TOWARDS LEARNING WITH HIGH CAUSAL FIDELITY FROM LONGITUDINAL EVENT DATA

by

Zhaobin Kuang

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

(Computer Sciences Department)

at the

UNIVERSITY OF WISCONSIN–MADISON

2018

Date of final oral examination: 19/07/18

Committee Member:

David Page, Department of Biostatistics and Medical Informatics, Advisor
Mark Craven, Department of Biostatistics and Medical Informatics, Thesis Reader
James Thomson, Department of Cell and Regenerative Biology, Thesis Reader
Stephen Wright, Department of Computer Sciences, Committee Member
Xiaojin Zhu, Department of Computer Sciences, Committee Member

In memory of my grandfather, Mr. Genwei Kuang (1933-2016).

ACKNOWLEDGMENTS

It is far beyond my expression to describe my gratitude towards the mentorship offered by my advisor, David Page. I am deeply indebted to David, whose generous support and patient guidance are constitutional to the fulfillment of this dissertation. David possesses infectious passion for developing better machine learning methods for medical and healthcare problems. His enthusiasm challenges and inspires me to pursue research problems with high real-world impact, and to strive to develop principled approaches with strong empirical performance and sound theoretical guarantees by identifying and resolving the unique challenges residing in these problems. While earning my doctorate under David's supervision is a landmark and culmination of my life thus far, David's profound influence towards my taste for and how to conduct research is destined to be the greatest fortune that I can ever imagine and hope for in my future as a researcher.

I am also grateful to my other committee members, Mark Craven, James Thomson, Stephen Wright, and Jerry Zhu. Mark not only directs the Center for Predictive and Computational Phenotyping, by which much of my research is funded, he is also a reader of this dissertation and an advocate of our research. His support and attention are sincerely appreciated. I am also honored and grateful to have Jamie as my dissertation reader. His insightful observations are foundational to many of our research outcomes. I am thankful to Steve for introducing me to the field of optimization. Many of our research efforts focus on optimization for machine learning, and his input is invaluable and indispensable to our results. I would also like to thank Jerry for being on my committee and being a phenomenal instructor in the machine learning courses that I have taken with him. It is these courses that equipped me with the necessary knowledge and background to pursue my research in the early stage of my study.

This dissertation is not possible without the efforts of our collaborators: Yujia Bao, Michael Caldwell, Sinong Geng, Jie Liu, Richard Maclin, David Page, Peggy Peissig, Vitor Santo Costa, Ron Stewart, James Thomson, Rebecca Willett, and Stephen Wright. I would like to gratefully acknowledge their inputs for the research

towards my dissertation.

I would also like to express my deep indebtedness to the advisors of my Master's program at the University of Minnesota, Duluth, Zhuangyi Liu and Richard Maclin, for preparing and helping me in every way possible at the time to pursue my doctorate, and for their sustained commitment and attention to my career development over the years.

Finally, I would also like to thank my family for their understanding, sacrifice, and perseverance over the years. As the single child in my family, I have spent almost all of the past six years in the United States to pursue my graduate study ever since I graduated from college in China. Many life-changing hardships took place at home during this time, and I salute my family's courage and efforts for conquering the challenges in life in return for my undivided attention to my Ph.D. study.

CONTENTS

ABSTRACT

Longitudinal event data (LED) are irregularly time-stamped multi-type event sequences collected from heterogeneous subjects throughout different time scales. In this dissertation, we are interested in developing machine learning models and algorithms to identify potential causal relationships among various event types from LED so as to provide actionable insights for better decision-making.

As a concrete example of LED, we consider the use of electronic health records (EHRs). By viewing the occurrences of different drug prescriptions, condition diagnoses, and physical measurements as different event types, we are interested in identifying potential causal relationships regarding how different drugs could influence the occurrences of various conditions and the values of different physical measurements. This problem leads to two pivotal health applications: computational drug repositioning (CDR) and adverse drug reaction (ADR) discovery.

To deliver better CDR and ADR discovery, we focus on developing machine learning models and algorithms with high causal fidelity. Causal fidelity is concerned with whether a method can effectively identify signals residing in the data that indicate potential causality. By confronting various theoretical, methodological, and empirical issues stemming from the intricacies of LED, our models and algorithms strive to deliver high causal fidelity via the identification of signals in LED that are reflective of potential causal relationships among various event types. This leads to the title of the dissertation, *Towards Learning with High Causal Fidelity from Longitudinal Event Data*.

The primary content of the dissertation is hence to present how high causal fidelity can be achieved in CDR, ADR discovery, and beyond. Our solution is to identify and address three fundamental challenges constitutional to the intrinsic nature of LED - *inhomogeneity*, *irregularity*, and *interplay* - summarized as the 3-I challenge. We demonstrate that by a careful treatment of the 3-I challenge, it is possible to develop machine learning models and algorithms with high causal fidelity, as shown by the improved performance of CDR and ADR discovery exhibited in this dissertation.

Part I

Preliminaries

1 INTRODUCTION

Longitudinal event data (LED) are irregularly time-stamped multi-type event sequences collected from heterogeneous subjects throughout different time scales. The ubiquity of LED has been redefining decision-making in numerous domains as a data-driven process. Examples abound:

- In business analytics, purchasing events of different items from millions of customers are collected, and retailers are interested in how a distinct market action or the sales of one particular type of item could boost or hinder the sales of another type (Han et al., 2011).
- In search analytics, web search keywords from billions of web users are usually mapped into various topics (e.g. travel, education, weather), and search engine providers are interested in the interplay among these search topics for a better understanding of user preferences (Gunawardana et al., 2011).
- In health analytics, electronic health records (EHRs) contain clinical encounter events from millions of patients collected over decades, including drug prescriptions and condition diagnoses, among others. Unraveling the relationships between different drugs and different conditions is vital to answering some of the most pressing medical and scientific questions such as drug-drug interaction detection (Tatonetti et al., 2012), comorbidity identification, adverse drug reaction (ADR) discovery (Simpson et al., 2013; Bao et al., 2017a; Kuang et al., 2017c), computational drug repositioning (Kuang et al., 2016a,c), and precision medicine (Liu et al., 2013a, 2014a).

All these data analytics problems beg a foundational question in machine learning: *can we identify potential causal relationships among various event types from LED in order to provide actionable insights for decision-making?*

This dissertation is dedicated to providing an affirmative answer to the aforementioned question via the development of machine learning models and algorithms

with high causal fidelity. Causal fidelity is concerned with whether a method can effectively identify signals residing in the data that indicate potential causality. By confronting various theoretical, methodological, and empirical issues stemming from the intricacies of LED, our approaches strive to achieve high causal fidelity via the identification of signals in LED that are reflective of potential causal relationships among various event types. This leads to the title of the dissertation, *Towards Learning with High Causal Fidelity from Longitudinal Event Data*.

Throughout the dissertation, as an example of LED, we use EHRs as our data source, where various drug prescription records, condition diagnosis records, and physical measurement records are collected from a massive number of patients over long periods of time. We view the occurrences of different drugs, conditions, and physical measurements as different event types. To empirically demonstrate the high causal fidelity of our machine learning models and algorithms to the relationships among these event types, we consider two pivotal health applications: computational drug repositioning (CDR) and adverse drug reaction (ADR) discovery. The primary content of the dissertation is hence to present how high causal fidelity can be achieved in these two applications and beyond. Our solution, in a nutshell, is to identify and address three fundamental challenges constitutional to the intrinsic nature of LED - *inhomogeneity*, *irregularity*, and *interplay* - summarized as the 3-I challenge.

While for the ease of presentation our discussion centers around CDR and ADR discovery, the conclusions and lessons learned from this dissertation are widely applicable to other scenarios where learning with high causal fidelity from LED is essential.

1.1 CDR and ADR Discovery

CDR is the task of finding new indications (uses) for existing drugs using computational methods. In recent years, CDR has been steadily rising to prominence. In 2013 alone, among the 84 drug products introduced to the US market, 20% of them were due to new indications for existing drugs (Li et al., 2016). Compared with traditional

de novo drug development, CDR can be cheaper, faster and safer. Traditionally, developing a new drug from scratch is expensive due to pharmacological research as well as clinical trails conducted for the drug. Furthermore, the fulfillment of such a drug development process can take decades. A new drug could also be unsafe as it might generate rare harmful events that were not observed before. On the other hand, since CDR identifies new indications for existing drugs, a significant portion of the drug developmental process can be bypassed, resulting in potentially substantial savings in money and time. Furthermore, since existing drugs have usually been available on the market for a period of time, their pharmacological properties are relatively well-known, potentially decreasing the risks of taking these drugs due to the uncertainty in their safety profiles.

In contrast to CDR, *ADR discovery is the task of finding unexpected, and negative effects of drugs when prescribed to patients.* ADR is a major public health challenge. It is estimated that ADRs cause 4.2-30% of hospitalizations in the United States and Canada, with an approximated relevant annual cost of 30.1 billion US dollars in the United States (Sultana et al., 2013). Although the U.S. Food and Drug Administration (FDA) has established one of the most rigorous drug preapproval procedures in the world, many potential ADRs of a drug may not be identified in the developmental stage. During the preapproval clinical trials, a drug might be tested on just at most a few thousand people. Therefore, ADRs with low occurrence rates are not likely to be identified in this relatively small population. However, these ADRs might occur and even become a public heath concern after the drug is introduced to the market, where potentially millions of people with much more diverse physiological profiles are taking the drug. Therefore, post-approval surveillance methods that can effectively detect potential ADRs in time are highly desirable to address this major public health challenge.

1.2 Electronic Health Records

Large-scale clinical longitudinal event data, such as electronic health records and insurance claim data, provide a unique data source of potentially invaluable knowl-

Drug Prescription Records

PATIENT_ID	DRUG_NAME	PRESCRIPTION_DATE
1	HUMALOG	Jan-28-2005
1	HUMALOG	Jun-17-2005
2	INSULIN	Mar-07-1998

Fasting Blood Glucose Records

PATIENT_ID	DATE	VALUE
1	Jan-28-2005	130
2	Apr-13-1998	95
2	Aug-12-1998	140

Condition Diagnosis Records

PATIENT_ID	DATE	CONDITION
1	Feb-07-2006	DIABETES
2	May-25-1997	BLEEDING

Figure 1.1: Electronic Health Records (EHRs)

edge for the tasks of CDR and ADR discovery. Figure 1.1 illustrates a set of EHRs for two patients. For each patient during his/her observational period, various types of information about the patient, such as drug prescriptions, condition occurrences, physical measurements, and demographic information are collected.

The diverse and abundant patient-oriented information available in EHRs might encode potential correlational and even causal information that is yet to be discovered. For example, an unexpected decrease in blood sugar level of a patient after a prescription of a drug that was not previously known to have a blood sugar decreasing effect might imply a potential CDR candidate for blood sugar control, while the occurrence of an unanticipated condition after a drug prescription might indicate an unknown ADR of that drug.

Nevertheless, the identification of the aforementioned potential effects of the drugs from this type of LED can be extremely challenging. In this dissertation, we identify and address three fundamental challenges constitutional to the intrinsic nature of LED: *inhomogeneity*, *irregularity*, and *interplay*. They are summarized as

the 3-I challenge in Section 1.3.

1.3 The 3-I Challenge

1.3.1 Inhomogeneity

Subject and time inhomogeneities are induced by the longitudinal nature of LED, where data from potentially millions of diverse subjects might be collected over a time span of decades. For example, in EHRs, the baseline occurrence rates of a heart attack (myocardial infarction, MI) are different among distinct patients (subject inhomogeneity), because some patients are in poorer health than others and hence are more prone to suffer an MI. Even within a patient, the baseline occurrence rate of MIs changes over time (time inhomogeneity): an elderly person might be more inclined to suffer an MI compared to when he or she was younger; the recurrence of MIs also tends to be higher compared to first occurrence. Furthermore, these inhomogeneities might not even be directly observed in real-world data.

To account for inhomogeneity, we propose and evaluate the *baseline regularization* (BR) models (Kuang et al., 2016a, 2017c), where subject and time inhomogeneities of the baseline occurrence rates of a condition are taken into consideration when we model how prescriptions of various drugs can alter the occurrence of the condition in question. BR models improve performance for ADR discovery from EHRs. They also offer the first approach to CDR using EHRs. Details of BR models will be presented in Part II.

1.3.2 Irregularity

In LED, events occur spontaneously and irregularly (e.g. patients only visit doctors when necessary instead of on a daily basis). Therefore, instead of having a full observation of all the variables at any time point, LED are temporally irregular in nature.

Our attempt at directly modeling the temporal irregularity of LED is via the use of point process models (Bao, Kuang et al., 2017) in an ADR discovery from EHRs setting. Intuitively, if the ADR in question is an acute effect of a particular drug, then the corresponding drug prescription events that occur right before the speculated adverse reaction event should deserve more attention. In contrast, if an ADR is due to long-term and high-dosage use of a medication, then all the corresponding drug prescription events in the past should be taken into consideration. Handling irregularity will be the focus of Part III, where we will also present a model variant (Kuang et al., 2016a, 2018a) for CDR.

1.3.3 Interplay

The goal of causal fidelity in this dissertation dictates the requirement of a more thorough understanding of the interplay among different event types of LED that arise from the intricate dynamics of nature and human activities. For example, in EHRs, effects of thousands of drugs on thousands of conditions and physical measurements need to be evaluated. To make things more complicated, when prescribed simultaneously, different drugs can interact with each other, resulting in unexpected conditions or changes in physical measurements that might not occur when these drugs are taken individually.

A graphical model representing the joint distribution among all the drugs and conditions can offer insights via its structure into the interplay among different variables, yielding potentially interesting and improved causal relationship discovery. Indeed, many leading causal inference approaches (Spirtes et al., 2000; Kalisch and Bühlmann, 2007; Pearl, 2009; Ogarrio et al., 2016; Hernan and Robins, 2018) are based on learning from graphical models. Our contribution is to develop more efficient and effective algorithms for graphical model learning. Since LED usually consist of binary or count variables (e.g. *whether* vs *how many times* a drug is prescribed to a particular patient on a particular day, or *whether* vs *how many times* a condition is diagnosed for a particular patient on a particular day), we focus specifically on graphical model learning algorithms over multivariate binary and

Table 1.1: Organization of the Dissertation

Challenge	Chapter	Paper & Venue	Application
Inhomogeneity	3	Kuang et al., KDD16	CDR
	4	Kuang et al., KDD17	ADR
Irregularity	5	Bao, Kuang et al., MLHC17	ADR
	6	Kuang et al., Invited Book Chapter Kuang et al., IJCAI16	CDR
Interplay	7	Geng*, Kuang* et al., ICML18	ADR, Count Data
	8	Geng*, Kuang* et al., UAI18	Binary Data
	9	Kuang et al., NIPS17	Binary Data

* indicates equal contribution. Authors are listed in α-β order.

count distributions (Kuang et al. 2017a, Geng*, Kuang* et al., 2018a; Geng*, Kuang* et al., 2018b). The details are reported in Part IV.

1.4 Thesis Statement

This dissertation supports the following thesis:

> *By identifying and addressing three fundamental challenges constitutional to the intrinsic nature of longitudinal event data - inhomogeneity, irregularity, and interplay - we are able to deliver machine learning models and algorithms with high causal fidelity that provide actionable insights for better decision-making.*

1.5 Organization

The rest of the dissertation is organized as follows. The inhomogeneity challenge is addressed by Chapter 3 and Chapter 4. Using CDR as an example, individual-inhomogeneity is the subject of Chapter 3, while time-inhomogeneity is the subject of Chapter 4 using ADR discovery as an example. The irregularity challenge is addressed in Chapter 5 and Chapter 6, where ADR discovery and CDR are considered

as applications, respectively. Chpater 9, 8, and 7 deal with the interplay challenge via the use of graphical models over multivariate binary/count distributions, with an application to ADR discovery in Chapter 7. The organization is summarized in Table 1.1.

2 RELATED WORK

This chapter describes previous work in the literature that is related to our research problems. In subsequent sections, we will give a brief overview of related work on CDR, ADR discovery, high-dimensional statistical learning, and causal fidelity.

2.1 Related Work on CDR

With the advent of the big data era, abundant data sources that collect rich drug-related information are emerging. Leveraging these large-scale heterogeneous drug-related data sources, CDR has become an active research area that has the potential to deliver more effective drug repositioning. There have been several comprehensive reviews in the literature on CDR (Hurle et al., 2013; Li et al., 2016). Many methods leverage genotypic and transcriptomic information (Lamb, 2007; Kuhn et al., 2010), as well as drug molecular structure and drug combination information (Liu et al., 2010b; Knox et al., 2011). A prior study that used EHRs to validate a potential indication of *one* existing drug has also been reported in Xu et al. (2014). However, to the best of our knowledge, research projects that explore EHRs to identify a potential indication from *multiple* existing drugs simultaneously have not been reported in the literature. Our proposed algorithms will address exactly this setting.

2.2 Related Work on ADR Discovery

To provide post-approval surveillance methods that can effectively detect potential ADRs, in 2008 the Observational Medical Outcomes Partnership (OMOP 2016c) was launched to conduct methodological research for medical product safety surveillance. In mid-2013, OMOP became the groundwork of the Innovation in Medical Evidence Development and Surveillance (IMEDS) program (IMEDS, 2016), whose goal is to further advance the research for post-market safety surveillance of med-

ical products. Recently, the Observational Health Data Sciences and Informatics (OHDSI) program was established. Independent of IMEDS, OHDSI engages many of the original investigators in OMOP as part of its research team, with the mission of developing and applying methods to observational data to answer real-world clinical questions (OHDIS, 2016).

An important contribution from OMOP was the establishment of a set of ground truth ADRs for methodological validation (OMOP, 2016b). Based on the opinions of expert panels, 9 drug-condition pairs were identified as ADRs, 44 were identified as negative controls, and 2 were identified as positive benefits. With this set of ground truth, experiments conducted by OMOP suggest that methods using a self-controlled design are the best performers (OMOP, 2016a).

A major inspiration for our proposed methods is a self-controlled method (Farrington et al., 2018) called self-controlled case series (SCCS). SCCS is a type of Poisson regression model originally designed to capture adverse events for the evaluation of vaccine safety (Farrington, 1995). An SCCS model uses its regression coefficients to estimate the ADR occurrence rate under drug exposures. The contrast of the ADR occurrence rates between exposed and unexposed periods as well as the control of time-invariant confounding factors are addressed implicitly by subtle probability conditioning, since each patient has both exposed and unexposed time periods, and so can serve as his/her own control. In the multiple SCCS setting (Simpson et al., 2013), for a particular ADR occurrence, many concomitant drug exposures can be taken into account at the same time as different regressors to evaluate the impact of each drug on the occurrence of the ADR.

2.3 Related Work on Statistical Learning

Many of the machine learning models and algorithms presented in this dissertation address (high-dimensional) statistical learning problems. Solving these problems can usually be formulated as the sum of a loss function and a regularizer (Negahban

et al., 2009):

$$\arg\min_{\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{\beta}; \mathbf{X}) + \mathcal{R}(\boldsymbol{\beta}). \tag{2.1}$$

In (2.1), $\mathbf{X}$ represents the data and $\boldsymbol{\beta}$ represents the parameters of the optimization problem. $\mathcal{L}(\boldsymbol{\beta}; \mathbf{X})$ represents the loss function and $\mathcal{R}(\boldsymbol{\beta})$ represents the regularizer. The loss function is usually chosen to be smooth and convex (e.g. the square loss function) and $\mathcal{R}(\boldsymbol{\beta})$ is usually chosen to encourage a certain type of *sparsity* in the parameters.

