = \sum_{e' \subset [n_v]} P_{L^*}(e' \subset E) \sum_{e' \subset e \subset [n_v]} (P_{L^*}(E = e | e' \subset E) \log P_{L^*}(E = e | e' \subset E) \\ - P_{L^*}(E = e | e' \subset E) \log P_L(E = e | e' \subset E)) \\ = \sum_{e' \subset [n_v]} P_{L^*}(e' \subset E) \times \mathrm{KL}(\mathrm{DiPH}_{e'}(L^*), \mathrm{DiPH}_{e'}(L)),\end{aligned} \tag{C.2}$$

where KL stands for the Kullback-Leibler divergence between two probability distributions. Also, note that

$$\begin{aligned}\Phi_1(L^*) - \Phi_1(L) &= w \sum_{e \subset [n_v]} (P_{L^*}(e) \log(P_{L^*}(e)) - P_{L^*}(e) \log(P_L(e))) \\ &= w \times \mathrm{KL}(\mathrm{DiPH}(L^*), \mathrm{DiPH}(L)).\end{aligned} \tag{C.3}$$

Therefore, combining (C.2) and (C.3), we have that

$$\Phi(L^*) - \Phi(L) = \sum_{e' \subset [n_v]} P_{L^*}(e' \subset E) \times \mathrm{KL}(\mathrm{DiPH}_{e'}(L^*), \mathrm{DiPH}_{e'}(L)) + w \times \mathrm{KL}(\mathrm{DiPH}(L^*), \mathrm{DiPH}(L))$$

is a linear combination of multiple KL divergences. By the property of KL divergence, we know that

$$\Phi(L^*) \geq \Phi(L),$$

and

$$\begin{aligned}&\Phi(L^*) = \Phi(L) \\ \iff &\mathrm{KL}(\mathrm{DiPH}_{e'}(L^*), \mathrm{DiPH}_{e'}(L)) = 0 \text{ for all } e' \subset \mathcal{V}, \text{ and } \mathrm{KL}(\mathrm{DiPH}(L^*), \mathrm{DiPH}(L)) = 0 \\ \iff &\mathrm{DiPH}_{e'}(L^*) = \mathrm{DiPH}_{e'}(L), \text{ and } \mathrm{DiPH}(L^*) = \mathrm{DiPH}(L)\end{aligned}$$

Therefore, we have

$$\arg\max_{L\in\mathbb{S}^+_{n_v}} \Phi(L) = \{SL^*S | S \in \mathbb{D}\},$$

and

$$\arg\max_{\substack{V\in\mathbb{R}_{n_v\times d}, \|V_{i\cdot}\|_2=1,\\ \alpha_i\geq 0, \beta\geq 0}} \Phi(V,\alpha,\beta) = \{(SV^*O, \beta^*, \alpha^*) | S \in \mathbb{D} \text{ and } O \in \mathbb{O}_d\},$$

which concludes our proof for Lemma C.1.1.

Having Lemma C.1.1, we can prove Theorem 4.3.1 in a similar way as Theorem 2 in Yu and Zhu (2023).

Let $\bar{\mathcal{L}}$ be the enlarged feasible space of L:

$$\bar{\mathcal{L}} = \{L = (v_i^T v_j)_{i,j=1}^{n_v} + \text{diag}(\alpha) | v_i^* \in \mathbb{R}^d, \alpha_i \geq 0, \text{ and } \|v_i\|_2 > 0 \text{ is constant}\}.$$

We first present the following Theorem C.1.2:

Theorem C.1.2 *Consider a DiPH model $H(v_1^*, v_2^*, ..., v_{n_v}^*, \beta^*, \alpha^*)$, with the set nodes $\mathcal{V} = [n_v]$ and the set of hyperedges $\mathcal{E} = \{e_1, e_2, ..., e_{n_e}\}$ independently generated with the parameters from the DiPH model. Let $\hat{v}_1, \hat{v}_2, ..., \hat{v}_{n_v}, \hat{\alpha}, \hat{\beta}, \hat{L}$ be the estimated parameter using the optimization problem*

$$\arg\max_{\substack{V\in\mathbb{R}_{n_v\times d}, \|V_{i\cdot}\|_2=1,\\ \alpha_i\geq 0, \beta\geq 0}} \ell_{\text{mixed}} \tag{C.4}$$

and

$$\hat{L} := \arg\max_{L\in\bar{\mathcal{L}}} \ \ell_{\text{mixed}}. \tag{C.5}$$

When $n_v > 2d$ and $\{v_1^*, v_2^*, ..., v_{n_v}^*\}$ span $\mathbb{R}^d$, we have

$$\min_{O\in\mathbb{O}_d, s_i=\pm 1} \sum_{i=1}^{n_v} \|\hat{v}_i - s_i O v_i^*\|_2 \xrightarrow{p} 0,$$

$$\|\hat{\alpha} - \alpha^*\|_2 \xrightarrow{p} 0,$$

$$|\hat{\beta} - \beta^*| \xrightarrow{p} 0,$$

$$\min_{S\in\mathbb{D}_{n_v}} \|\hat{L} - SL^*S\|_F \xrightarrow{p} 0,$$

as $n_e \to \infty$.

Note that the only difference between Theorem C.1.2 and Theorem 4.3.1 is the feasible space. Because the feasible space in Theorem 4.3.1 is a subset of the one in Theorem C.1.2, one can easily show Theorem 4.3.1 is a direct corollary of Theorem C.1.2. Therefore we only need to prove Theorem C.1.2, which is actually Theorem 4 in Yu and Zhu (2023). We refer readers to Yu and Zhu (2023) for the proof of Theorem C.1.2 and conclude our proof.

C.1.2 Proof of Theorem 4.3.2

The same idea of the proof of Theorem 4.3.1 can be used for proving Theorem 4.3.2. Let Φ_p be the expected pseudo-log-likelihood, i.e., $\Phi_p = \mathrm{E}(\ell_p)$. We provide the following Lemma C.1.3

Lemma C.1.3 *It is true that*

$$\arg\max_{L\in\mathbb{S}_{n_v}^+} \Phi_p(L) = \{SL^*S | S \in \mathbb{D}\},$$

and

$$\arg\max_{\substack{V\in\mathbb{R}_{n_v\times d}, \|V_{i\cdot}\|_2=1,\\ \alpha_i\geq 0, \beta\geq 0}} \Phi_p(V, \alpha, \beta) = \{(SV^*O, \beta^*, \alpha^*) | S \in \mathbb{D} \text{ and } O \in \mathbb{O}_d\}.$$

Proof of Lemma C.1.3. Note that Φ_p is the same as Φ_2 in the proof of Lemma C.1.1. By the proof of Lemma C.1.1, we know that

$$\Phi_p(L^*) - \Phi_p(L) = \sum_{e' \subset [n_v]} P_{L^*}(e' \subset E) \times \mathrm{KL}(\mathrm{DiPH}_{e'}(L^*), \mathrm{DiPH}_{e'}(L)),$$

which is a linear combination of multiple KL divergences. Therefore,

$$\Phi_p(L^*) \geq \Phi_p(L),$$

and

$$\begin{aligned} & \Phi_p(L^*) = \Phi_p(L) \\ \iff & \mathrm{KL}(\mathrm{DiPH}_{e'}(L^*), \mathrm{DiPH}_{e'}(L)) = 0 \text{ for all } e' \subset \mathcal{V} \\ \iff & \mathrm{DiPH}_{e'}(L^*) = \mathrm{DiPH}_{e'}(L). \end{aligned}$$