Since the seminal work of Tibshirani (1996) on the least absolute shrinkage and selection operator (LASSO, or lasso), tremendous research efforts have been invested in the field of high-dimensional statistics to provide better sparse estimation procedures with provable statistical and optimizational guarantees. Comprehensive reviews on the field of high-dimensional statistical learning can be found in Friedman et al. (2001), Bühlmann and Van De Geer (2011), and Hastie et al. (2015), among others.

2.4 Related Work on Causal Fidelity

Causal fidelity concerns whether a method can effectively identify signals residing in the data that indicate potential causality. Formally and conventionally, the field of causal inference offers a broad spectrum of approaches to achieve high causal fidelity with theoretical guarantees in various scenarios (Granger, 1969; Pearl, 2009; Imbens and Rubin, 2015; Hernan and Robins, 2018). However, many of the aforementioned methods hinge on restrictive assumptions (e.g. no unobserved confounding assumption, a.k.a. NUCA) to draw formal inference in causality. They are also not well-equipped to deal with massive quantities of LED characterized by the 3-I challenges.

Perhaps the most relevant work to this dissertation is the notion of Granger causality in a point process setting (we say that a covariate "Granger causes" a response if the history of the covariate in question can improve the performance of predicting the value of the response, see for example Granger 1969 and Xu et al.

2016), as well as causal inference via the use of Bayesian networks (a.k.a. directed graphical models, Pearl 2009). Nonetheless, point processes models (Gunawardana et al., 2011; Weiss et al., 2012; Weiss and Page, 2013; Du et al., 2016) usually focus on pinpointing the exact occurrence times of events, and hence might be inadequate to model EHRs because the timestamp information collected in EHRs can be noisy. Furthermore, the complexity of learning a Bayesian network from scratch can be intimidating because of the vast search space (Spirtes et al., 2000; Kalisch and Bühlmann, 2007; Ogarrio et al., 2016).

In contrast, throughout this dissertation, we will be presenting approaches that leverage event histories (Granger causality) without focusing on predicting the time-point when an event occurs. We will also be focusing on presenting efficient learning algorithms for undirected graphical models (a.k.a. Markov random field) instead of for Bayesian networks; this decision is without loss of generality, as an undirected graphical model can serve as a template to construct Bayesian networks.

Another branch of research that adheres to achieving high causal fidelity empirically from complex real world data is the use of linear mixed models (LMMs) for genome-wise association studies (GWAS) from single nucleotide polymorphism (SNP) data. By identifying and modeling the factors that could potentially confound causal relationships in the SNP data, a series of LMMs (Lippert et al., 2011; Widmer et al., 2014; Weissbrod et al., 2015; Heckerman et al., 2016; Kadie and Heckerman, 2017; Heckerman, 2018) is developed to deliver more efficient GWAS with higher causal fidelity. Despite the empirical success of LMMs, a formal causal justification usually receives less attention due to the complex nature of SNP data that renders inapplicable many assumptions indispensable for formal justification. Delivering causal fidelity from LED faces similar issues. In this dissertation, we will be focusing on improving empirical performance of CDR and ADR discovery by addressing the 3-I challenge.

Part II

Inhomogeneity

EHRs are longitudinal in nature: health statuses of many patients are reflected by EHRs over a long period of time. Therefore, inhomogeneity of the data not only occurs among distinct subjects, but also evolves substantially over time. In Part II, we present our research that addresses subject-inhomogeneity (Chapter 3) and time-inhomogeneity (Chapter 4). Our approaches offer the first attempt at CDR using EHRs (Chapter 3), and improve the performance of ADR discovery (Chapter 4). Thus this dissertation suggests that high causal fidelity can be achieved in practice by modeling the inhomogeneity of the data appropriately.

3 COMPUTATIONAL DRUG REPOSITIONING VIA CONTINUOUS SELF-CONTROLLED CASE SERIES

3.1 Introduction

Computational drug repositioning (CDR) is the task of identifying new potential indications for existing drugs using computational approaches and drug-related data sources. As noted in Chapter 1, CDR has becoming increasingly important because the traditional process of *de novo* drug discovery can be slow, expensive, and risky (Ashburn and Thor, 2004). There have been several comprehensive reviews in the literature on CDR (Hurle et al., 2013; Li et al., 2015). Many methods leverage genotypic and transcriptomic information (Lamb, 2007; Kuhn et al., 2010), as well as drug molecular structure and drug combination information (Liu et al., 2010b; Knox et al., 2011). A prior study that used Electronic health records (EHRs) to validate a potential indication of *one* existing drug has also been reported (Xu et al., 2014).

We are interested in identifying a potential indication from *multiple* existing drugs simultaneously using EHR. As an initial attempt, we examine the *numeric* values of fasting blood glucose (FBG) level recorded in patients' EHRs *before* and *after* some drugs are prescribed to those patients, in the hope of identifying previously unknown potential uses of drugs to control blood glucose level. For this purpose, we extend the self-controlled case series (SCCS) model that has been widely used in the adverse drug reactions (ADRs) discovery community (Simpson et al., 2013) to handle *continuous* numeric response, hence the name of our model, continuous self-controlled case series (CSCCS).

Since EHRs are usually collected from a massive number of individuals with diverse health profiles, identifying and modeling the inhomogeneity among different patients (e.g. a diabetic patient tends to have higher FBG levels in general compared to a healthy person) is crucial to the causal fidelity of a CDR algorithm. CSCCS offers a solution via the use of self-controlled designs, where each patient serves as

his/her own control, so that the changes of FBG levels within an individual can be attributed to how drug prescription history of the individual could potentially influence the FBG level every time such a measurement is taken. For example, an antibiotic drug taken ten years ago might have less, if any, influence on the FBG level than an anti-diabetic drug taken a day before that FBG level is measured. To determine how long a drug can potentially influence a patient, we furthermore propose a data-driven approach that leverages change point detection (Muggeo, 2003), resulting in estimations of different time spans of influence for different drugs. Our contributions are three-fold:

- To the best of our knowledge, this is the first translation of SCCS methodology from ADR discovery to CDR. Our work is a pilot study evaluating the use of temporal ordering information between numeric physical measurements and drug prescriptions available in EHRs for the knowledge discovery process of CDR.
- Based on the insightful observations of Xu et al. (2012), we derive our CSCCS model from a fixed effect model and hence extend the original SCCS model to address continuous numeric response variables.
- We introduce to the CDR and ADR discovery community a data-driven approach for adaptively determining the time spans of influence of different drugs to the patients.

3.2 Continuous Self-Controlled Case Series

3.2.1 Notation

Figure 3.1 visualizes an example of health records for two patients. To confine the time span of a drug that has potential influence on that patient, we use the concept of *drug era*, which is recorded with its start date, end date and the name (or id) of the drug. We consider a patient to be under consistent influence of a drug during a

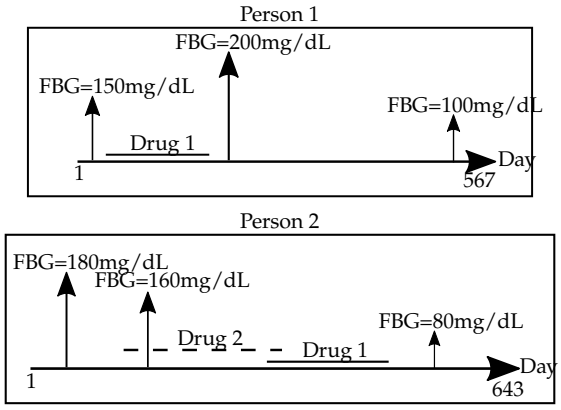


Figure 3.1: An example of electronic health records for two patients with drug eras and fasting blood glucose (FBG) measurements.

drug era of that drug. However, drug era information is not readily available in most EHRs. Instead, drug prescription information with the name of a drug and the start date of the prescription is usually provided in observational data. How to construct drug eras from prescription records is a challenging and significant task for both CDR and ADR discovery (Nadkarni, 2010; Ryan, 2010). We provide a data-driven approach to this task in Section 3.4.

Let there be N patients with FBG measurements and M different drugs in the EHR. We construct a cohort using all the FBG measurement records as well as all the drug era records from all the N patients. Furthermore, we use a continuous random variable y_{ij}, where $i \in \{1, 2, \cdots, N\}$, $j \in \{1, 2, \cdots, J_i\}$, to denote the value of the j^{th} FBG measurement taken among a total number of J_i measurements during the observation period of the i^{th} person. Similarly, we use a binary variable x_{ijm}, $i \in \{1, 2, \cdots, N\}$, $j \in \{1, 2, \cdots, J_i\}$, $m \in \{1, 2, \cdots, M\}$ to denote the exposure status of the m^{th} drug of the i^{th} person at the date when the j^{th} FBG measurement is taken, with 1 representing exposure and 0 otherwise.

3.2.2 The Linear Fixed Effect Model

We treat the y_{ij}'s as the response variables and first consider the following linear regression model:

$$\mathbb{E}[y_{ij}|\mathbf{x}_{ij}] = \alpha_i + \boldsymbol{\beta}^\top \mathbf{x}_{ij}, \tag{3.1}$$

where

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_M \end{bmatrix}^\top, \quad \mathbf{x}_{ij} = \begin{bmatrix} x_{ij1} & x_{ij2} & \cdots & x_{ijM} \end{bmatrix}^\top,$$

α_i, which is called the *nuisance parameter*, represents the individual effect of the i^{th} person on the value of y_{ij}, invariant to day j, drug m, and other patients, and $\mathbb{E}[\cdot]$ represents the expectation.

The parameter of interest in this problem is $\boldsymbol{\beta}$, which represents the effect of each of the M drugs on the response $\mathbf{y}$ when a patient is under the joint exposure statuses specified by $\mathbf{x}_{ij}$. More specifically, suppose the m^{th} component of $\boldsymbol{\beta}$, β_m, is evaluated to a negative number, that is to say, exposure to the m^{th} drug will cause the FBG level to decrease. If this drug is not known to be prescribed for lowering FBG, such a decrease is an indicator that this drug might have the potential to be repositioned to help diabetic patients control their blood glucose level, given further investigation.

In this setting, fitting a linear regression model is equivalent to solving the following least squares problem:

$$\arg\min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \frac{1}{2} \left\| \mathbf{y} - \begin{bmatrix} \mathbf{Z} & \mathbf{X} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix} \right\|_2^2, \tag{3.2}$$

where

$$\begin{aligned} \boldsymbol{\alpha} &= \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_N \end{bmatrix}^\top, \quad \mathbf{Z} = \operatorname{diag}(\mathbf{1}_1, \cdots, \mathbf{1}_N), \\ \mathbf{y} &= \begin{bmatrix} y_{11} & \cdots & y_{1J_1} & \cdots & y_{N1} & \cdots & y_{NJ_N} \end{bmatrix}^\top, \\ \mathbf{X} &= \begin{bmatrix} \mathbf{x}_{11} & \cdots & \mathbf{x}_{1J_1} & \cdots & \mathbf{x}_{N1} & \cdots & \mathbf{x}_{NJ_N} \end{bmatrix}^\top, \end{aligned}$$

where $\mathbf{Z}$ is a block diagonal matrix with $\mathbf{1}_i$ being a $J_i \times 1$ vector where all the components are 1. The least squares problem in (3.2) is a linear *fixed effect model* with $\boldsymbol{\alpha}$ being a nonrandom quantity whose i^{th} component α_i, can be interpreted as the *average* FBG measurement level of the i^{th} patient taken over time without exposing to any drugs.

3.2.3 Deriving CSCCS from the Linear Fixed Effect Model

Like the SCCS model, the motivation behind the CSCCS model is to use only $\boldsymbol{\beta}$ as a parsimonious parameterization to predict the response vector $\mathbf{y}$. Inspired by the work in (Xu et al., 2012), where the equivalence between the Poisson fixed effect model and the SCCS model is established, we are able to derive the CSCCS model from the linear fixed effect model in (3.2) in a similar fashion. Let

$$\ell(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2} \left\| \mathbf{y} - \begin{bmatrix} \mathbf{Z} & \mathbf{X} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix} \right\|_2^2.$$

We consider,

$$\frac{\partial \ell(\boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial \boldsymbol{\alpha}} = \mathbf{0} \Rightarrow \boldsymbol{\alpha} = \left(\mathbf{Z}^\top \mathbf{Z}\right)^{-1} \mathbf{Z}^\top (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \bar{\mathbf{y}} - \bar{\mathbf{X}}\boldsymbol{\beta}, \tag{3.3}$$

where $\bar{\mathbf{y}}$ is an $N \times 1$ vector with the i^{th} component, $\bar{y}_i = \frac{1}{J_i} \sum_{j=1}^{J_i} y_{ij}$, and $\bar{\mathbf{X}}$ is an $N \times M$ matrix with the i^{th} row, $\bar{\mathbf{X}}_{i\cdot} = \frac{1}{J_i} \sum_{j=1}^{J_i} \mathbf{x}_{ij}^\top$. Substituting (3.3) into (3.2) results in the CSCCS model:

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2} \|\mathbf{y} - \mathbf{Z}\bar{\mathbf{y}} - \left(\mathbf{X} - \mathbf{Z}\bar{\mathbf{X}}\right)\boldsymbol{\beta}\|_2^2. \tag{3.4}$$

The model in (3.4) is in the desired form of parsimonious parameterization in that the optimization problem is defined only in the space of $\boldsymbol{\beta}$, and the nuisance parameter $\boldsymbol{\alpha}$ is eliminated.

The CSCCS model is a linear model and hence CSCCS is able to predict *continuous* response $\mathbf{y}$. The model is *self-controlled* in that each FBG measurement and

their corresponding drug exposure statuses are adjusted by their mean within each individual. The model also utilizes *case series* in that only cases (patients that have at least one FBG measurement) are admitted in the cohort.

CSCCS is derived from its linear fixed effect model counterpart. This derivation shares the same spirit with the equivalence between the original SCCS and the Poisson fixed effect model; in this sense, CSCCS extends SCCS to address numeric response in the new setting.

Although both models in (3.2) and (3.4) can be considered as linear models, from the perspective of implementation efficiency, the explicit form of CSCCS in (3.4) is of vital importance for the task of CDR using large-scale EHRs. This is because the parameter of interest in our task is $\boldsymbol{\beta}$ and the nuisance parameters do not provide direct information in evaluating the impact of a drug in changing FBG level. In the setting of large-scale EHRs, where tens of thousands of patient records might be admitted into the cohort as cases, the dimension of the nuisance parameter can potentially be very high. In this scenario, without the access to a special purpose solver for the fixed effect model, solving a model in the form of (3.2) using only a general purpose linear model solver can be time consuming or even infeasible. On the contrary, using the explicit form of CSCCS in (3.4), a general purpose linear model solver only needs to find solutions in the space of $\boldsymbol{\beta}$, a parameter whose dimension is only as large as the number of drugs available in the cohort, which is a much smaller number than the dimension of nuisance parameters.

3.3 Challenges in EHR data

Several challenges arise when we apply CSCCS to EHR data. In this section, we present the further refinements we perform on the CSCCS model presented in (3.4) in order to address these challenges.

3.3.1 High Dimensionality

EHR data is a type of high-dimensional longitudinal data. While tens of thousands of patient records might be admitted into the cohort, effects of thousands of drugs on the FBG level need to be evaluated simultaneously, introducing a high-dimensional problem. This motivates us to incorporate sparsity into our model using the penalty (Tibshirani, 1996),

$$\arg\min_{\beta} \frac{1}{2}\|\mathbf{y} - \mathbf{Z}\bar{\mathbf{y}} - \left(\mathbf{X} - \mathbf{Z}\bar{\mathbf{X}}\right)\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1, \tag{3.5}$$

where $\lambda > 0$ is a tuning parameter determining the level of sparsity.

The incorporation of this penalty essentially assumes that only a small portion of drugs are related to the change of FBG level, and the rest of them do not have significant effect on changing FBG level when patients are exposed to those drugs. With the L_1 penalization, most components of $\boldsymbol{\beta}$ will be evaluated to zero or a number that is close to zero. The result is, instead of evaluating the effect of *each* of the M drugs on FBG level, L_1 penalized CSCCS only selects a subset of drugs that, in some sense, are most correlated to the change of FBG level, and estimates their relative strength and direction of change among the drugs chosen.

3.3.2 Irregular Time Dependency

The linear fixed effect model assumes that all responses are independent of each other. The meaning of independence is two-fold. On one hand, responses from different patients are independent of each other. To explain differences across patients (e.g. some patients tend to have higher FBG levels than others in general), $\boldsymbol{\alpha}$ is used with each component representing the time-invariant effect of each patient on the response. On the other hand, responses observed at different times are independent of each other. To explain differences across time (e.g. FBG levels observed in early age *might be* lower than those in old age), a time-dependent variable that has the same value across all patients can be introduced. That is to

say:

$$\mathbb{E}[y_{ij}|\mathbf{x}_{ij}] = \alpha_i + t_j + \boldsymbol{\beta}^\top \mathbf{x}_{ij}, \tag{3.6}$$

where t_j is the time-dependent nuisance parameter whose value depends only on the time when the j^{th} measurement is taken. If observations are recorded regularly across time, (3.6) defines a *two-way fixed effect model*, as opposed to the *one-way fixed effect model* defined in (3.2) (Frees, 2004).

In practice, a one-way model might be preferred over a two-way model if we assume that the heterogeneity across different individuals is much more significant than that across time. However, in the task of CDR from EHRs, this assumption might be too restrictive. To begin with, EHRs usually contain observational data of patients that are recorded over decades. Therefore, it is probable that the baseline FBG levels of patients change significantly over the years. This is especially true when some persistent FBG level altering events, such as the diagnosis of diabetes, occur to some patients. Furthermore, the length of observation periods varies dramatically among patients. Therefore, we do not have a fully observed and consistent dataset to model the set of time-dependent nuisance parameters. Last but not least, the incorporation of time-dependent nuisance parameters is proposed in a setting where data are collected regularly. With the irregular nature of EHR data, modeling time-dependent nuisance parameters directly with a classic two-way fixed effect model is impractical.

To address the aforementioned challenges without much loss in efficiency, we consider a reasonable assumption: given y_{ij} and $y_{ij'}$, where $j \neq j'$, but the dates of the two measurements taken are very close to each other, we assume the two corresponding time-dependent nuisance parameters are equal to each other, i.e. $t_j = t_{j'}$. More specifically,

$$\mathbb{E}[y_{ij}|\mathbf{x}_{ij}] = \alpha_i + t_j + \boldsymbol{\beta}^\top \mathbf{x}_{ij}, \quad \mathbb{E}[y_{ij'}|\mathbf{x}_{ij'}] = \alpha_i + t_{j'} + \boldsymbol{\beta}^\top \mathbf{x}_{ij'},$$
$$|d_{ij} - d_{ij'}| \leqslant \tau \Rightarrow t_j = t_{j'},$$

where d_{ij} and $d_{ij'}$ represent that the j^{th} and j'^{th} measurements of the i^{th} patient are

taken at the d_{ij}^{th} day and $d_{ij'}^{th}$ day of the observation period, and τ is a predetermined threshold. Then,

$$\mathbb{E}\left[y_{ij} - y_{ij'} | \mathbf{x}_{ij}, \mathbf{x}_{ij'}\right] = \boldsymbol{\beta}^\top \left(\mathbf{x}_{ij} - \mathbf{x}_{ij'}\right) \equiv \boldsymbol{\beta}^\top \boldsymbol{\delta}_{ij}, \tag{3.7}$$

where the nuisance parameters are eliminated. Therefore, the quantity in (3.7) depends only on $\boldsymbol{\beta}$ and the data.

Based on this formulation, we can reconstruct the CSCCS model to address irregular time dependency as follow. Firstly, given τ, construct a cohort of patients that have at least two consecutive FBG measurements taken within τ days. Secondly, solve the following lasso problem:

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2} \|\mathbf{D}\mathbf{y} - \mathbf{D}\mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1, \tag{3.8}$$

where $\mathbf{D}$, when multiplied with $\mathbf{y}$ or $\mathbf{X}$, generates the difference between the measurement of an earlier record and the corresponding measurement of its adjacent later-measured record of the same patient, with the constraint that the two records are collected within a time span of τ days.

Note that the model in (3.8) is not equivalent to the model in (3.5). However, the model in (3.8) can still be considered as a variant of CSCCS in that its parameterization is still restricted to $\boldsymbol{\beta}$, with the goal of predicting a continuous response, using data subtraction within the *same* patient as a self-controlled mechanism, and only admitting cases into the cohort. We call the the model in (3.8) as CSCCS for adjacent response, or CSCCSA.

3.3.3 Confounding

Another challenge an algorithm must tackle is the confounding issue arises due to the complex nature of clinical observational data. In the setting of EHRs, one important confounding issue is called *confounding by co-medication*. Consider drug A and drug B, where only drug A can lower FBG level and drug B has no significant effect on changing blood sugar. However, drug B is usually prescribed with drug A.

In this case, drug B can be a confounder if we only evaluate the marginal correlation between each drug and FBG level. Another confounding issue in this setting is *confounding by comorbidity*. Consider the FBG-lowering drug A given to a diabetic patient. Following the prescription of drug A, some other conditions could occur to this patient since diabetes can lead to various comorbidities (AACE). To treat a newly introduced condition, drug B is prescribed to the patient. In this case, if we again consider only the marginal correlation between drug B and FBG level, one might draw the conclusion that drug B could lower FBG level since after the prescription of drug A, the FBG level has decreased.

In the two aforementioned confounding issues, drug B is called an *innocent bystander*. Like multiple SCCS (Simpson et al., 2013), multiple CSCCS can effectively handle the innocent bystander confounding problem (a.k.a. Simpson's Paradox). This is because the confounder seems to spuriously correlated to the FBG level when we consider their marginal correlation. However, using a multiple linear model like CSCCS, the joint exposure statuses of both drug A and drug B can be considered simultaneously. Therefore, CSCCS might be able to identify that the decrease of FBG level occurs only when conditioning on the exposure of drug A and hence rule out drug B in the model.

In terms of addressing various confounding issues, CSCCS inherits most of the strengths and weaknesses from SCCS, due to the close relationship between the two models. While CSCCS might address reasonably well the innocent bystander confounding problem, it might not be well suited to handle confounding issues such as time-varying confounding (Daniel et al., 2013). In Section 3.5, we empirically evaluate the performance of CSCCS in the CDR task and illustrate how its performance is related to its capabilities of addressing various confounding issues.

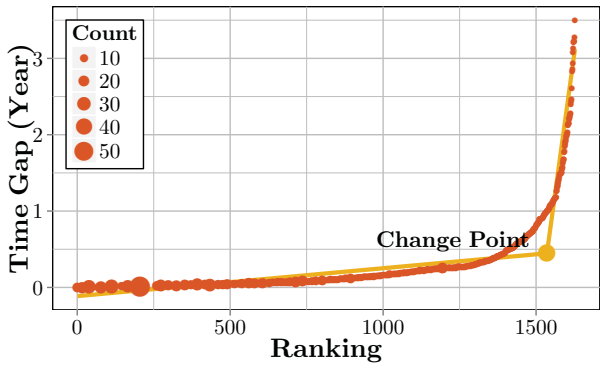


Figure 3.2: Time gap of Humalog in ascending order: the size of dots represents the number of time gaps that share the same value.

3.4 Building Drug Eras from Drug Prescription Records

A prerequisite of CSCCS is the availability of drug era information of each drug prescribed to each patient. However, drug era information is usually not provided in most EHRs. Instead, drug prescription records of each patient are kept, usually with the name (or id) of the drug and the date of prescription. Constructing drug eras from drug prescription records is an important but challenging task for both CDR using CSCCS and ADR discovery.

3.4.1 Drug Era in Common Data Model

A heuristic proposed in the Common Data Model (CDM) (Reisinger et al., 2010) by Observational Medical Outcome Partnership (OMOP) is to first consider the prescription dates of each prescription record as the start date of the drug era. It then assumes that each drug era lasts n days and hence computes the end date of the drug era accordingly. Within the same patient, we assume there is only one drug prescription record of the same drug in a given date. In this way, drug eras of the same drug within each patient constructed as before start from different dates. For an adjacent pair of drug eras of the same drug within the same patient, we call

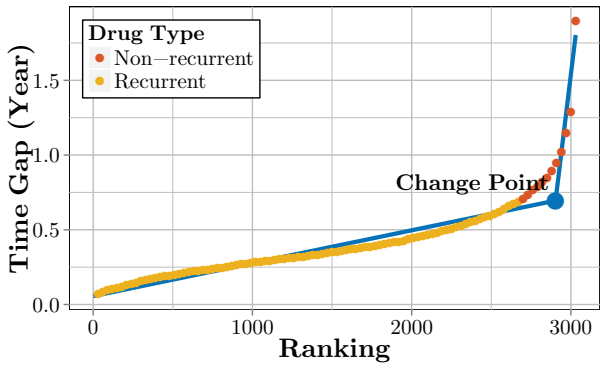


Figure 3.3: Change points of all drugs in the EHRs in ascending order.

the drug era that starts earlier a *former era*, and the other a *latter era*. CDM defines a parameter called *persistence window*. If the start date of the latter era, subtracted by the end date of the former era, is no larger than the persistence window, CDM merges the two drug eras into one, using the start date of the former era as the start date of the new era and the end date of the latter era as the end date of the new era. CDM tries to merge as many drug eras of the same drug within the same patient as possible in this fashion, until every resultant drug era of the same drug within the same patient is separated by more than persistence window amount of time. In CDM, both n and the persistence window are usually set to thirty days.

The intuition behind this heuristic is to build a longer drug era if the prescription date of an adjacent pair of records of the same drug are close enough to each other. A natural question to ask is *how large* the time gap between the two adjacent prescription records can be for us to still consider them close enough?

3.4.2 Constructing Drug Eras via Change Point Analysis

Instead of specifying a predetermined threshold on time gap as it is in CDM, we answer this question via a data-driven approach: for each drug, we compute the time gaps between all adjacent pairs of prescription records. We then sort these time gaps in ascending order. A visualization of the values of the time gaps of Humalog against their relative rankings is given in Figure 3.2. From Figure 3.2,

we notice that the distribution of time gaps can be approximated by a piecewise linear model with a change point close to the end of the sample with large time gap values. The smaller time gaps can be fitted well by the flat linear segment of the model while the larger time gaps can be fitted well by the steep linear segment. This phenomenon leads to a reasonable assumption that the smaller time gaps are sampling from a different underlying distribution than that of the larger time gaps. The smaller time gaps sampling from the same distribution correspond to the adjacent pairs of prescription records that we can consider close enough to each other to construct a lasting drug era. A threshold we can use to distinguish the two types of time gaps is the change point of the piecewise linear model.

For each drug with at least fifty prescription records in the EHRs, we perform change point detection analysis in the aforementioned fashion using `R` package `segmented`. We plot the change points of all the drugs against their relative rankings after sorting them in ascending order in Figure 3.3. Interestingly, there is also a change point in Figure 3.3. A possible explanation of the existence of a change point in Figure 3.3 is that in EHR data, drug prescriptions of some particular drugs are recurrent in order to battle chronic disease. For example, a diabetic patient needs long-term prescriptions of some FBG lowering drugs. On the other hand, the prescriptions of some other drugs are non-recurrent, such as antibiotics. We consider the change point in Figure 3.3 as a threshold to distinguish recurrent drugs from non-recurrent drugs in the EHR because a reasonable expectation is that if a drug is recurrent, the gap between an adjacent pair of prescription records of that drug from the same patient will tend not to be too large and hopefully under the change point specified in Figure 3.3.

We extend the heuristic provided in CDM as follow: We first denote the mean of all change point values of the recurrent drugs in the EHR as γ. For all the recurrent drugs, we set their corresponding $\mathfrak{n}$'s and the value of their persistence windows to $\frac{\gamma}{2}$. We then set $\mathfrak{n} = 0.04$year (approximately two weeks) for all non-recurrent drugs and 0 as the value of their persistence windows.

3.5 Experiments

As far as we know, our CSCCS model is the first of its kind to explicitly use temporal ordering information in EHRs for CDR. How do we evaluate the performance of a method that utilizes this type of information? As a preliminary endeavor, we try to answer this question by addressing two major challenges for our experiments.

3.5.1 Lack of a Baseline Method

The first challenge we need to handle is the lack of a baseline method that also utilizes temporal ordering information in an EHR for CDR. Inspired by the idea of disproportionality analysis from the pharmacovigilance literature (Montastruc et al., 2011), we propose the *pairwise mean* (PM) method as a baseline method. PM assigns a real-valued score to each of the M drugs in the EHR to represent how likely the drug decreases FBG level, and a smaller score implies a stronger decreasing tendency. The score of the m^{th} drug, s_m, is computed as follow: first, for the i^{th} patient who has FBG measurements within two years before *and* after the *first* prescription of the m^{th} drug, we compute the mean of those FBG measurements before and after the first prescription, denoted as b_{mi} and a_{mi}, respectively; second, compute s_m as:

$$s_m = \frac{1}{N_m} \sum_{i=1}^{N_m} (a_{mi} - b_{mi}),$$

where N_m is the number of patients that have FBG measurements two years before and after the first prescription of the m^{th} drug.

3.5.2 Incomplete Ground Truth

Unlike the task of ADR discovery from the EHR, where numerous research efforts have been invested on developing a set of ground truth (OMOP, 2015) drug-adverse-reaction pairs so that algorithms can be run and evaluated, we do not have access to such a ground truth set for the task of CDR from EHRs. We use Marshfield Clinic EHR as our data source and there are about two thousand drugs for evaluation. To

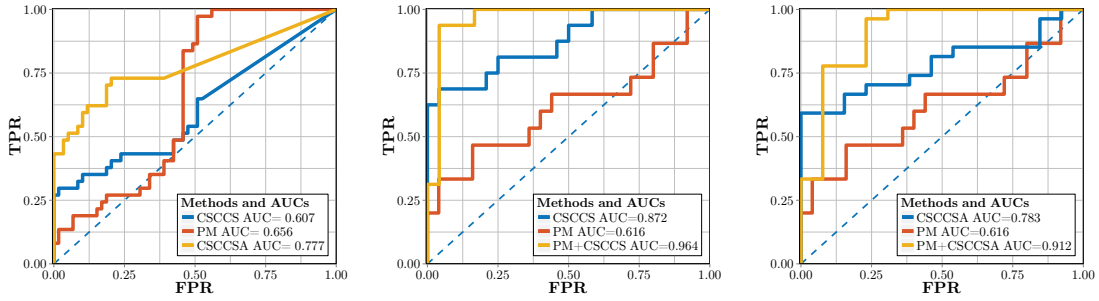
Table 3.1: A summary of three types of drugs discovered by the three algorithms

	PM	CSCCS	CSCCSA
decrease	15	16	27
increase	1	1	0
potential	24	23	13

evaluate the performance of our algorithm without knowing the glucose altering effect of every drug, we focus on the top forty most promising drugs generated by PM, CSCCS, and CSCCSA, as shown in Table 3.2, Table 3.3, and Table 3.4, respectively.

In these three tables, rows that are shaded in green represent the drugs commonly prescribed for lowering glucose while rows that are shaded in red represent the drugs commonly prescribed for increasing glucose. The two types of drugs in the three tables are all manually labeled. Drugs in the unshaded rows might potentially be irrelevant, or might constitute new discoveries. These drugs are discussed in further detail in Section 3.5.7. A summary of the number of each of the three types of drugs discovered by the three algorithms are given in Table 3.1.

In CSCCSA, we set τ defined in Section 3.3.2 to four years. In Table 3.2, the counts and scores are N_m's and s_m's defined in Section 3.5.1, while in Table 3.3 and Table 3.4, the counts are the L_1 norm of the columns in $\mathbf{X}$ corresponding to different drugs, and the scores are the regression coefficients of different drugs. We only consider drugs with counts greater than or equal to eight. For CSCCS and CSCCSA, we first construct drug eras using the method described in Section 3.4, where we determine that $\gamma = 0.34$ years. We then use a lasso penalty for variable selection to generate a long list of about two hundred drugs, and we present the top forty among those selected drugs as the short list. The number eight and forty could be tuned to optimize accuracy but were fixed here beforehand for practical reasons. Drugs with fewer than eight prescriptions might not have sufficient evidence to support a new use. Evaluating more than forty results per method was too large a burden for human literature review.



(a) PM, CSCCS, and CSCCSA (b) CSCCS (c) CSCCSA

Figure 3.4: ROC curves of different methods evaluated using different subsets of ground truth given in Table 3.2, Table 3.3, and Table 3.4.

3.5.3 Dataset

EHRs of 64515 patients from Marshfield Clinic are used in the CSCCS and CSCCSA experiments, providing 219306 FBG measurement records and 2980 drug candidates.

3.5.4 Receiver Operating Characteristic

As shown in Tables 3.2–3.4, all three methods capture a reasonable number of drugs that are prescribed for lowering glucose among their top forty candidates. We therefore consider identifying drugs prescribed for glucose-lowering as a binary classification task and use receiver operating characteristic (ROC) curves as well as area under ROC (AUROC) to evaluate the performance of each algorithm.

We first construct the ROC curves of the three methods using the union list of drugs from Tables 3.2–3.4. The three ROC curves are presented in Figure 3.4a. Since we perform variable selection in CSCCS and CSCCSA, some drugs might be assigned scores of zero and hence are considered irrelevant to the prediction of FBG level. In these cases, we put these drugs at the bottom of the union list and consider them to be identified as positive examples by the algorithms only at the very end. This results in the straight line segment of the ROC curves of CSCCS and CSCCSA at the liberal region. Figure 3.4a shows that CSCCSA has the highest

AUC, outperforming CSCCS and PM by a significant margin, while PM and CSCCS have similar AUCs. However, in the more conservative region where there is drug support for all three methods, CSCCS outperforms PM while CSCCSA maintains the best performance. This phenomenon suggests that the modeling assumptions of CSCCS and CSCCSA are able to provide insights into making reasonable prediction of FBG level.

Figure 3.4b uses the forty drugs in Tables 3.2 and 3.3 to generate the ROC curves, in red for PM and in blue for CSCCS. As a comparison, we also plot the ROC curve of the following ensemble strategy: we first use the top forty drugs in Table 3.3 as a result of variable selection via CSCCS, then we compute the PM scores over the selected drugs. By comparing the AUCs of the three curves, we notice that the ensemble method outperforms CSCCS and PM, while CSCCS outperforms PM. Since the scores used to construct the CSCCS ROC curve are regression coefficients of drug exposure statuses under a lasso penalty, the lack of an oracle property for the lasso (Wainwright, 2009a) might potentially trade off the inherent order among drugs for a sparse model. However, such a trade-off is arguably beneficial, based on the significant improvement of AUC of CSCCS compared with the AUC of PM.

Figure 3.4c is generated similarly as Figure 3.4b. The ensemble of CSCCSA with PM outperforms the two individual algorithms. Although the AUC of CSCCSA is less than that of CSCCS, it is worthy to notice that all but one true positive drugs in Table 3.3 are discovered in Table 3.4 at the top fifteen positions. Other than that, CSCCSA is also able to discover twelve more true positives that CSCCS does not capture among its top forty discoveries.

3.5.5 Precision at K

The task of CDR from EHRs is somewhat analogous to web search. Specifically, the algorithm should select only a few drugs that have interesting unexpected effects on the response: returning too many results makes it infeasible for human experts to evaluate the potential effect of the selected drugs. This is similar to users performing web search on a search engine, where typically only the quality of the

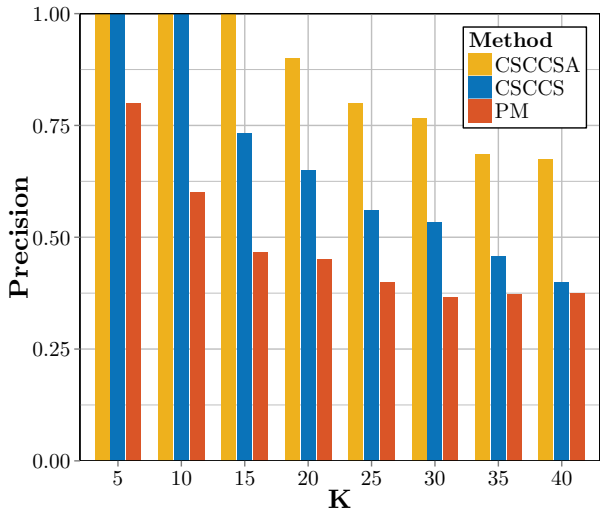


Figure 3.5: Precision at K of PM, CSCCS, and CSCCSA.

results on the first page, or the first K results, matters. Based on this observation, an algorithm with a high precision-at-K value is desirable. Figure 3.5 shows the precision of each of the three algorithms at different positions (K) in the task of identifying drugs prescribed for lowering glucose. CSCCSA achieves the highest performance at all positions. CSCCS outperforms PM significantly at smaller K's, but the performances of the two algorithms are similar at larger K's. This is consistent with results in Table 3.1, showing that CSCCSA is able to identify more prescribed drugs for lowering glucose than the other two methods. Moreover, these drugs are at the very top of Table 3.4. Therefore, precision-at-K provides evidence for CSCCSA's utility for CDR from EHRs.

3.5.6 Drugs with Known Glucose Increasing/ Decreasing Effects

From Tables 3.2–3.4, we notice that CSCCSA discovers the most number of drugs prescribed for lowering glucose among the three methods under consideration. This reaffirms our belief that CSCCSA is a promising method for CDR from EHR. Furthermore, we also notice that drugs prescribed for increasing glucose are reported in all but the table of CSCCSA.

In Table 3.2, sucrose is observed as a false positive using PM. Based on its count, this might be a spurious correlation in the data. This is even more probable when we consider the fact that the effect of sucrose on blood glucose level is short-term, and sucrose is not a drug that consistently enter patients' EHR for a long period of time. However, PM considers the glucose measurement records of the patients within two years before and after the first prescription of sucrose, during which many stronger confounding factors could have occurred to alter the glucose level.

In Table 3.3, glucagon is identified. Glucagon is given to diabetic patients that take glucose-lowering drugs to avoid hypoglycemia. However, glucagon alone is not frequently administered. Therefore, in the data, we observe the co-occurrence of glucagon with various glucose-lowering drugs. While glucagon alone increase blood glucose, combining with glucose-lowering drugs usually results in the decrease of blood sugar. On the other hand, we did not have enough data where glucagon is prescribed alone to observe the responses. Therefore, the algorithm will consider glucagon to have glucose-lowering effects since most of the time the occurrence of glucagon is accompanied by blood sugar decreasing medications. The algorithm might even consider it as a strong glucose-lowering drug because the actual glucose-lowering drugs are coded in various names in the EHR, hence dispersing the effect, while glucagon is coded only by a few different names.