As a result, we have

$$\arg\max_{L \in \mathbb{S}^+_{n_v}} \Phi_p(L) = \{SL^*S | S \in \mathbb{D}\},$$

and

$$\arg\max_{\substack{V \in \mathbb{R}_{n_v \times d}, \|V_{i\cdot}\|_2 = 1, \\ \alpha_i \geq 0, \beta \geq 0}} \Phi_p(V, \alpha, \beta) = \{(SV^*O, \beta^*, \alpha^*) | S \in \mathbb{D} \text{ and } O \in \mathbb{O}_d\},$$

which concludes our proof for Lemma C.1.3. Having Lemma C.1.3, the proof of Theorem 4.3.2 is the same as Theorem 4.3.1. So we omit the proof here.

BIBLIOGRAPHY

Agrawal, S., Garg, L., Shah, M., Agarwal, M., Patel, B., Singh, A., Garg, A., Jorde, U. P., and Kapur, N. K. (2018). Thirty-day readmissions after left ventricular assist device implantation in the united states: insights from the nationwide readmissions database. *Circulation: Heart Failure*, 11(3):e004628.

Anderson, T. W. (1958). *An introduction to multivariate statistical analysis*, volume 2. Wiley, New York.

Berge, C. (1970). *Graphes et hypergraphes*, volume 25702. Dunod, Paris.

Browne, M. W. (2000). Cross-validation methods. *Journal of Mathematical Psychology*, 44(1):108–132.

Butler, R. R. (2007). Icd-10 general equivalence mappings: Bridging the translation gap from icd-9. *Journal of American Health Information Management Association*, 78(9):84–86.

Cai, X., Gao, J., Ngiam, K. Y., Ooi, B. C., Zhang, Y., and Yuan, X. (2018). Medical concept embedding with time-aware attention. *arXiv preprint arXiv:1806.02873*.

Cameron, A. C. and Trivedi, P. K. (2013). *Regression analysis of count data*, volume 53. Cambridge University Press.

Che, Z., Purushotham, S., Cho, K., Sontag, D., and Liu, Y. (2018). Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8(1):1–12.

Chodrow, P. S. (2020). Configuration models of random hypergraphs. *Journal of Complex Networks*, 8(3):cnaa018.

Choi, E., Bahadori, M. T., Song, L., Stewart, W. F., and Sun, J. (2017a). Gram: graph-based attention model for healthcare representation learning. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 787–795.

Choi, E., Schuetz, A., Stewart, W. F., and Sun, J. (2016). Medical concept representation learning from electronic health records and its application on heart failure prediction. *arXiv preprint arXiv:1602.03686*.

Choi, E., Schuetz, A., Stewart, W. F., and Sun, J. (2017b). Using recurrent neural network models for early detection of heart failure onset. *Journal of the American Medical Informatics Association*, 24(2):361–370.

Feng, Y., Min, X., Chen, N., Chen, H., Xie, X., Wang, H., and Chen, T. (2017). Patient outcome prediction via convolutional neural networks based on multi-granularity medical concept embedding. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 770–777. IEEE.

Friel, N., Rastelli, R., Wyse, J., and Raftery, A. E. (2016). Interlocking directorates in irish companies using a latent space model for bipartite networks. *Proceedings of the National Academy of Sciences*, 113(24):6629–6634.

Fries, J. (2016). Recurrent neural networks vs. joint inference for clinical temporal information extraction. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1274–9.

Goh, K.-I., Cusick, M. E., Valle, D., Childs, B., Vidal, M., and Barabási, A.-L. (2007). The human disease network. *Proceedings of the National Academy of Sciences*, 104(21):8685–8690.

Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., and Stanley, H. E. (2000). Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220.

Greening Jr, B. R., Pinter-Wollman, N., and Fefferman, N. H. (2015). Higher-order interactions: understanding the knowledge capacity of social groups using simplicial sets. *Current Zoology*, 61(1):114–127.