3.5.7 Confounding and Potential Drugs

We now turn to the discussion of the drugs discovered by the three algorithms in Table 3.2–3.4 that are not prescribed for glucose increasing/decreasing. We will make use of a list providing drugs that can influence blood glucose level (DiabetesInControl, 2015) to aid our evaluation process.

The Blessing and the Curse of Marginal Correlations

We discuss the results in Table 3.2. According to DiabetesInControl (2015), Actigall can cause blood glucose level to increase while amphotericin B can cause blood glucose level to decrease. An interesting drug that is also brought to our attention

Table 3.2: Top forty drugs: PM-Glucose.

INDX	CODE	DRUG NAME	SCORE	COUNT
1	5226	LANTUS	-41.672	34
2	6646	NOVOFINE 31	-38.709	33
3	5789	METFORMIN HYDROCHLORIDE	-38.623	10
4	5806	METHENAM/MBLU/BA/SAL/ATROP/HYO	-36.710	10
5	4811	INSULIN NPH	-34.573	23
6	6652	NOVOLOG	-29.895	54
7	4336	HABITROL	-29.871	16
8	6044	MONISTAT	-29.721	14
9	9080	SURFAK	-29.655	14
10	9155	SYRNG W-NDL DISP INSUL 0.333ML	-29.439	30
11	4500	HUMULIN	-29.186	36
12	9008	SUGAR SUBSTITUTE	-28.971	10
13	10176	VORICONAZOLE	-28.538	10
14	1305	BUDEPRION SR	-27.444	9
15	8450	ROXICODONE	-27.428	12
16	9534	TRANDATE	-25.978	8
17	4802	INSULIN	-24.507	697
18	3849	FLURBIPROFEN SODIUM	-24.403	11
19	8316	REZULIN	-24.287	135
20	5257	LENALIDOMIDE	-22.875	8
21	4485	HUMALOG	-22.852	67
22	1389	CAL	-22.817	61
23	144	ACTIGALL	-22.237	36
24	8998	SUCROSE	-22.125	18
25	3843	FLUPHENAZINE HCL	-22.094	8
26	3682	FERROUS FUMARATE	-21.225	10
27	9104	SYMLINPEN 120	-20.333	12
28	1868	CHLORAMBUCIL	-20.268	14
29	4171	GLUCOTROL XL	-19.719	828
30	504	AMPHOTERICIN B	-19.672	24
31	3778	FLEXOR	-19.287	14
32	8241	REGULAR INSULIN	-19.205	39
33	824	AVANDIA	-19.140	487
34	5783	METAPROTERENOL	-18.920	10
35	4434	HIBICLENS	-18.863	10
36	5815	METH/ME BLUE/BA/PHENY/ATP/HYOS	-18.727	11
37	5010	JANUVIA	-18.716	11
38	4813	INSULIN NPL/INSULIN LISPRO	-18.515	126
39	4595	HYDROMORPHONE	-18.470	17
40	7626	POLYMYXIN B SULFATE MICRONIZED	-18.456	11

Table 3.3: Top forty drugs: CSCCS-Glucose.

INDX	CODE	DRUG NAME	SCORE	COUNT
1	7470	PIOGLITAZONE HCL	-13.502	3075
2	8437	ROSIGLITAZONE MALEATE	-13.465	1019
3	6656	NPH HUMAN INSULIN ISOPHANE	-10.963	2874
4	4497	HUM INSULIN NPH/REG INSULIN HM	-10.869	1829
5	160	ACTOS	-7.665	1125
6	824	AVANDIA	-7.543	1239
7	4837	INSULN ASP PRT/INSULIN ASPART	-7.067	258
8	4806	INSULIN GLARGINE HUM.REC.ANLOG	-5.571	4213
9	9152	SYRING W-NDL DISP INSUL 0.5ML	-5.301	4186
10	8316	REZULIN	-3.611	444
11	3227	ENALAPRIL	-3.218	1103
12	6382	NEEDLES INSULIN DISPOSABLE	-3.148	2827
13	4970	ISOSORBIDE DINITRATE	-3.122	1220
14	9623	TRICOR	-3.119	821
15	3686	FERROUS SULFATE	-2.898	4820
16	1760	CELEXA	-2.887	1473
17	4802	INSULIN	-2.806	1526
18	4118	GLUCAGON HUMAN RECOMBINANT	-2.722	1639
19	5786	METFORMIN	-2.625	3838
20	7731	PRAVACHOL	-2.458	1700
21	2512	DARBEPOETIN ALFA IN ALBUMN SOL	-2.359	426
22	6210	MYCOPHENOLATE MOFETIL	-2.253	724
23	2830	DILTIAZEM	-2.216	1021
24	5636	MAVIK	-2.150	2242
25	4132	GLUCOPHAGE	-2.133	6736
26	4525	HYDRALAZINE HCL	-2.095	792
27	4106	GLIMEPIRIDE	-2.034	3384
28	7129	PAXIL	-2.033	2021
29	2426	CYANOCOBALAMIN (VITAMIN B-12)	-1.992	4080
30	4833	INSULIN ZINC HUMAN REC	-1.945	116
31	10392	ZOLOFT	-1.926	2417
32	6069	MORPHINE SULFATE	-1.889	899
33	10333	ZESTRIL	-1.787	2032
34	1216	BLOOD SUGAR DIAGNOSTIC	-1.665	19832
35	10199	WARFARIN SODIUM	-1.632	9223
36	3937	FOSINOPRIL SODIUM	-1.540	2660
37	6499	NIFEDIPINE	-1.524	1472
38	1003	BENAZEPRIL HCL	-1.462	1586
39	9994	VERAPAMIL HCL	-1.433	1856
40	1573	CAPTOPRIL	-1.418	1989

Table 3.4: Top forty drugs: CSCCSA-Glucose.

INDX	CODE	DRUG NAME	SCORE	COUNT
1	4485	HUMALOG	-11.786	124
2	7470	PIOGLITAZONE HCL	-10.220	3075
3	8437	ROSIGLITAZONE MALEATE	-9.731	1019
4	4837	INSULN ASP PRT/INSULIN ASPART	-9.658	258
5	6382	NEEDLES INSULIN DISPOSABLE	-9.464	2827
6	4171	GLUCOTROL XL	-8.117	2853
7	4106	GLIMEPIRIDE	-7.940	3384
8	160	ACTOS	-7.721	1125
9	824	AVANDIA	-6.802	1239
10	9152	SYRING W-NDL DISP INSUL 0.5ML	-6.623	4186
11	4132	GLUCOPHAGE	-6.322	6736
12	4184	GLYBURIDE	-6.021	8879
13	4170	GLUCOTROL	-5.721	1259
14	4208	GLYNASE	-5.670	591
15	416	AMARYL	-5.599	2240
16	4107	GLIPIZIDE	-5.563	9993
17	844	AXID	-4.682	189
18	2830	DILTIAZEM	-4.297	1021
19	4806	INSULIN GLARGINE HUM.REC.ANLOG	-4.175	4213
20	5787	METFORMIN HCL	-4.147	19584
21	2824	DILAUDID	-4.076	39
22	5786	METFORMIN	-3.890	3838
23	7731	PRAVACHOL	-3.532	1700
24	1760	CELEXA	-3.517	1473
25	4497	HUM INSULIN NPH/REG INSULIN HM	-3.501	1829
26	9889	URSODIOL	-3.132	376
27	4813	INSULIN NPL/INSULIN LISPRO	-2.972	623
28	4133	GLUCOPHAGE XR	-2.845	765
29	6445	NEURONTIN	-2.615	1418
30	6656	NPH HUMAN INSULIN ISOPHANE	-2.500	2874
31	9379	THIAMINE HCL	-2.383	341
32	1636	CARDURA	-2.198	1079
33	1218	BLOOD SUGAR DIAGNOSTIC DRUM	-2.073	2593
34	8025	PROZAC	-2.037	1525
35	8316	REZULIN	-1.895	444
36	9136	SYRINGE & NEEDLE INSULIN 1 ML	-1.885	3542
37	4802	INSULIN	-1.812	1526
38	7674	POTASSIUM CHLORIDE	-1.779	9842
39	4804	INSULIN ASPART	-1.752	2476
40	1200	BLOOD-GLUCOSE METER	-1.719	5289

Table 3.5: Top forty drugs: CSCCSA-LDL.

INDX	CODE	DRUG NAME	SCORE	COUNT
1	8444	ROSUVASTATIN CALCIUM	-17.052	27122
2	5368	LIPITOR	-16.908	118468
3	2395	CRESTOR	-16.234	3535
4	8720	SIMVASTATIN	-15.790	206064
5	3584	EZETIMIBE/SIMVASTATIN	-14.721	19396
6	790	ATORVASTATIN CALCIUM	-13.982	151106
7	941	BAYCOL	-12.924	1236
8	10383	ZOCOR	-11.451	26514
9	10186	VYTORIN	-9.877	9047
10	5487	LOVASTATIN	-9.238	45286
11	3583	EZETIMIBE	-8.093	32595
12	7731	PRAVACHOL	-6.729	16525
13	10336	ZETIA	-6.678	6623
14	7733	PRAVASTATIN SODIUM	-6.638	33708
15	5261	LESCOL XL	-6.358	873
16	9183	TAMOXIFEN CITRATE	-4.777	3095
17	5893	MEVACOR	-4.172	4205
18	2175	COLACE	-4.016	4349
19	9182	TAMOXIFEN	-3.764	2048
20	5260	LESCOL	-3.716	6251
21	475	AMLODIPINE/ATORVASTATIN	-2.779	1272
22	494	AMOXICILLIN/POTASSIUM CLAV	-2.495	4186
23	2110	CLOPIDOGREL BISULFATE	-2.271	50059
24	4616	HYDROXYCHLOROQUINE SULFATE	-2.240	5888
25	5281	LEVAQUIN	-2.194	1464
26	3471	ESTROGEN CON/M-PROGEST ACET	-1.929	5896
27	7496	PLAVIX	-1.471	14220
28	8225	RED YEAST RICE	-1.345	5468
29	3746	FLAGYL	-1.169	278
30	6540	NITROGLYCERIN	-1.103	94747
31	2959	DOCUSATE SODIUM	-1.084	32872
32	3475	ESTROGENS CONJUGATED	-1.033	22480
33	3686	FERROUS SULFATE	-0.990	32496
34	7768	PREMARIN	-0.969	5513
35	865	AZITHROMYCIN	-0.959	9861
36	2811	DIGOXIN	-0.908	31353
37	4132	GLUCOPHAGE	-0.779	14764
38	493	AMOXICILLIN	-0.715	11214
39	1985	CIPROFLOXACIN	-0.651	989
40	9946	VARENICLINE TARTRATE	-0.636	10794

is buderprion SR. Buderprion SR is an antidepressant prescribed for the treatment of depressive disorder. For diabetic patients with depression, buderprion SR can help to alleviate their depressive symptom, making them in a better mood. This in turn has a positive effect on better controlling blood glucose level for longer period of time (Lustman et al., 2007). PM is able to discover the blood glucose lowering effect of buderprion SR, even with a mere support of nine patients. The fact that PM considers the marginal correlation of each drug-indication pair independently makes it more likely to discover interesting drug-indication pairs with a weaker support. However, spurious correlations, especially those caused by the innocent bystander problem, are also more likely to be reported this way.

Comparing the results from Table 3.2 with those from Table 3.3 and Table 3.4 could justify our argument. In Table 3.2, Habitrol is a nicotine patch, and Monistat, Voriconazole, amphotericin B, and Hibiclens are all used to treat fungal infection. Interestingly, fungal infection is a comorbidity of diabetes (Vazquez and Sobel, 1995; ADA), and smokers are also more inclined to be diabetic (CDC). On the other hand, we cannot find any drugs that are related to fungal infection or quitting smoking in Table 3.3 and Table 3.4. This comparison suggests that the aforementioned drugs in Table 3.2 generated by marginal association methods like PM might be innocent bystanders while a multiple regression approach such as CSCCS and CSCCSA might significantly help to alleviate this type of confounding issue.

Potential drugs found by CSCCS and CSCCSA

Results in Table 3.3 are as follows. A study (Vermes et al., 2003) indicates that enalapril helps to decrease the occurrence rate of diabetes in patients with chronic heart failure. Tricor might also have the potential to lower blood sugar level, based on the findings in Damci et al. (2003) and Balakumar et al. (2014). Vitamin B12 is another interesting drug for consideration. In a rat model used by Chow and Stone (1957), deficiency in vitamin B12 is linked to hyperglycemia. However, blood glucose level can be decreased by providing vitamin B12. A recent study suggests that diabetic patients under metformin might experience vitamin B12 deficiency (Ting

et al., 2006). In a study on depressive patients, Zoloft, which is an antidepressant, is linked to the increase of insulin level after its prescription (Kesim et al., 2011). Zestril, which is the brand name of lisinopril, is found to inhibit high blood sugar level in rats (Balakumar et al., 2014). Captopril is also reported to improve daily glucose profile among non-insulin-dependent patients (Kodama et al., 1990). However, hydralazine HCl is linked to glucose-increasing in a rat model, according to the findings in Satoh et al. (1980). Nifedipine, verapamil HCl, and morphine sulfate can decrease blood sugar while captopril interacting with hydrochlorothiazide could cause high blood sugar, according to the list in DiabetesInControl (2015). The potential glucose-lowering drugs discovered indicate that CSCCS is a reasonable method for the task of CDR.

Results in Table 3.4 are as follows. Pravachol is a member of a popular class of drugs called statins which are prescribed to lower cholesterol level. Although the Food and Drug Administration (FDA) has added blood-glucose-increase warnings to all the drugs in the statin class (FDA, 2014), Pravachol itself has been considered to have blood-glucose lowering effects (Freeman et al., 2001; Carter et al., 2013). The fact that CSCCSA can single out this particular drug from other statin class drug members indicates the potential of the algorithm to distinguish among similar drugs that have subtle differences. Celexa has a mild but non-significant effect on FBG level reduction in a study with seventeen depressive patients (Amsterdam et al., 2006). Several cases of hypoglycemia linked to the use of Neurontin have also been reported (Scholl et al., 2015). Thiamine is reported to reduce the adverse effect of hyperglycemia by inhibiting certain biological pathways (vinh quoc Luong and Nguyen, 2012) and deficiency of thiamine is observed in diabetic patients (Page et al., 2011). Cardura is found to reduce insulin resilience in a study on hypertensive patients with diabetes (Inukai et al., 2004). Prozac can cause high or lower blood sugar while diltiazem is linked to low blood glucose level (DiabetesInControl, 2015).

3.5.8 Experiments on Low-density Lipoprotein

To demonstrate the potential of our methodology, we also apply our method to predict the numeric value of low-density lipoprotein (LDL). We first construct drug eras from drug prescription records with the approach proposed in Section 3.4, where γ is computed as 0.36 years. We then run CSCCSA and generate a long list of about two hundred drugs. We report the top forty drugs from the list in Table 3.5. No confirmed false positives are discovered in the table while all the confirmed true positives are reported at the very top of the list. Some entries of hormone are discovered, which are linked to the decrease of LDL in drug/laboratory tests (FDA, c). Interestingly, many entries of antibiotics are discovered, and all of them are classified as non-recurrent drugs by the algorithm in Section 3.4. This is consistent with the clinical practice that antibiotics are usually not prescribed for long-term use. Some antibiotics have also been considered to manage cholesterol level, with literature support dating back to the 1950's (Samuel, 1979; Kesäniemi and Grundy, 1984; Jenkins et al., 2005). The experimental results on LDL suggest that our algorithm is not fine-tuned to boost the performance on discovering drugs that control FBG level. Instead, it is readily applicable to other important numeric clinical measurements that might lead to interesting discoveries in drug repositioning.

3.6 Discussion

We have introduced the CSCCS model for the task of CDR using EHRs. To the best of our knowledge, the proposed model is the first of its kind to extensively leverage temporal ordering information from EHR to predict indications for multiple drugs at the same time. The CSCCS model extends the SCCS model that is popular in the ADR community to address a continuous response. As an initial effort, we evaluate our methodology on the task of discovering potential blood-sugar-lowering indications for a variety of drugs in a real world EHR. We develop a set of experimental evaluation methods specific to this problem in order to estimate the performance of our method. Our experimental results suggest that CSCCS can not

only discover existing indications but is also able to identify potentially new use of drugs. We hence believe that CSCCS is a promising model to aid the knowledge discovery process in CDR.

Future applications and extensions of the CSCCS model are exciting. To begin with, CSCCS can be applied to a broad variety of numeric responses such as blood pressure level, cholesterol level, or body weight, to name a few. Therefore, potentially new indications of drugs to control the aforementioned important physical measurements can be examined in the same paradigm. Furthermore, many other sources of patient information, such as demographic information, diagnosis codes, other type of lab measurements, as well as interactions among all these information sources can be taken into consideration to facilitate the prediction of the physical measurement level.

4 PHARMACOVIGILANCE VIA BASELINE REGULARIZATION WITH LARGE-SCALE LONGITUDINAL OBSERVATIONAL DATA

4.1 Introduction

Pharmacovigilance (PhV, Harpaz et al. 2012, 2015) is the science and activities relating to the surveillance and prevention of adverse events caused by pharmaceutical products *after* they are introduced to the market. While CDR looks for beneficial side effects of drugs, pharmacovigilance looks for adverse drug effects. In addition, often CDR examines continuous responses such as the lab measurements in Chapter 3 and 6, while pharmacovigilance examines binary responses such as "bleeding" or "heart attack", though this distinction is weaker. Thus we have a synergy suggesting that many of the same ideas and even tools can be used for both tasks.

In response to several recent prominent public health hazards (Suchard et al., 2013a) due to adverse drug reactions (ADRs), governments, industries, and other stakeholders across the world have been building effective PhV systems to safeguard admissible profit-risk profiles of drug products on the market. Major PhV systems (Hripcsak et al., 2015; Robb et al., 2012; Findlay, 2015) nowadays leverage a network of large-scale longitudinal observational databases (LODs) (Harpaz et al., 2015) such as electronic health records (EHRs) and medical insurance claim databases that contain individual-level time-stamped rich medical data collected globally from hundreds of millions of individuals. All the databases within the network are updated periodically and are converted to the same format; various ADR discovery algorithms can hence be run regularly on different databases without any modifications to achieve proactive drug safety surveillance.

An efficient algorithm that can deliver accurate ADR identification (high causal fidelity) using LODs is hence of utmost importance to the performance of PhV systems. A leading algorithm is the Multiple Self-Controlled Case Series (MSCCS) method (Simpson et al., 2013). Using the occurrence of a condition of interest from

different patients at different times as the response variable, and the corresponding exposure statuses of various drugs as the features, MSCCS is a parsimonious representation of a fixed effect Poisson regression model (Xu et al., 2012). In MSCCS, each patient acts as his or her own control, during exposed (case) or unexposed (control) periods of time, thus controlling even for latent and unconsidered factors, provided they are *time-invariant*.

However, due to the longitudinal nature of the data, simply adjusting for time-invariant confounding does not suffice to deliver accurate modeling. For example, the occurrence rate of adverse events such as heart attacks usually increases as the observed individual ages. Moreover, patients that previously had heart attacks will also be prone to have another one in the future. Neither of the aforementioned *time-inhomogeneous* occurrence rates of heart attack can be modeled by adjusting for time-invariant confounding via MSCCS.

By assuming an individual-specific, time-dependent occurrence rate of adverse events, the mission of the proposed Baseline Regularization (BR) method is to provide flexibility to model the temporal inhomogeneity nature of LODs, in the hope of delivering more effective ADR discovery with high causal fidelity. Our contributions are three-fold:

- BR is the first general-purpose ADR discovery algorithm following a self-controlled design that exploits the time-inhomogeneous perspective of individual profiles in large-scale LODs.

- BR is deeply connected to and is a generalization of some of the existing models in the literature. BR not only directly generalizes MSCCS, it is also a generalized linear model that extends (Kuang et al., 2016a), which deals with baseline regularization in a linear model setting.

- Experimental results suggest that incorporating the heterogeneity among different patients and different times help to improve the performance in identifying *benchmark* ADRs from the Observational Medical Outcomes Partnership ground truth (Simpson, 2011).

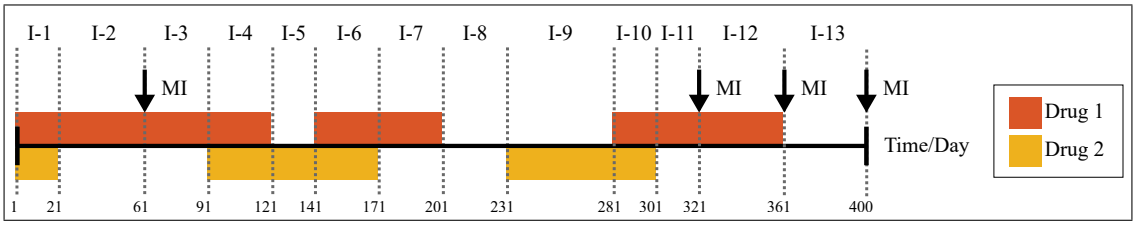


Figure 4.1: Visualization of a patient's EHR. MI: Myocardial Infarction (heart attack). See the beginning of 4.2.1 for more descriptions.

4.2 Model Specification

4.2.1 Background

Figure 4.1 visualizes the EHR from a patient that has taken two drugs and has had four heart attacks throughout his 400 days of observation. The rectangular bands in different colors represent different *drug eras*, each representing a consecutive time period during which the patient was exposed to a particular drug. A drug era is recorded with its *start date*, *end date*, and the name of the drug. The black arrows pointing downwards annotated with MI (Myocardial Infarction) represent the date on which the patient had a heart attack. The gray dashed lines and the indices on the top of the figure represent different *intervals*, a concept that we will define later in Section 4.2.3. In this chapter, we consider the *multiple-drug, single-ADR* setting. As an illustrative example, our task of using the EHR from the patient presented in Figure 4.1 and from *many* other patients is to determine whether the exposure to certain drugs might cause the occurrence of MI as an adverse event.

Suppose there are M drugs and N patients in the EHR database. We use J_i to represent the total number of days of observation available in the EHR of patient i, where $i \in \{1, 2, \cdots, N\}$. We use χ_{ijm} to represent a binary drug exposure status of drug m on the j^{th} day during the observation of the i^{th} patient, where $j \in \{1, 2, \cdots, J_i\}$, and $m \in \{1, 2, \cdots, M\}$. $\chi_{ijm} = 1$ represents exposure and $\chi_{ijm} = 0$ represents non-exposure. We further use y_{ij} to represent a binary MI occurrence variable with $y_{ij} = 1$ meaning that the i^{th} patient has an MI on the j^{th} day during the observation, and $y_{ij} = 0$ otherwise. With the notation introduced above, we

can consider y_{ij}'s as a response variable and χ_{ijm}'s as features. Following the convention of MSCCS, we will use a Poisson regression model (instead of a logistic regression model even though the response is binary) to depict the relationship between the response variable and the features, resulting in the following log-likelihood function:

$$\log \mathcal{L}(\boldsymbol{\tau}, \boldsymbol{\beta}) = \sum_{i=1}^{N} \sum_{j=1}^{J_i} y_{ij} \left(\tau_{ij} + \boldsymbol{\chi}_{ij}^{\top} \boldsymbol{\beta}\right) - \exp\left(\tau_{ij} + \boldsymbol{\chi}_{ij}^{\top} \boldsymbol{\beta}\right), \tag{4.1}$$

where

$$\boldsymbol{\chi}_{ij} = \begin{bmatrix} \chi_{ij1} & \chi_{ij2} & \cdots & \chi_{ijM} \end{bmatrix}^{\top}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_M \end{bmatrix}^{\top},$$
$$\boldsymbol{\tau} = \begin{bmatrix} \tau_{11} & \tau_{12} & \cdots & \tau_{1J_1} & \cdots & \tau_{N1} & \tau_{N2} & \cdots & \tau_{NJ_N} \end{bmatrix}^{\top}.$$

The occurrence rate of MI to the i^{th} patient on the j^{th} day during observation is hence given by $\exp\left(\tau_{ij} + \boldsymbol{\chi}_{ij}^{\top} \boldsymbol{\beta}\right)$, from which we can infer that the rate is determined by two contributing factors. The first one depends on the joint drug exposure statuses, described by $\boldsymbol{\chi}_{ij}$, and the effect of each drug on the occurrence rate of MI, given by $\boldsymbol{\beta}$. If the value of a particular component of $\boldsymbol{\beta}$ is especially large, then the occurrence rate of MI will increase upon the exposure of the corresponding drug. Therefore, such a drug *might* potentially cause MI as an ADR. The second factor is the *baseline parameter* τ_{ij}, which models the inherent occurrence rate of MI for the i^{th} patient on day j excluding the interference of the effects from other covariates modeled by $\boldsymbol{\beta}$.

4.2.2 Baseline Regularization

Baseline Parameters

The introduction of the baseline parameters τ_{ij}'s in (4.1) is strikingly simple, and yet it offers tremendous flexibility to portray the heterogeneity of adverse event occurrence rates among different patients, and during different time periods within

the same patient.

For example, a person who has High Blood Pressure (HBP) might have an inherently higher risk for heart attack compared with a healthy person. Therefore, the baseline parameters for the HBP patients might be higher compared with those of a healthy person. Within the same individual, commonsense-supported observations in the EHRs often suggest that one should vary baseline parameters temporally: for example, the risk for heart attack tends to increase in general as a person ages; a patient who has a history of heart attack might also be more likely to have another heart attack in the future. In both cases, a set of baseline parameters with increasing tendency along time within the same patient might be introduced to model such observations.

On the other hand, MSCCS makes the following more restrictive modeling assumptions:

$$\tau_{ij} = \alpha_i, \quad \forall i \in \{1, 2, \cdots, N\}, \quad \forall j \in \{1, 2, \cdots, J_i\}.$$

That is, MSCCS assumes that baseline parameters can only differ among different patients. Within the same patient, baseline parameters do not vary across time. While this modeling assumption is reasonable to address for time-invariant confounding such as gender, socioeconomic status, and genetic profile, it easily fails to model the aforementioned time-inhomogeneous occurrence rates.

Regularization

An observant reader might have already noticed that the modeling flexibility introduced by baseline parameters τ_{ij}'s in (4.1) comes with the steep cost of *overparameterization*: the number of baseline parameters introduced is equal to the sample size of the data! Furthermore, in a typical EHR setting there could be thousands of drugs available. Modeling the effects of all these drugs will introduce a $\boldsymbol{\beta}$ whose dimension is easily on the order of thousands. The high dimensionality of both $\boldsymbol{\tau}$ and $\boldsymbol{\beta}$ motivates us to reduce the degrees of freedom of the model via *sparse* regularization, which results in the *baseline regularization* optimization problem as

follows:

$$\arg\min_{\boldsymbol{\tau},\boldsymbol{\beta}} -\log \mathcal{L}(\boldsymbol{\tau},\boldsymbol{\beta}) + \lambda_1 \|\boldsymbol{\beta}\|_1 + \sum_{i=1}^{N}\sum_{j=1}^{J_i-1} \lambda_2 |\tau_{i,j+1} - \tau_{ij}| + \lambda_3 \|\boldsymbol{\tau}\|_2^2 . \tag{4.2}$$

Here in (4.2) we use a lasso penalty to regularize $\boldsymbol{\beta}$ because we assume that among thousands of drugs, there can only be a few that influence the occurrence rate of MI. We use a fused lasso penalty (Tibshirani et al., 2005; Johnson, 2013; Ramdas and Tibshirani, 2015) to regularize $\boldsymbol{\tau}$. The intuition behind using this penalty is that we assume the change between two *adjacent* baseline parameters is steady and gradual, and hence the baseline occurrence rate should not differ much from one day to another between two days that are adjacent to each other.

We also use a ridge penalty to regularize $\boldsymbol{\tau}$. The necessity for including this penalty can be seen from the observations between day 201 and day 230 in Figure 4.1. During this time period (interval I-8), $y_{ij} = 0$, and $\boldsymbol{\chi}_{ij} = \mathbf{0}$, $\forall j \in \{201, 201, \cdots, 230\}$, where for convenience we assume that the patient in Figure 4.1 is indexed by i. Therefore, during this time period, τ_{ij}'s will tend to be very negative in order to drive the occurrence rate $\exp(\tau_{ij})$ to a number that is very close to zero for a maximum likelihood interpretation of the data. In this scenario, a very negative τ_{ij} might overfit the data. Therefore, a ridge penalty that encourages smaller magnitudes of τ_{ij}'s is desirable to avoid overfitting. Furthermore, a ridge penalty can also encourage similarity among different components of $\boldsymbol{\tau}$, which also helps to foster the assumption that adjacent baseline occurrence rates should not differ much from one another. Using a ridge penalty is a common practice in many other densely parameterized machine learning models, with the most famous and popular example being (deep) neural networks (Mitchell, 1997; Goodfellow et al., 2016).

4.2.3 Scaling up Baseline Regularization

Even with the regularization introduced in (4.2), the computational burden of solving the BR model can still be staggeringly heavy. This is because a typical

EHR database can easily contain billions of days of observations from all the patients; each day will require a separate baseline parameter to describe the baseline occurrence rate of an adverse event.

Intervals

To achieve scalability without much loss of modeling flexibility, we learn lessons from the idea of *data squashing* (Madigan et al., 2002; Simpson et al., 2013) that exploits the discreteness and the sparsity of the data under consideration. Specifically, within the observational history of a particular patient, we define an *interval* as a consecutive time period during which the drug exposure statuses of *all* drugs *and* the cumulative number of adverse event occurrences remain unchanged.

Based on this definition, Figure 4.1 visualizes a patient's EHR that is divided into thirteen intervals. Each interval is indexed by I-k on the top of the figure, where $k \in \{1, 2, \cdots, 13\}$. The start date of each interval is passed through by a gray dashed line. Therefore, a previous interval ends right before a dashed line. For example, inclusively, I-1 starts from day 1 and ends at day 20 instead of day 21. Similarly, I-2 starts from day 21 and ends at day 60 instead of day 61. An exception for the unchanged cumulative adverse event occurrence restriction upon an interval is allowed if an adverse event occurs at the end of the observation. For example, in Figure 4.1, we consider I-13 ranges from day 361 to the end of the observation (day 400) even if on the last day there is a new occurrence of MI. The reason for allowing such an exception is to avoid a short (one day) interval at the end of an observation.

The concept of an interval provides convenience in describing the data concisely, and hence achieves the goal of data squashing. In Figure 4.1, instead of describing the data using information from 400 days, we can now use information from only thirteen intervals.

Parameter Tying

To reduce the number of baseline parameters used for modeling, we tie similar parameters together to the same value. Specifically, we consider two parameter

tying strategies.

- **Interval Tying**: We can consider that the baseline parameters within the same interval are the same. In this case, within a patient, the number of baseline parameters used is equal to the number of intervals instead of the number of days of observation. In Figure 4.1, this parameter tying strategy reduces the number of baseline parameters from 400 to thirteen.
- **Occurrence Tying**: We can even further tie baseline parameters from similar intervals together. For example, since ADRs are usually recurrent, and the baseline risk of getting a subsequent ADR usually changes compared with getting the first one, we can tie intervals that have the same cumulative number of adverse event occurrences together. In Figure 4.1, this parameter tying strategy will further reduce the number of baseline parameters from thirteen to four, partitioned as:

$$\{\{\text{I-1}, \text{I-2}\}, \{\text{I-3}, \cdots, \text{I-11}\}, \{\text{I-12}\}, \{\text{I-13}\}\}.$$

Reformulation

We now reformulate the BR model in (4.2) using intervals and parameter tying. Let K_i denote the number of intervals that the EHR of the i^{th} patient is partitioned into. Let κ_i represent the number of baseline parameters used in BR after parameter tying either via interval tying or via occurrence tying. We define the vector of baseline parameters after tying as:

$$\mathbf{t} = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1\kappa_1} & \cdots & t_{N1} & t_{N2} & \cdots & t_{N\kappa_N} \end{bmatrix}^\top,$$

Then the baseline parameter for each interval can also be represented as a vector: $\mathbf{Zt}$, where $\mathbf{Z}$ is a $\left(\sum_{i=1}^{N} K_i\right) \times \left(\sum_{i=1}^{N} \kappa_i\right)$ binary design matrix that maps the tied baseline parameters to the baseline parameters for each interval. Note that if the interval tying strategy is adopted, then $\kappa_i = K_i$, and $\mathbf{Z} = \mathbf{I}$, where $\mathbf{I}$ represents an identity matrix. This is because under the interval tying strategy, each interval will

have its own baseline parameter. Furthermore, we use l_{ik} to represent the duration (time length) of the k^{th} interval from the i^{th} patient, where $k \in \{1, 2, \cdots, K_i\}$. And we use n_{ik} to represent the number of adverse event occurrences during the k^{th} interval of the i^{th} patient. We further use an $M \times 1$ binary vector $\mathbf{x}_{ik}$ to represent the drug exposure statuses during the k^{th} interval of the i^{th} patient. The reason why we need only one exposure vector to represent multiple days within an interval is due to the property of unchanged drug exposure statuses of any one interval. Stacking up l_{ik}'s, n_{ik}'s, and $\mathbf{x}_{ik}$'s results in their vector and matrix representations:

$$\begin{aligned}
\mathbf{l} &= \begin{bmatrix} l_{11} & l_{12} & \cdots & l_{1K_1} & \cdots & l_{N1} & l_{N2} & \cdots & l_{NK_N} \end{bmatrix}^\top, \\
\mathbf{n} &= \begin{bmatrix} n_{11} & n_{12} & \cdots & n_{1K_1} & \cdots & n_{N1} & n_{N2} & \cdots & n_{NK_N} \end{bmatrix}^\top, \\
\mathbf{X} &= \begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} & \cdots & \mathbf{x}_{1K_1} & \cdots & \mathbf{x}_{N1} & \mathbf{x}_{N2} & \cdots & \mathbf{x}_{NK_N} \end{bmatrix}^\top.
\end{aligned}$$

Using $\mathbf{Z}$, $\mathbf{t}$, $\mathbf{X}$, $\boldsymbol{\beta}$, $\mathbf{l}$, and $\mathbf{n}$, we can rewrite the log-likelihood function in (4.1) in a matrix and vector form as follows:

$$\log \mathcal{L}(\mathbf{t}, \boldsymbol{\beta}) = \mathbf{n}^\top (\mathbf{Z}\mathbf{t} + \mathbf{X}\boldsymbol{\beta}) - \mathbf{l}^\top \exp(\mathbf{Z}\mathbf{t} + \mathbf{X}\boldsymbol{\beta}), \tag{4.3}$$

where $\exp(\cdot)$ represents a component-wise exponentiation.

After parameter tying, the fused lasso penalties imposed on $\boldsymbol{\tau}$ in (4.2) become fused lasso penalties imposed on the adjacent components of $\mathbf{t}$ that are from the same patient because under parameter tying:

$$\sum_{i=1}^{N} \sum_{j=1}^{J_i - 1} \lambda_2 |\tau_{i,j+1} - \tau_{ij}| = \sum_{i=1}^{N} \sum_{k=1}^{\kappa_i - 1} \lambda_2 |t_{i,k+1} - t_{ik}|. \tag{4.4}$$

We define $\mathbf{D}_q$ and $\mathbf{D}$ as follows:

$$\mathbf{D}_q = \begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & & \\ & & & -1 & 1 \end{bmatrix}_{(q-1)\times q}, \qquad \mathbf{D} = \begin{bmatrix} \mathbf{D}_{\kappa_1} & & & \\ & \mathbf{D}_{\kappa_2} & & \\ & & \ddots & \\ & & & \mathbf{D}_{\kappa_N} \end{bmatrix}, \tag{4.5}$$

where $\mathbf{D}_q$ is a $(q-1)\times q$ *first difference* matrix, and $\mathbf{D}$ is a *blockwise* first difference matrix. Note that $q \in \mathbb{N}_+$, and we define $D_1 = 0$.

With (4.2), (4.3), (4.4), and (4.5), we can reformulate the BR problem compactly as:

$$\arg\min_{\mathbf{t},\boldsymbol{\beta}} -\log \mathcal{L}(\mathbf{t}, \boldsymbol{\beta}) + \lambda_1 \|\boldsymbol{\beta}\|_1 + \lambda_2 \|\mathbf{D}\mathbf{t}\|_1 + \lambda_3 \|\mathbf{t}\|_2^2, \tag{4.6}$$

where we impose the same strength of ridge regularization using λ_3 on all the components of $\mathbf{t}$.

4.3 Optimization Algorithm

This section provides an optimization algorithm for solving the compact BR model in (4.6). Following the idea of `glmnet` (Friedman et al., 2010), we adopt an iteratively reweighted least squares (IRLS) approach to quadratically approximate the negative log-likelihood function. Observe that both the negative log-likelihood function and its quadratic approximation are convex, and $\boldsymbol{\beta}$ and $\mathbf{t}$ are *separable* in the regularization terms; we hence can perform blockwise minimization that alternates between $\boldsymbol{\beta}$ and $\mathbf{t}$ to achieve convergence (Tseng, 2001).

4.3.1 Quadratic Approximation

At iteration p, the iterates $\mathbf{t}^{(p)}$ and $\boldsymbol{\beta}^{(p)}$ are given. We therefore can perform a quadratic approximation of (4.3) centered at the current iterates, in order to search for the next iterates that are closest to optimality in the vicinity of the current iterates. Optimizing the quadratic approximation is equivalent to solving a weighted least

squares problem as follows:

$$\arg\min_{\mathbf{t},\boldsymbol{\beta}} \frac{1}{2} \left\| z^{(p)} - \mathbf{Z}\mathbf{t} - \mathbf{X}\boldsymbol{\beta} \right\|^2_{\mathbf{W}^{(p)}}, \tag{4.7}$$

where the working response is:

$$z^{(p)} = \mathbf{Z}\mathbf{t}^{(p)} + \mathbf{X}\boldsymbol{\beta}^{(p)} + \mathbf{W}^{(p)^{-1}}\mathbf{n} - \mathbf{1}, \tag{4.8}$$

with $\mathbf{W}^{(p)} = \mathbf{L}\mathbf{S}^{(p)}$. $\mathbf{L} = \operatorname{diag} \mathbf{l}$, and $\mathbf{S}^{(p)} = \operatorname{diag} \mathbf{s}^{(p)}$ are diagonal matrices constructed by the elements of $\mathbf{l}$ and $\mathbf{s}^{(p)}$ respectively, with $\mathbf{s}^{(p)} = \exp\left(\mathbf{Z}\mathbf{t}^{(p)} + \mathbf{X}\boldsymbol{\beta}^{(p)}\right)$; $\mathbf{1}$ is a column vector of all ones, and $\|\mathbf{a}\|^2_{\mathbf{W}} = \mathbf{a}^\top \mathbf{W}\mathbf{a}$, with $\mathbf{a}$ being a column vector and $\mathbf{W}$ being a positive diagonal matrix.