Hamedani, A. G., Blank, L., Thibault, D. P., and Willis, A. W. (2021). Impact of icd-9 to icd-10 coding transition on prevalence trends in neurology. *Neurology: Clinical Practice*, 11(5):e612–e619.

Hoff, P. D. (2003). Random effects models for network data. In *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers*. Citeseer.

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

Jacobs, D. M., Noyes, K., Zhao, J., Gibson, W., Murphy, T. F., Sethi, S., and Ochs-Balcom, H. M. (2018). Early hospital readmissions after an acute exacerbation of chronic obstructive pulmonary disease in the nationwide readmissions database. *Annals of the American Thoracic Society*, 15(7):837–845.

Johnson, A., Pollard, T., and Mark III, R. (2019). Mimic-iii clinical database demo (version 1.4). *PhysioNet*, 10:C2HM2Q.

Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L.-w. H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Anthony Celi, L., and Mark, R. G. (2016). Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3(1):1–9.

Karwa, V. and Petrović, S. (2016). Discussion of" coauthorship and citation networks for statisticians". *The Annals of Applied Statistics*, 10(4):1827–1834.

Khera, R., Dorsey, K. B., and Krumholz, H. M. (2018). Transition to the icd-10 in the united states: an emerging data chasm. *Jama*, 320(2):133–134.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kolte, D., Khera, S., Sardar, M. R., Gheewala, N., Gupta, T., Chatterjee, S., Goldsweig, A., Aronow, W. S., Fonarow, G. C., Bhatt, D. L., et al. (2017). Thirty-day readmissions after transcatheter aortic valve replacement in the united states: insights from the nationwide readmissions database. *Circulation: Cardiovascular Interventions*, 10(1):e004472.

Kulesza, A., Taskar, B., et al. (2012). Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3):123–286.

Kusnoor, S. V., Blasingame, M. N., Williams, A. M., DesAutels, S. J., Su, J., and Giuse, N. B. (2020). A narrative review of the impact of the transition to icd-10 and icd-10-cm/pcs. *JAMIA Open*, 3(1):126–131.

Lambert, D. (1992). Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*, 34(1):1–14.

Lange, K. and Lange, K. (2013). Convex minimization algorithms. *Optimization*, pages 415–444.

Latała, R. (2005). Some estimates of norms of random matrices. *Proceedings of the American Mathematical Society*, 133(5):1273–1282.

Lichman, M. and Smyth, P. (2018). Prediction of sparse user-item consumption rates with zero-inflated poisson regression. In *Proceedings of the 2018 World Wide Web Conference*, pages 719–728.

Loeys, T., Moerkerke, B., De Smet, O., and Buysse, A. (2012). The analysis of zero-inflated count data: Beyond zero-inflated poisson regression. *British Journal of Mathematical and Statistical Psychology*, 65(1):163–180.

Lv, X., Guan, Y., Yang, J., and Wu, J. (2016). Clinical relation extraction with deep learning. *International Journal of Hybrid Information Technology*, 9(7):237–248.

Ma, Z., Ma, Z., and Yuan, H. (2020). Universal latent space model fitting for large networks with edge covariates. *Journal of Machine Learning Research*, 21(4):1–67.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Min, Y. and Agresti, A. (2005). Random effect models for repeated measures of zero-inflated count data. *Statistical Modelling*, 5(1):1–19.

Miotto, R., Li, L., Kidd, B. A., and Dudley, J. T. (2016). Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. *Scientific Reports*, 6(1):1–10.

Mullahy, J. (1986). Specification and testing of some modified count data models. *Journal of Econometrics*, 33(3):341–365.

Nguyen, D., Luo, W., Venkatesh, S., and Phung, D. (2018). Effective identification of similar patients through sequential matching over icd code embedding. *Journal of Medical Systems*, 42(5):1–13.

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.