The derivation of the quadratic approximation for (4.3) basically follows from deriving the quadratic approximation of a standard Poisson regression model and the details are provided in Section 4.6.

4.3.2 Blockwise Minimization

With quadratic approximation, at iteration p with the iterates $\mathbf{t}^{(p)}$ and $\boldsymbol{\beta}^{(p)}$ available, the next iterates can be obtained by considering the following optimization problem:

$$\mathbf{t}^{(p+1)}, \boldsymbol{\beta}^{(p+1)} = \arg\min_{\mathbf{t},\boldsymbol{\beta}} \frac{1}{2} \left\| z^{(p)} - \mathbf{Z}\mathbf{t} - \mathbf{X}\boldsymbol{\beta} \right\|^2_{\mathbf{W}^{(p)}} + \lambda_1 \|\boldsymbol{\beta}\|_1 + \lambda_2 \|\mathbf{D}\mathbf{t}\|_1 + \lambda_3 \|\mathbf{t}\|_2^2. \tag{4.9}$$

We will adopt a blockwise minimization strategy that fixes $\mathbf{t}$ and $\boldsymbol{\beta}$ alternatively and solves for the other until the iterates reach the optimality of (4.9). The optimization can hence be formulated as iterating between two steps: a $\boldsymbol{\beta}$-step and a $\mathbf{t}$-step.

β-Step

We first initialize $\tilde{\mathbf{t}} = \mathbf{t}^{(p)}$. For each $\boldsymbol{\beta}$ step, we fix $\mathbf{t} = \tilde{\mathbf{t}}$ and solve the subproblem with respect to only $\boldsymbol{\beta}$ for $\tilde{\boldsymbol{\beta}}$:

$$\tilde{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2} \left\| z^{(p)} - \mathbf{Z}\tilde{\mathbf{t}} - \mathbf{X}\boldsymbol{\beta} \right\|^2_{\mathbf{W}^{(p)}} + \lambda_1 \|\boldsymbol{\beta}\|_1 , \tag{4.10}$$

which is an L_1-regularized linear regression problem that can be solved efficiently by existing packages.

t-Step

For each $\mathbf{t}$ step, we fix $\boldsymbol{\beta} = \tilde{\boldsymbol{\beta}}$, and solve the subproblem with respect to only $\mathbf{t}$ for $\tilde{\mathbf{t}}$:

$$\tilde{\mathbf{t}} = \arg\min_{\mathbf{t}} \frac{1}{2} \left\| z^{(p)} - \mathbf{X}\tilde{\boldsymbol{\beta}} - \mathbf{Z}\mathbf{t} \right\|^2_{\mathbf{W}^{(p)}} + \lambda_2 \|\mathbf{D}\mathbf{t}\|_1 + \lambda_3 \|\mathbf{t}\|_2^2 . \tag{4.11}$$

The problem in (4.11) is equivalent to:

$$\tilde{\mathbf{t}} = \arg\min_{\mathbf{t}} \frac{1}{2} \left\| \boldsymbol{\nu}^{(p)} - \mathbf{t} \right\|^2_{\boldsymbol{\Omega}^{(p)}} + \lambda_2 \|\mathbf{D}\mathbf{t}\|_1 , \tag{4.12}$$

with

$$\boldsymbol{\Omega}^{(p)} = \mathbf{Z}^\top \mathbf{W}^{(p)} \mathbf{Z} + 2\lambda_3 \mathbf{I}, \quad \boldsymbol{\nu}^{(p)} = {\boldsymbol{\Omega}^{(p)}}^{-1} \left(\mathbf{Z}^\top \mathbf{W}^{(p)} \left(z^{(p)} - \mathbf{X}\tilde{\boldsymbol{\beta}} \right) \right) .$$

The derivation from (4.11) to (4.12) is based on algebraic manipulation. Specifics are presented in the Section 4.6. The problem in (4.12) is a *blockwise weighted fused lasso signal approximator* problem. Efficient *linear time* algorithms exist for solving this type of problem (Davies and Kovac, 2001; Condat, 2013; Johnson, 2013; Ramdas and Tibshirani, 2015). Furthermore, from (4.5) we notice that $\mathbf{D}$ is blockwise, so the solutions to different blocks are independent of each other. Therefore, (4.12) can be partitioned into various independent subproblems that can be solved *in parallel* for further speedup.

Algorithm 1 Baseline Regularization

Require: $\mathbf{Z}$, $\mathbf{X}$, $\mathbf{D}$, $\mathbf{l}$, $\mathbf{n}$, λ_1, λ_2, and λ_3.
Ensure: $\hat{\boldsymbol{\beta}}$ and $\hat{\mathbf{t}}$.
1: Randomly initialize $\boldsymbol{\beta}^{(0)}$ and $\mathbf{t}^{(0)}$.
2: $p \leftarrow 0$.
3: **while** true **do** ▷ Outer loop: quadratic approximation
4: Compute $\mathbf{W}^{(p)}$ and $\mathbf{z}^{(p)}$ via (4.8).
5: $\tilde{\mathbf{t}} \leftarrow \mathbf{t}^{(p)}$.
6: **while** true **do** ▷ Inner loop: blockwise minimization
7: Solve for $\tilde{\boldsymbol{\beta}}$ via (4.10). ▷ $\boldsymbol{\beta}$-Step
8: Solve for $\tilde{\mathbf{t}}$ via (4.12). ▷ $\mathbf{t}$-Step
9: **if** Inner loop stopping criteria met **then**
10: $p \leftarrow p + 1$, $\boldsymbol{\beta}^{(p)} \leftarrow \tilde{\boldsymbol{\beta}}$, and $\mathbf{t}^{(p)} \leftarrow \tilde{\mathbf{t}}$.
11: **break**.
12: **end if**
13: **end while**
14: **if** Outer loop stopping criteria met **then**
15: $\hat{\boldsymbol{\beta}} \leftarrow \boldsymbol{\beta}^{(p)}$, and $\hat{\mathbf{t}} \leftarrow \mathbf{t}^{(p)}$.
16: **return** $\hat{\boldsymbol{\beta}}$ and $\hat{\mathbf{t}}$.
17: **end if**
18: **end while**

4.3.3 Implementation

The optimization algorithm for the BR model is summarized in Algorithm 1. Several important implementation details follow:

- To solve the problem in Step 7, we use the `glmnet` (Friedman et al., 2010) package available in `R`. To solve the problem in Step 8, we use the functions from the `C` library of the `glmgen` (Ramdas and Tibshirani, 2015) package in `R`. Both implementations are considered to be the state-of-the-art solvers for the respective subproblems.

- To avoid the divergence issue due to an ill-conditioned $\mathbf{W}^{(p)}$, we set all the diagonal elements of $\mathbf{W}^{(p)}$ that are smaller than a certain threshold, ϵ, to that threshold. In our experiments, we choose $\epsilon = 10^{-5}$. Our compact BR

model by design helps to alleviate the ill-conditioned issue because a diagonal element of $\mathbf{W}^{(\mathtt{p})}$ represents the *cumulative* occurrence rate of adverse events during an entire interval. Ridge regularization over baseline parameters also helps to avoid small diagonal elements.

- Selection of the inner loop stopping criteria in Step 9 and the outer loop stopping criteria in Step 14 is problem-specific. We describe our choice in Section 4.4.4.

Our algorithmic framework shares similarities with that of `glmnet`. Both methods in the outer loop perform a quadratic approximation to a generalized linear model negative log-likelihood objective with non-smooth regularization. Both methods leverage an efficient inner loop blockwise minimization solver for the approximated problem. Therefore, both can be considered being in the family of proximal Newton methods (Sra et al., 2012; Pena and Tibshirani, 2016). Compared with first order methods, it is well known that the proximal Newton method shares the same fast convergence rate as the usual Newton method in terms of the number of (proximal) Newton's steps needed (i.e., the number of outer loop iterations needed). However, proximal Newton methods suffer from inefficiency due to the expensive evaluations of the Hessian matrix in general. Therefore, the fact that methods under the proximal Newton framework such as `glmnet` can deliver solutions for even large-scale problems efficiently is counter-intuitive at first glance, and yet is actually attainable using an efficient inner solver (Pena and Tibshirani, 2016). Further illustrated by the experimental results to come, our algorithm provides yet another example demonstrating that the proximal Newton framework, with appropriate execution, can have the potential to handle large-scale problems effectively.

Table 4.1: Summary statistics of the experiment cohort

Statistics	Values
# patients	216,660
# condition (adverse event) records	1,982,000
# drug prescription records	9,089,238
Average observation duration	11.3 years

4.4 Experiments

4.4.1 The Benchmark Task

To empirically evaluate the performance of our proposed method, we use a ground truth set of 53 drug-condition pairs generated by a selective combination of ten different drugs and nine different conditions proposed by the Observational Medical Outcomes Partnership (OMOP) (Simpson, 2011), which was a pilot project in the U.S. aiming to conduct methodological research for the identification of ADRs from LODs. Among the 53 drug-condition pairs, 9 pairs are identified as *positive* cases (confirmed ADRs), and the remaining 44 are identified as *negative* controls. Distinguishing positive cases from the negative controls in the OMOP ground truth is widely considered to be a benchmark task for ADR discovery from LODs.

4.4.2 Data Source

We use the Marshfield Clinic EHR database as our data source. Being a pioneer for deploying EHR systems, Marshfield Clinic EHR database is one of the richest and the most historic in the United States, with coded diagnoses recorded as early as in 1960, and other electronic contents dating back to the 1980s (Powell et al., 2012). We convert the diagnosis records and the drug prescription records in the EHRs to a format that is compliant with the vocabularies used in the OMOP ground truth. Following the design of MSCCS, we admit a patient into the cohort if he or she has at least one condition of interest (adverse event) occurrence throughout the entire observation. We also further restrict our attention to patients with at least

one OMOP ground truth drug prescription record during the entire observation. Table 4.1 provides summary statistics of the cohort used in our experiments.

4.4.3 Cohort Design

We consider two important cohort design choices:

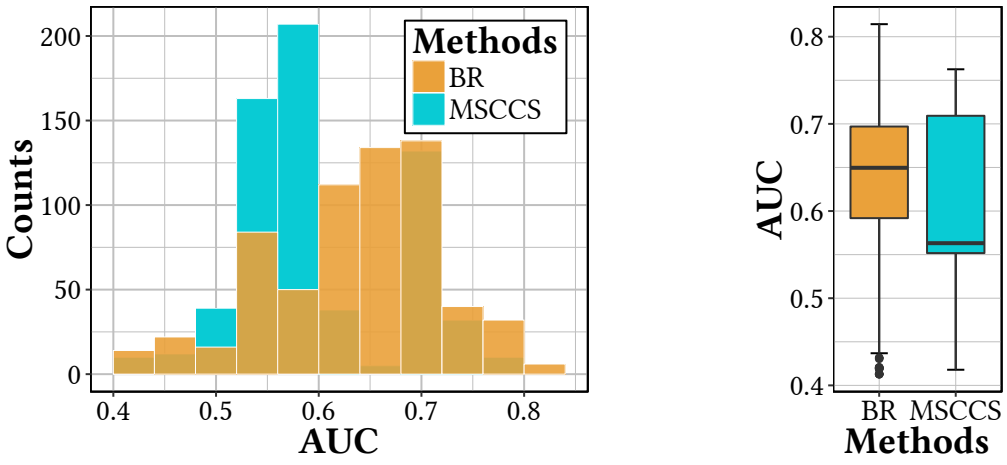
- **Risk Window Design**: a risk window is a time span that follows right after the end of a drug era during which the patient is still considered under exposure. Three types of risk windows are considered, `none`, `one month`, and `lasting`. The names of the risk windows are suggestive of their meanings.
- **Minimum Duration Design**: duration is the time length of the observation for a patient. Other than meeting the cohort admission requirement specified in Section 4.4.2, we admit a patient only when his or her observation duration surpasses the minimum duration threshold. We set three different minimum duration thresholds in our experiments, `none`, `three months`, and `six months`.

4.4.4 BR Algorithmic Design

Stopping Criteria

We denote the Euclidean norm of the difference of the two parameter vectors from the last two inner (outer) loop iterations as δ_i (δ_o). We denote the number of inner (outer) loop iterations that have run so far as c_i (c_o).

The design of the inner loop stopping criteria follows a coarse-to-fine strategy depending on how close the current outer loop iterate is to optimality. Specifically, the inner loop stopping criteria are met if any one of the following three conditions is true: (1) $\delta_o > 10$ and $\delta_i < 0.05\delta_o$; (2) $\delta_o \leqslant 10$ and $\delta_i < \max\left\{10^{-3}\delta_o, 10^{-4}\right\}$; (3) $c_i \geqslant 200$. The first criterion is useful when the current outer loop iterate is far from optimality (characterized by $\delta_o > 10$). In this case, a small number of inner loop iterations can decrease the objective effectively such that $\delta_i < 0.05\delta_o$, but further



(a) Overlapping Histograms

(b) Boxplot

Figure 4.2: Overall performance of BR and MSCCS measured by AUC among 648 different experimental configurations.

inner loop iterations do not yield much more progress. Therefore, this criterion allows the first several iterations that make significant progress, but truncates the rest that are not as effective. The second criterion determines when the inner loop stops when the current outer loop iterate is close to optimality (characterized by $\delta_o \leqslant 10$). In this case, the inner loop estimation needs to be more accurate to ensure that solving subsequent quadratic approximations can further decrease the objective. Therefore, the second criterion dictates that the inner loop will stop only when the estimation error is reasonably small.

The outer loop stopping criteria are met if either one of the following two conditions is true: (i) $c_o \geqslant 60$; (ii) $\delta_o < 10^{-4}$. Note that after each outer loop iteration, c_i is reset to 0.

Tuning Parameters

Since there are only ten different drugs available in the OMOP ground truth, the dimension of $\mathbf{X}$ is low. Therefore, we decide not to regularize $\boldsymbol{\beta}$ at all by simply setting $\lambda_1 = 0$ to decrease the complexity of the design choice space. Nonethe-

less, we still use `glmnet` to solve the resultant standard weighted least squares problem due to its matrix-vector friendly interface and high efficiency. We choose $\lambda_2 \in \{0.1, 0.5, 1, 2, 4, 8\}$, and $\lambda_3 \in \left\{0, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\right\}$. Note that to avoid over-parameterization λ_2 cannot be too small. And finally, we also vary the two parameter tying strategies in Section 4.2.3.

The selection of λ_2, λ_3, and parameter tying strategies, along with the nine cohort design choices in Section 4.4.3, result in 648 different experimental configurations. Since there are nine different types of conditions, the number of BR models that are evaluated in our experiments is $648 \times 9 = 5832$.

4.4.5 MSCCS Algorithmic Design

An MSCCS model is an equivalent compact representation of a fixed effect Poisson regression model (Xu et al., 2012). We therefore are able to use `glmnet` as a solver for MSCCS by learning the corresponding fixed effect Poisson regression model directly. MSCCS is a model that is only related to $\boldsymbol{\beta}$, upon which we impose a ridge penalty in our experiments. Since both BR and MSCCS share the same cohort design choices, to generate 648 experimental configurations for MSCCS as well, we use a list of 72 tuning parameters for the ridge penalty by ranging the `lambda` option in the `glmnet` function in `R` from 10^{-10} to 10 evenly in logarithmic scale. MSCCS without a ridge penalty is also considered. We also apply MSCCS on each of the nine different conditions, resulting in a total of 5832 different MSCCS models.

4.4.6 Metrics

For each of the 5832 models from both methods (BR and MSCCS), we rank the drugs in ascending order of the corresponding coefficients in the learned $\boldsymbol{\beta}$. For each of the two methods, among the models that have the *same* experimental configurations, we compute the area under curve (AUC) of receiver operating characteristics (ROC) using the OMOP ground truth and the *rankings* generated in the previous step. In this way, for both BR and MSCCS, we have 648 AUCs, each for one of the experimental configurations.

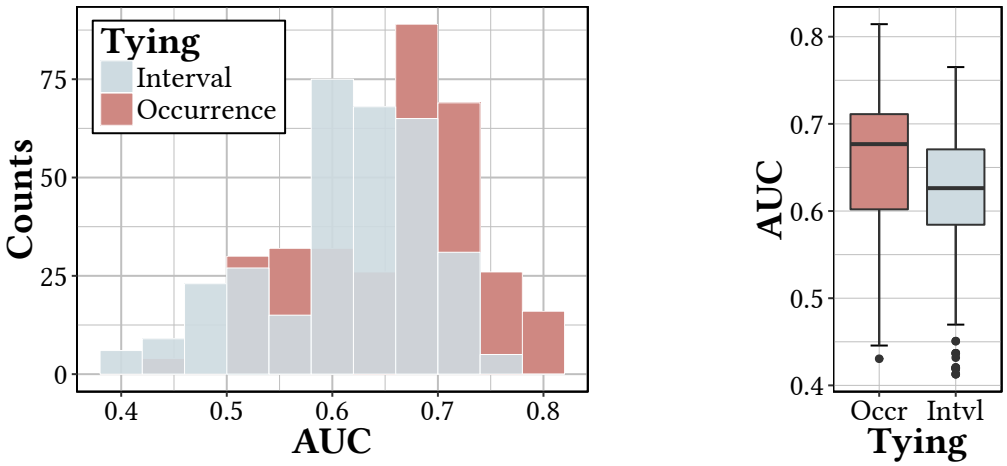
4.4.7 Results of Overall Performance

Since the deployed methods for ADR discovery from LODs usually reported their performances on all experimental configurations (Ryan et al., 2012; Madigan et al., 2013; Norén et al., 2013; Ryan et al., 2013a,b; Schuemie et al., 2013; Suchard et al., 2013b), following this protocol, we also analyze the performances of BR and MSCCS under all of our experimental configurations.

Figure 4.2 visualizes the distributions of AUCs of BR and MSCCS across all 648 experimental configurations. The histogram and the box in brown represent the AUC distribution of BR and the cyan ones represent MSCCS. Compared with the AUC distribution of MSCCS, the AUC distribution of BR shifts significantly towards higher AUC intervals, with most experimental configurations achieving AUCs of more than 0.6. On the other hand, most of the experimental configurations for MSCCS achieve AUCs only between 0.5 and 0.6, which is an indication that most experimental configurations of MSCCS lack the discriminative power to separate the positive cases from the negative controls. The comparison of the overall performances between the two methods suggests that exploiting the time-inhomogeneous nature of EHR data can potentially help to more accurately quantify the effects of drugs on the occurrence rate of adverse events.

4.4.8 Results of Parameter Tying

Figure 4.3 illustrates the effects of the two parameter tying strategies presented in Section 4.2.3 on the performance of various BR models. The distribution generated by occurrence tying lies in a range with higher AUCs compared with the distribution generated by interval tying. This phenomenon might be related to the clinical belief that baseline recurrence rates of adverse events tend to be different from the first occurrence rate. While occurrence tying offers a principal way to quantify this type of prior belief, interval tying might introduce redundant flexibility that focuses on perturbational baseline difference between every adjacent pair of intervals, resulting in the potential tendency to overfit the data.