Rose, C. E., Martin, S. W., Wannemuehler, K. A., and Plikaytis, B. D. (2006). On the use of zero-inflated and hurdle models for modeling vaccine adverse event count data. *Journal of Biopharmaceutical Statistics*, 16(4):463–481.

Sánchez-Valle, J., Tejero, H., Fernández, J. M., Juan, D., Urda-García, B., Capella-Gutiérrez, S., Al-Shahrour, F., Tabarés-Seisdedos, R., Baudot, A., Pancaldi, V., et al. (2020). Interpreting molecular similarity between patients as a determinant of disease comorbidity relationships. *Nature Communications*, 11(1):2854.

Shi, X., Li, X., and Cai, T. (2021). Spherical regression under mismatch corruption with application to automated knowledge translation. *Journal of the American Statistical Association*, 116(536):1953–1964.

Shickel, B., Tighe, P. J., Bihorac, A., and Rashidi, P. (2017). Deep ehr: a survey of recent advances in deep learning techniques for electronic health record (ehr) analysis. *IEEE Journal of Biomedical and Health Informatics*, 22(5):1589–1604.

Song, Y., Liu, X., Zhu, X., Zhao, B., Hu, B., Sheng, X., Chen, L., Yu, M., Yang, T., and Zhao, J. (2016). Increasing trend of diabetes combined with hypertension or hypercholesterolemia: Nhanes data analysis 1999–2012. *Scientific Reports*, 6(1):1–9.

Suresh, H., Hunt, N., Johnson, A., Celi, L. A., Szolovits, P., and Ghassemi, M. (2017). Clinical intervention prediction and understanding using deep networks. *arXiv preprint arXiv:1705.08498*.

Topaz, M., Shafran-Topaz, L., and Bowles, K. H. (2013). Icd-9 to icd-10: evolution, revolution, and current debates in the united states. *Perspectives in Health Information Management/AHIMA, American Health Information Management Association*, 10(Spring).

Tran, T., Nguyen, T. D., Phung, D., and Venkatesh, S. (2015). Learning vector representation of medical objects via emr-driven nonnegative restricted boltzmann machines (enrbm). *Journal of Biomedical Informatics*, 54:96–105.

Tu, K., Cui, P., Wang, X., Wang, F., and Zhu, W. (2018). Structural deep embedding for hyper-networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Van der Vaart, A. W. (2000). *Asymptotic statistics*, volume 3. Cambridge University Press.

Wang, Z., Ma, S., Wang, C.-Y., Zappitelli, M., Devarajan, P., and Parikh, C. (2014). Em for regularized zero-inflated regression models with applications to postoperative morbidity after cardiac surgery in children. *Statistics in Medicine*, 33(29):5192–5208.

Ward, M. D. and Hoff, P. D. (2007). Persistent patterns of international commerce. *Journal of Peace Research*, 44(2):157–175.

Ward, M. D., Siverson, R. M., and Cao, X. (2007). Disputes, democracies, and dependencies: A reexamination of the kantian peace. *American Journal of Political Science*, 51(3):583–601.

Wollman, J. (2011). Icd-10: A master data challenge. *Health Management Technology*, 32(7):16–20.

Wu, Y., Jiang, M., Lei, J., and Xu, H. (2015). Named entity recognition in chinese clinical text using deep neural network. *Studies in Health Technology and Informatics*, 216:624.

Yu, X. and Zhu, J. (2023+). Modeling hypergraph with diversity and heterogeneous popularity (submitted).

Zhang, X., Xu, G., and Zhu, J. (2022). Joint latent space models for network data with high-dimensional node variables. *Biometrika*, 109(3):707–720.

Zhen, Y. and Wang, J. (2023). Community detection in general hypergraph via graph embedding. *Journal of the American Statistical Association*, 118(543):1620–1629.

Zhou, D., Huang, J., and Schölkopf, B. (2006). Learning with hypergraphs: Clustering, classification, and embedding. *Advances in Neural Information Processing Systems*, 19.