(a) Overlapping Histograms

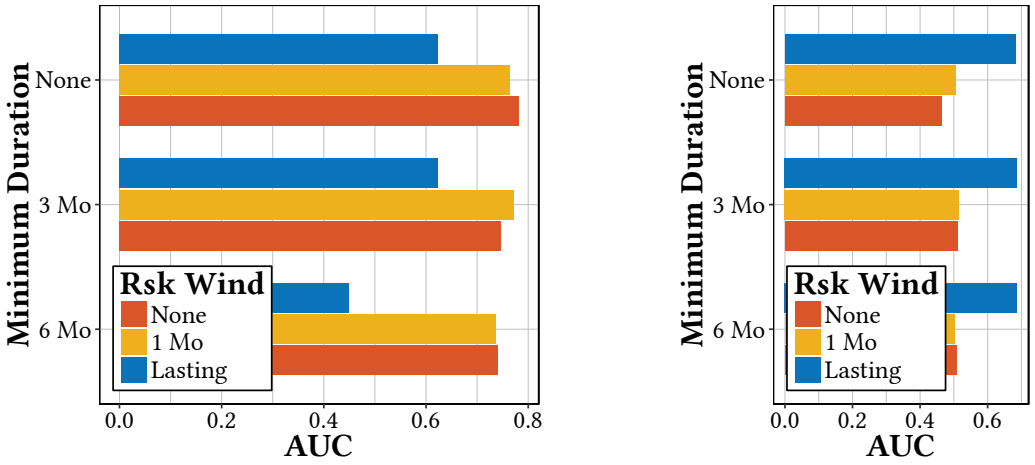
(b) Boxplot

Figure 4.3: Performance of BR using the two parameter tying strategies in Section 4.2.3 evaluated among 648 different experimental configurations, each strategy is evaluated upon 324 configurations.

4.4.9 Model Selection and Generalization

To demonstrate how well BR can predict unseen adverse events, for a given cohort design choice, we perform Leave-One-Condition-Out-Cross-validation (LOCOCV): for each of the nine conditions, we jointly and adaptively pick λ_2, λ_3, and the tying strategy that perform the best on the other eight conditions. In this way, we are able to use the top performer on the known ground truth to predict the unknown. We find LOCOCV to be a reasonable model selection strategy because, in essence, BR transforms the unsupervised learning of ADRs into a supervised learning problem. During learning, *none* of the ground truth label information is used. In this scenario, using LOCOCV helps us to maximize the number of training instances that can be used without worrying about the overfitting issues introduced by the ADR label information.

The AUCs of the nine different cohort design choices generated by LOCOCV are given in Figure 4.4. Other than under the `lasting` risk window, the AUCs of LOCOCV under other configurations exceed 0.7. In comparison, the best LOCOCV



(a) BR (b) MSCCS

Figure 4.4: Performance of leave-one-condition-out-cross-validation (LOCOCV) among the nine cohort design choices.

AUC from MSCCS is less than 0.7, which occurs when using a `lasting` risk window. Other configurations of MSCCS provide AUCs of around 0.5. The reasons why we are committed to various cohort design choices are that both BR and MSCCS share the same set of cohort design choices, and that given a cohort design, the data (i.e., $\mathbf{X}$, $\mathbf{l}$, and $\mathbf{n}$) used by the two methods are exactly the same, and hence a fair comparison between the two methods can be achieved. Furthermore, in a practical setting, committing to a particular design choice can also help to facilitate the comparison of performances among multiple data sources (Simpson et al., 2013).

4.4.10 Best Performers

In the literature of ADR discovery from LODs, it is customary to report the best performer of a method learned from a data source (Suchard et al., 2013b; Ryan et al., 2013b; Norén et al., 2013; Ryan et al., 2013a; Madigan et al., 2013; Schuemie et al., 2013). Therefore, we also report our top performers of BR and MSCCS in our experiments: the best BR model reaches an AUC of 0.814, with a `none` risk window, a `six months` minimum duration threshold, using occurrence tying, $\lambda_2 = 0.5$, and

$\lambda_3 = 0.1$. The best performer of MSCCS reaches an AUC of 0.763, with a `lasting` risk window, a `three months` minimum duration threshold, and `lambda`$\approx$`2.5e-3`.

4.5 Discussion

We have proposed baseline regularization for ADR discovery from LODs. We provide an effective algorithm from the proximal Newton framework for solving the BR model and compare the performance of BR with MSCCS in a set of diverse experimental configurations. Future research directions include running BR on other LODs for reproducibility, and accelerating the algorithm by incorporating stochasticity (Nesterov, 2012; Zhao et al., 2014; Wright, 2015) and parallelism (Wright, 2015).

4.6 Auxiliary Results

4.6.1 Quadratic Approximation of (4.3)

Let

$$f(\mathbf{t}, \boldsymbol{\beta}) = -\log \mathcal{L}(\mathbf{t}, \boldsymbol{\beta}) = -\mathbf{n}^\top (\mathbf{Z}\mathbf{t} + \mathbf{X}\boldsymbol{\beta}) + \mathbf{l}^\top \mathbf{s},$$

where $\mathbf{s} = \exp(\mathbf{Z}\mathbf{t} + \mathbf{X}\boldsymbol{\beta})$. Note that $\mathbf{s} > \mathbf{0}$ (each component of $\mathbf{s}$ is strictly larger than 0) as long as $\mathbf{Z}\mathbf{t} + \mathbf{X}\boldsymbol{\beta}$ is bounded. For the ease of derivation, we also assume that $\begin{bmatrix}\mathbf{Z} & \mathbf{X}\end{bmatrix}$ is a column full rank matrix. In this way, an invertible Hessian of $f(\mathbf{t}, \boldsymbol{\beta})$ can be guaranteed. The gradient and the Hessian of $f(\mathbf{t}, \boldsymbol{\beta})$ are:

$$\nabla f(\mathbf{t}, \boldsymbol{\beta}) = \begin{bmatrix}\mathbf{Z}^\top \\ \mathbf{X}^\top\end{bmatrix} (\mathbf{L}\mathbf{s} - \mathbf{n}), \nabla^2 f(\mathbf{t}, \boldsymbol{\beta}) = \begin{bmatrix}\mathbf{Z}^\top \\ \mathbf{X}^\top\end{bmatrix} \mathbf{W} \begin{bmatrix}\mathbf{Z} & \mathbf{X}\end{bmatrix}, \tag{4.13}$$

where $\mathbf{W} = \mathbf{L}\mathbf{S}$, and $\mathbf{S} = \operatorname{diag} \mathbf{s}$.

At iteration p, $\mathbf{t}^{(p)}$ and $\boldsymbol{\beta}^{(p)}$ are given. One can show that optimizing the quadratic approximation of $f(\mathbf{t}, \boldsymbol{\beta})$ around $\mathbf{t}^{(p)}$ and $\boldsymbol{\beta}^{(p)}$ is equivalent to computing a Newton's update. Using (4.13) and following (Murphy, 2012), a Newton's update

for $\mathbf{t}^{(p+1)}$ and $\boldsymbol{\beta}^{(p+1)}$ is given as:

$$\begin{bmatrix} \mathbf{t}^{(p+1)} \\ \boldsymbol{\beta}^{(p+1)} \end{bmatrix} = \left(\begin{bmatrix} \mathbf{Z}^\top \\ \mathbf{X}^\top \end{bmatrix} \mathbf{W}^{(p)} \begin{bmatrix} \mathbf{Z} & \mathbf{X} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{Z}^\top \\ \mathbf{X}^\top \end{bmatrix} \mathbf{W}^{(p)} \mathbf{z}^{(p)},$$

which is the solution to the weighted least squares problem in (4.7), with $\mathbf{z}^{(p)}$ defined in (4.8).

4.6.2 Derivation from (4.11) to (4.12)

As a preparation, we state the following two algebraic facts as lemmas.

Lemma 1. Let $\mathbf{y}$ be an $n \times 1$ vector, let $\mathbf{X}$ be an $n \times p$ matrix, let $\boldsymbol{\beta}$ be a $p \times 1$ vector, and let $\mathbf{W}$ be a positive diagonal matrix. Then:

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_{\mathbf{W}}^2 = \arg\min_{\boldsymbol{\beta}} \frac{1}{2} (\mathbf{X}\boldsymbol{\beta})^\top \mathbf{W} (\mathbf{X}\boldsymbol{\beta}) - \left(\mathbf{y}^\top \mathbf{W}\right) (\mathbf{X}\boldsymbol{\beta}).$$

Proof. The equation obviously holds by expanding the left hand side of the equation and removing the quantities that are not related to $\boldsymbol{\beta}$. □

Lemma 2. Let $\mathbf{y}_1$ and $\mathbf{y}_2$ be two $n \times 1$ vectors, let $\mathbf{X}$ be an $n \times p$ matrix, let $\boldsymbol{\beta}$ be a $p \times 1$ vector, and let $\mathbf{W}_1$, $\mathbf{W}_2$ be two positive diagonal matrices. Then:

$$\begin{aligned} &\arg\min_{\boldsymbol{\beta}} \frac{1}{2} \|\mathbf{y}_1 - \mathbf{X}\boldsymbol{\beta}\|_{\mathbf{W}_1}^2 + \frac{1}{2} \|\mathbf{y}_2 - \mathbf{X}\boldsymbol{\beta}\|_{\mathbf{W}_2}^2 \\ &\quad = \arg\min_{\boldsymbol{\beta}} \frac{1}{2} \left\| (\mathbf{W}_1 + \mathbf{W}_2)^{-1} (\mathbf{W}_1\mathbf{y}_1 + \mathbf{W}_2\mathbf{y}_2) - \mathbf{X}\boldsymbol{\beta} \right\|_{\mathbf{W}_1+\mathbf{W}_2}^2. \end{aligned}$$

Proof. By applying Lemma 1, the quantities on both sides of the equality can be shown to be equal to:

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2} (\mathbf{X}\boldsymbol{\beta})^\top (\mathbf{W}_1 + \mathbf{W}_2) (\mathbf{X}\boldsymbol{\beta}) - \left(\mathbf{y}_1^\top \mathbf{W}_1 + \mathbf{y}_2^\top \mathbf{W}_2\right) (\mathbf{X}\boldsymbol{\beta}).$$

□

We now proceed to the derivation. For convenience, we omit all the (p) superscripts and we use $\boldsymbol{\nu} = \boldsymbol{z}^{(p)} - \mathbf{X}\tilde{\boldsymbol{\beta}}$. We first show that:

$$\begin{aligned}
&\arg\min_{\mathbf{t}} \frac{1}{2} \|\boldsymbol{\nu} - \mathbf{Z}\mathbf{t}\|_{\mathbf{W}}^2 + \lambda_2 \|\mathbf{D}\mathbf{t}\|_1 \\
&\qquad = \arg\min_{\mathbf{t}} \frac{1}{2} \left\| \left(\mathbf{Z}^\top \mathbf{W}\mathbf{Z}\right)^{-1} \mathbf{Z}^\top \mathbf{W}\boldsymbol{\nu} - \mathbf{t} \right\|_{\mathbf{Z}^\top \mathbf{W}\mathbf{Z}}^2 + \lambda_2 \|\mathbf{D}\mathbf{t}\|_1 .
\end{aligned}$$

This is true because by applying Lemma 1, the quantities on both sides of the equality are equal to:

$$\arg\min_{\mathbf{t}} -\boldsymbol{\nu}^\top \mathbf{W}\mathbf{Z}\mathbf{t} + \frac{1}{2} (\mathbf{Z}\mathbf{t})^\top \mathbf{W} (\mathbf{Z}\mathbf{t}) + \lambda_2 \|\mathbf{D}\mathbf{t}\|_1 .$$

It remains to show that

$$\arg\min_{\mathbf{t}} \frac{1}{2} \left\| \left(\mathbf{Z}^\top \mathbf{W}\mathbf{Z}\right)^{-1} \mathbf{Z}^\top \mathbf{W}\boldsymbol{\nu} - \mathbf{t} \right\|_{\mathbf{Z}^\top \mathbf{W}\mathbf{Z}}^2 + \lambda_3 \|\mathbf{t}\|_2^2 = \arg\min_{\mathbf{t}} \frac{1}{2} \|\boldsymbol{\nu} - \mathbf{t}\|_{\boldsymbol{\Omega}}^2 ,$$

which is an immediate consequence of applying Lemma 2 with the fact that $\mathbf{W}_1 = \mathbf{Z}^\top \mathbf{W}\mathbf{Z}$ and $\mathbf{W}_2 = 2\lambda_3 \mathbf{I}$.

Part III

Irregularity

EHRs are a collection of time-stamped events that occur irregularly and spontaneously. In Part III, we first present a principled approach to addressing the irregularity challenge via the use of point process models and kernel functions in Chapter 5. Our approach further improves the performance of ADR discovery. Subsequently in Chapter 6, we leverage the idea of kernel function for the task of CDR. Our results suggest that a careful treatment of the irregularity issue in the data can yield improved causal fidelity.

5 HAWKES PROCESS MODELING OF ADVERSE DRUG REACTIONS WITH LONGITUDINAL EVENT DATA

5.1 Introduction

As noted in Chapter 1, 2, 4, and 7, adverse drug reaction (ADR) discovery is the task of finding unexpected and negative effects of drugs prescribed to patients. ADR discovery is a major public health challenge. It is estimated that ADRs cause 4.2-30% of hospitalizations in the United States and Canada, with an approximated relevant annual cost of 30.1 billion US dollars in the United States (Sultana et al., 2013). Although the U.S. Food and Drug Administration (FDA) has established one of the most rigorous drug preapproval procedures in the world, many potential ADRs of a drug may not be identified in its developmental stage. During the preapproval clinical trials, a drug might be tested on just a few thousand people. Therefore, ADRs with low occurrence rates are likely not to be identified in this relatively small population. However, these ADRs might occur and even become a public health hazard after the drug is introduced to the market, where potentially millions of people with much more diverse profiles are taking the drug. Therefore, postmarketing surveillance methods that can quickly and effectively detect potential ADRs are highly desirable to address this major public health challenge.

Modern postmarketing surveillance (Robb et al., 2012; Findlay, 2015; Hripcsak et al., 2015) leverages machine learning algorithms for ADR discovery (Ryan et al., 2012; Madigan et al., 2013; Norén et al., 2013; Ryan et al., 2013a,b; Schuemie et al., 2013; Suchard et al., 2013b) on large-scale longitudinal event databases (LEDs) such as insurance claim databases and electronic health records (EHRs), where drug prescription records, adverse health outcome occurrences, and demographic information from millions of individuals are collected as time-event pairs. A leading model used for ADR discovery from LEDs is the multiple self-controlled case series (MSCCS, Simpson et al. 2013). In MSCCS, we only consider individuals with at least one occurrence of an adverse health outcome of interest as cases. By estimating

the occurrence rates of the adverse events when the individuals are exposed (or not exposed) to various drugs, each individual can serve as his/her own control, potentially linking the elevation of the occurrence rate of adverse events to the exposure of particular drugs and providing evidence for ADR discovery.

While MSCCS has gained tremendous empirical success (Simpson et al., 2013; Suchard et al., 2013b) in identifying benchmark ADRs defined by the Observational Medical Outcomes Partnership (OMOP), the model relies on somewhat restrictive assumptions due to the nature of irregular event occurrences in the data:

- **Drug Era Construction**: In MSCCS, upon the prescription of a drug to a patient, the patient is assumed to be under the exposure of the drug for a continuous period of time called drug era (Reisinger et al., 2010). Since in most EHRs, only the irregular time-stamped drug prescription records are available, drug eras are usually constructed manually based on heuristics that incorporate adjacent time-stamped drug prescription records of the same drug. A data-driven, drug-era-free approach that directly leverages the time-stamped information in the EHRs is hence highly desirable to represent the influence of a particular drug upon the occurrence of an adverse event.

- **Time-Invariant Drug Effect**: Standard MSCCS also assumes that during a drug era, the effect of the drug on the occurrence rate of the adverse event remains constant. This obviously is an over-simplification in practice, as different drugs exhibit different pharmacokinetics and exert different dynamic impacts at different times. While efforts have been made to extend self-controlled case series to address time-varying drug effects for a single drug (Schuemie et al., 2016), modeling time-varying drug effects on adverse events for multiple drugs in large-scale LEDs remains underdeveloped.

To circumvent the aforementioned weakness of MSCCS, we propose a log-linear Hawkes process (Hawkes, 1971a,b) for adverse drug reaction discovery with longitudinal event data. A central component of the Hawkes process is its flexible representation power to depict self-excitation and mutual-excitation of past events

of various types to future events via triggering influence functions. Specifically, we propose using dyadic influence functions in lieu of the construction of drug eras to represent the effect of a drug on the future occurrence rate of an adverse event. In this way, the influence of a drug on an adverse event is modulated by the gap between the drug prescription time and the adverse event occurrence, offering a solution to mitigate the irregularity issue of LED for higher causal fidelity.

To the best of our knowledge, this work is the first attempt to model longitudinal event data as a log-linear Hawkes process for ADR discovery. Experimental results on a real-world EHR demonstrate that the proposed method outperforms MSCCS in various settings.

5.2 Modeling framework

For each patient $p \in \{1, \ldots, P\}$, we observe $N_p > 0$ events. The i^{th} event is described by its time, $\tau_{p,i}$, and type, $m_{p,i}$, where $\tau_{p,i} \leqslant \tau_{p,i+1}$ for $i = 1, 2, \ldots, N_p - 1$. The times are generally discretized by EHR software to be accurate within eight hours. Assuming a sampling period of length $\Delta = 8$ hours, we let $x_{p,m,t}$ be the number of events at any time $\tau \in [\Delta t, \Delta(t+1))$ of type m for patient p. Event types m belong to a set $\mathcal{M} = \mathcal{D} \cup \mathcal{O}$, where $\mathcal{D}$ is the set of possible drug prescription events and $\mathcal{O}$ is the set of adverse health outcomes.

A complicating factor in predicting ADRs is that we do not know when a patient is actively taking a drug; we can only observe when the drug is prescribed, and different prescriptions can have different durations. This challenge has been noted before (Kuang et al., 2016c). A heuristic proposed in the Common Data Model (CDM, Reisinger et al. 2010) by Observational Medical Outcome Partnership (OMOP) is to assume that each drug has a *time-at-risk window*, which is comprised of (a) the drug era, or the times when a patient is assumed to be taking a drug based on the prescription date recorded in the EHR, and (b) the drug exposure window, or the times when a patient is assumed to still be reacting to a drug even though the prescription has ended.

In this chapter, we denote the length of the time-at-risk window as L. That is, L is a measure of real time (hours), and L/Δ is a measure of the number of discrete time intervals (*e.g.*, 8-hours periods) in which the EHR data is stored.

Throughout this chapter, we model the outcome events as realizations of a point process with time-varying rate λ. ADR analysis is the process of estimating λ from data and determining which factors from a patient's EHR most contribute to accurate predictions of ADRs. In what follows, we first describe the commonly-used *Multiple Self-Controlled Case Series* (MSCCS, Simpson et al. 2013) and then our proposed *log-linear Hawkes* model.

5.2.1 Multiple Self-Controlled Case Series Model

Multiple self-controlled case series (MSCCS, Simpson et al. 2013) is one of the leading methods for ADR discovery. Given L, the MSCCS model can be specified as follows. First, define

$$\tilde{x}_{p,m,t} := \begin{cases} 1, & \text{if } \exists s \in \{t - L/\Delta + 1, \dots, t\} \text{ such that } x_{p,m,s} > 0 \\ 0, & \text{otherwise} \end{cases};$$

then $\tilde{x}_{p,m,t}$ indicates whether patient p was prescribed drug m at any point in the past L/Δ time units up until time t. We may then model the log-rate of ADR $o \in \mathcal{O}$ for patient p at time t as

$$\log \lambda_{p,o,t} = \Delta b_{p,o} + \sum_{d \in \mathcal{D}} w_{o,d} \tilde{x}_{p,d,t} \tag{5.1}$$

for some unknown weights $\{w_{o,d}\}_{d \in \mathcal{D}}$ and unknown baseline event rate $b_{p,o}$, which can be different for each patient.

Given this rate, we model our observations of ADRs using a Poisson distribution, so that the probability of patient p experiencing outcome o at time t is

$$\mathbb{P}(x_{p,o,t} | \lambda_{p,o,t}) = \frac{e^{-\lambda_{p,o,t}} \lambda_{p,o,t}^{x_{p,o,t}}}{x_{p,o,t}!}. \tag{5.2}$$

The model in (5.1) says that the log of this rate parameter is the sum of a patient-specific baseline rate and a weighted combination of the different events the patient is simultaneously experiencing. *The weights* $\{w_{o,d}\}_{\substack{o\in\mathcal{O}\\ d\in\mathcal{D}}}$ *indicate how well we may predict outcome* o *based on a patient being on drug* d.

While this model is popular in the literature and practice (Simpson et al., 2013; Suchard et al., 2013b), choosing the time-at-risk window L can still confound analysis. The time-at-risk window L is generally chosen based on side information about common drug prescription durations, or is treated as a tuning parameter to be chosen based on data. If L is small, then the model behaves as if the patient is not on the drug L hours after the prescription is recorded, thus potentially masking longer-term causal effects. In contrast, if L is large, then it is difficult to distinguish the effect of a drug prescribed recently and a different drug prescribed in the distant past; in fact, MSCCS would treat those prescriptions as equal.

Another interpretation of the MSCCS model is that EHRs have *missing data* about which drugs patients are taking at what times. The $\tilde{x}_{p,d,t}$'s can be considered a combination of the original data and *imputed events* which may or may not be real. *Injecting artificial events into a patient's EHR poses significant risks for biased analysis leading to false conclusions.*

5.2.2 Hawkes model

We propose to use a Hawkes process model (Hawkes, 1971a,b; Daley and Vere-Jones, 2003) as an alternative to MSCCS. Hawkes processes have been used to model spike trains recorded from biological neural networks (Pillow et al., 2008), interactions within a social network (Hall and Willett, 2013), pricing changes within financial networks (Chavez-Demoulin and McGill, 2012), power failures in networked electrical systems (Ertekin et al., 2015), crime and military engagements (Linderman and Adams, 2014), and in a variety of other settings.

The log-linear Hawkes process shares several features with MSCCS, but is a continuous-time model that can *account for the influence of past events on future events*. To specify the Hawkes process, we first define a collection of K *influence functions*,

$\{\phi_k(\cdot)\}_{k=0}^{K-1}$. The Hawkes process can be expressed in terms of any influence functions, and we describe our specific choice for ADR analysis in Section 5.2.2. Given these influence functions, the log-rate of the Hawkes process has the form

$$\log \lambda_{p,o}(\tau) = b_{p,o} + \sum_{d \in \mathcal{D}} \sum_{k=0}^{K-1} w_{o,d,k} \sum_{\substack{i \leqslant N_p: \\ \tau_{p,i} \leqslant \tau \\ m_{p,i} = d}} \phi_k(\tau - \tau_{p,i}). \tag{5.3}$$

Similar to MSCCS, the log of this rate parameter is the sum of a patient-specific baseline rate and a weighted combination of patient features. Unlike MSCCS, *the Hawkes model naturally accounts for the influence of past events without requiring the analyst to inject artificial events into a patient's EHR to account for the (unknown) time-at-risk window.* The weights $\{w_{o,d,k}\}_{o \in \mathcal{O},\, d \in \mathcal{D},\, k \in \{1,\ldots,K\}}$ indicate how well we may predict outcome o based on a patient being on drug d according to the k^{th} influence function. Each influence function reflects how the influence of a past event changes based on how much time has elapsed since that event. We generally expect that more recent events have more bearing on a patient's risk of an ADR.

Choice of Influence Functions

For ADR analysis, we propose choosing the influence functions (ϕ_k's) to be piecewise constant functions supported on bounded intervals. Specifically, let K be the number of influence functions in our Hawkes model, and let L be the length of the maximum time-at-risk window.

For each k, we define an interval $I_k = [a_k, b_k)$ that satisfies the constraint that the collection of all K intervals cover the entire time-at-risk window $[0, L)$ (that is, $\bigcup_{k=0}^{K-1} I_k = [0, L)$). Then

$$\phi_k = \frac{1}{b_k - a_k} \mathbf{1}_{\{\tau \in I_k\}}.$$

Note that these ϕ_k's all integrate to one and are orthogonal to one another. By picking different pairs (a_k, b_k), we can jointly model short-term and long-term effects.

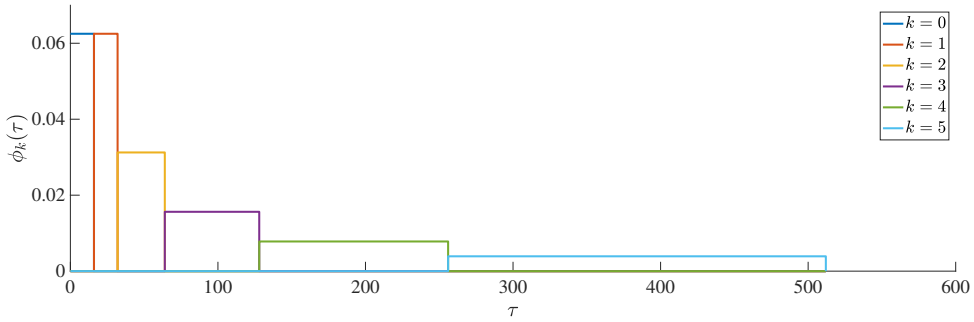


Figure 5.1: Dyadic influence functions for $L = 512$ and $K = 6$.

In our experiments, we focus on ϕ_k's where the intervals are chosen as follows. Define

$$\alpha_k := \begin{cases} 2^{K-1}/L, & k = 0, \\ 2^{K-k}/L, & k = 1, \dots, K-1, \end{cases}$$

and the intervals

$$I_k := \begin{cases} \left[0, 1/\alpha_k\right), & k = 0, \\ \left[1/\alpha_k, 2/\alpha_k\right), & k = 1, \dots, K-1 \end{cases}.$$

Then we define

$$\phi_k(\tau) = \alpha_k \mathbf{1}_{\{\tau \in I_k\}}$$

where $\mathbf{1}_{\{A\}} = \begin{cases} 1, & A \text{ true}, \\ 0, & A \text{ false} \end{cases}$ is the indicator function.

We refer to the above choice of influence functions (depicted in Figure 5.1) as *dyadic influence functions* because they are supported on dyadic intervals that correspond to dividing the interval $[0, L)$ in half repeatedly.

Hawkes processes with dyadic influence functions

In this subsection, we examine the Hawkes model of (5.3) in the specific case of dyadic influence functions. In particular, we note that for $\tau \in [\Delta t, \Delta(t+1))$,

$$\sum_{\substack{i \leqslant N_p: \\ \tau_{p,i} \leqslant \tau \\ m_i = d}} \phi_k(\tau - \tau_{p,i}) = \sum_{\substack{i \leqslant N_p: \\ \tau_{p,i} \leqslant \tau \\ m_i = d}} \alpha_k \mathbf{1}_{\tau - \tau_{p,i} \in I_k} = \alpha_k \sum_{s:(t-s)\Delta \in I_k} x_{p,d,s}.$$

Define

$$z_{p,d,t,k} := \Delta \alpha_k \sum_{s:(t-s)\Delta \in I_k} x_{p,d,s}.$$

Then by sampling (5.3) via integration over intervals of length Δ, we have

$$\log \lambda_{p,o,t} = \Delta b_{p,o} + \sum_{d \in \mathcal{D}} \sum_{k=0}^{K-1} w_{o,d,k} z_{p,d,t,k}. \tag{5.4}$$

The total influence of drug $d \in \mathcal{D}$ on outcome $o \in \mathcal{O}$ can be measured by $\sum_{k=0}^{K-1} w_{o,d,k}$.

Note that the weights are independent of the patient p and the times t, so that within a collection of EHRs, we have a large number of training samples that can be used to infer the weights. Also note that the sufficient statistics of the data, $z_{p,d,t,k}$, are simple functions of the data and independent of outcome o. Hence these statistics can be pre-computed once and used for all outcomes of interest.

Note that (5.4) is a generalization of a log-linear Poisson autoregressive processes (Zhu and Wang, 2011), for which Hall et al. (2016) have recently derived sample complexity bounds.

5.2.3 Comparing the two models

Contrasting the classical model in (5.1) and our proposed Hawkes model with dyadic influence functions in (5.4), we see that both model the log of the event rate $\lambda_{p,o,t}$ as a linear combination of sufficient statistics of the past data (either the $\tilde{x}$'s or the z's, respectively). Despite this superficial similarity, the models exhibit

very different behaviors. In particular, the $\tilde{x}$'s in (5.1) can be thought of as the collection of observed events plus *artificial, simulated* events injected into the model. In particular, we can think of every day when a patient is taking a drug but the drug is not noted that day in the EHR as *missing data*. The MSCCS approach essentially imputes values for the missing data by assuming all people are taking all drugs for the same amount of time. Clearly this imputation is inaccurate, and these inaccuracies can bias inference of which drugs are causing which ADRs.

In contrast, the Hawkes model in (5.4) does not require us to explicitly impute missing data. The idea is that different drugs may have different impacts after different delays after the initial prescription, and different potential delays are captured by the different ϕ_ks. In effect, when we learn the parameters $\{w_{o,d,k}\}_{\substack{o\in\mathcal{O},d\in\mathcal{D}\\ k\in\{0,\dots,K-1\}}}$, we are learning the strength of the impact of drug d when the time since it was prescribed is on the order of 2^k. Thus this model is more flexible than the MSCCS model.

Note that (5.1) is similar (but not equivalent to) (5.4) for $K = 1$ if the same value of L is used. In particular, if a patient was prescribed a drug multiple times in the past L hours, then MSCCS would treat this a single drug occurrence in the time-at-risk window. In contrast, the Hawkes model suggests the multiple prescriptions have a cumulative effect. Since the number of prescriptions within a time-at-risk window L is generally small, these models can have similar empirical performances for $K = 1$.

Note that the number of weights to be inferred in (5.1) is equal to the number of drugs being evaluated. The number of weights to be inferred in our Hawkes model (5.4) is equal to the product of the number of drugs, $|\mathcal{D}|$, and K, the number of different influence functions in the model. Thus while using the Hawkes process with multiple influence functions can reduce bias in estimating ADRs, (5.4) has a larger (by a factor of K) parameter space than (5.1). We adjust for this larger parameter space in our inference method by using sparsity regularization, as described below.

5.3 Inference approach

Let $\mathbf{b} \coloneqq (b_{p,o})_{p\in\{1,\dots,P\},\, o\in\mathcal{O}}$ and $\boldsymbol{w} \coloneqq (w_{o,d,k})_{o\in\mathcal{O},d\in\mathcal{D},\, k\in\{0,\dots,K-1\}}$ denote the model parameters. (The model parameters for MSCCS can be represented this way with $K = 1$.) Using the Poisson likelihood in (5.2), we have that the negative log-likelihood of patient p's occurrences of outcome o is proportional to

$$\ell_{p,o,t}(b_{p,o}, \boldsymbol{w}) \coloneqq \lambda_{p,o,t} - x_{p,o,t} \log \lambda_{p,o,t} \tag{5.5}$$

We define the average negative log-likelihood over all patients as:

$$\ell(\mathbf{b}, \boldsymbol{w}) = \frac{1}{P} \sum_{p=1}^{P} \sum_{o\in\mathcal{O}} \sum_{t} \ell_{p,o,t}(b_{p,o}, \boldsymbol{w}).$$

Note that in our Hawkes model (5.4), $\ell_{p,o,t}$ is piecewise constant over t, so the log-likelihood can be efficiently computed via data squashing (Madigan et al., 2002; Simpson et al., 2013). In order to avoid overfitting and obtain an interpretable result, we induce sparsity by adding an ℓ_1 (LASSO) penalty (Tibshirani, 1996) on $\boldsymbol{w}$, resulting in the following optimization problem:

$$\min_{\mathbf{b},\boldsymbol{w}} \ell(\mathbf{b}, \boldsymbol{w}) + \lambda \|\boldsymbol{w}\|_1, \tag{5.6}$$

where $\lambda > 0$ is a tuning parameter controlling the level of sparsity.

The objective function in (5.6) is convex and can be minimized using a variety of approaches (*cf.* Wright et al. (2009)). Empirically we find that alternating between minimizing $\mathbf{b}$ (which has a closed form solution) and updating $\boldsymbol{w}$ using FISTA (Beck and Teboulle, 2009) yields fast convergence and quickly computable updates.

5.4 Experiments

5.4.1 OMOP task

To evaluate methods for ADR discovery, OMOP established a challenge problem of ranking drug-outcome pairs as possible ADRs. From ten different drugs and ten different outcomes, 53 drug-outcome pairs are labeled by OMOP as ground-truth true or false ADRs based on information on drug labels, for example calling warfarin-bleeding and ACE inhibitor-angioedema true pairs while calling ACE inhibitor-bleeding a false pair. From this ground truth, any algorithm that can rank drug-condition pairs from most- to least-likely ADRs can be evaluated via an ROC curve. This task is difficult because many ADRs are (thankfully) rare, in addition to all the ordinary challenges of causal discovery from LEDs, such as confounding by other measured or unmeasured variables, which may also vary over time.

5.4.2 Data description

We employ a de-identified version of Marshfield Clinic health system's EHR, which has been used for clinical care since the mid 1980s, serving primary, secondary, and tertiary care clinicians throughout Central and Northern Wisconsin (Powell et al., 2012). The system uses a variety of data gathering techniques to capture and code patient encounter information including diagnoses, laboratory results, procedures, medications, and vital sign measurements such as height, weight, blood pressures, etc. This longitudinal data is linked for each patient and exists in electronic form back to the early 1960's. Data consist of date-stamped events such as diagnosis codes and drug prescriptions; dates are encoded as patient age in 1/1000 years, for privacy reasons.

We extract ten drug prescription records and ten diagnosis records from the de-identified EHRs according to the definitions of the vocabularies used in the OMOP ground truth. We admit a patient into the cohort if the length of the observation for the patient is at least three months. The resulting cohort contains 327,824 patients with 1,940,681 adverse health outcome occurrences and 11,211,769 drug

prescription records. The average observation duration for the patients in our cohort is 9.1 years. Following the design of MSCCS, we restrict our attention to patients with at least one occurrence of the outcome o when we are inferring the weight for that outcome.

5.4.3 Metrics

Since the log-likelihood for both models is separable across different health outcomes, the influence is not directly comparable among different outcomes. We define the normalized score $S_{o,d}$ for each drug-outcome pair in MSCCS and the Hawkes process model as following:

$$S_{o,d} = \frac{w_{o,d}}{\sqrt{\sum_{d \in \mathcal{D}} w_{o,d}^2}} \text{ in MSCCS}, \quad S_{o,d} = \frac{\sum_{k=0}^{K-1} w_{o,d,k}}{\sqrt{\sum_{d \in \mathcal{D}} \left(\sum_{k=0}^{K-1} w_{o,d,k} \right)^2}} \text{ in Hawkes.}$$

For quantitative metrics, we report the area under the curve (AUC) of receiver operating characteristics (ROC) using the OMOP ground truth and the scores defined above.

5.4.4 Evaluation

To choose the shrinkage parameter λ for both MSCCS and the Hawkes process model, we perform leave-one-condition-out cross-validation (LOCOCV): for each of the ten outcomes, we adaptively pick $\lambda \in \{0, 10^{-8}, 10^{-7}, 10^{-6}\}$ that perform the best on the other nine conditions.

Figure 5.2 presents the AUC of MSCCS and Hawkes with various K for L $\in$ {three months, six months, one year}. Note that for fixed L, both MSCCS and Hawkes make use of the information in the past L hours to model the occurrence of adverse health outcomes at each time. We observe that the Hawkes process model consistently outperforms MSCCS when more then one influence function is used, effectively indicating that modeling drug-dependent time-at-risk windows (captured in a data-dependent manner by the Hawkes model) is beneficial to ADR discovery.

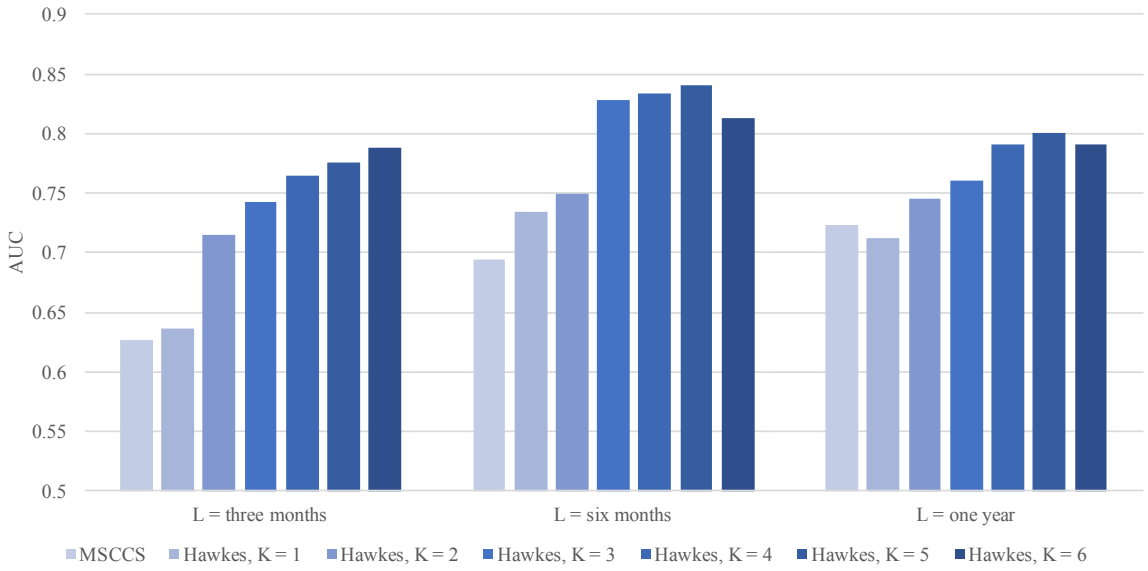


Figure 5.2: AUC for MSCCS and Hawkes models with various L and K.

In the literature of ADR discovery from LEDs (Norén et al., 2013; Simpson et al., 2013; Suchard et al., 2013b; Schuemie et al., 2016), different methods are compared under their best settings. To test the highest AUC for both models, we vary L from 22 days to ten years and K from 1 to 7. The best performers of MSCCS reaches an AUC of 0.7449 at L = four years, while the Hawkes process model reaches its best AUC of 0.8409 at K = 5, L = six months. To demonstrate how well the Hawkes process model and MSCCS can predict unseen adverse drug reactions in practice, we perform LOCOCV to adaptively and jointly pick λ, L and K. The AUC after LOCOCV for MSCCS is 0.6970, while the AUC after LOCOCV for Hawkes is 0.8258, indicating that the expressive power of the the Hawkes process model better coincides with the ADR signals encoded in the data.

Figure 5.3 shows the rank of the true ADR-causing drug among all ten drugs for each of the nine true ADR pairs. Rank of one means the true ADR-causing drug is assigned the highest score among all ten drugs by the method. Notice that the eighth and ninth pairs are both associated with the same outcome, so ranking one true causing drug to the first place and the other true causing drug to the second

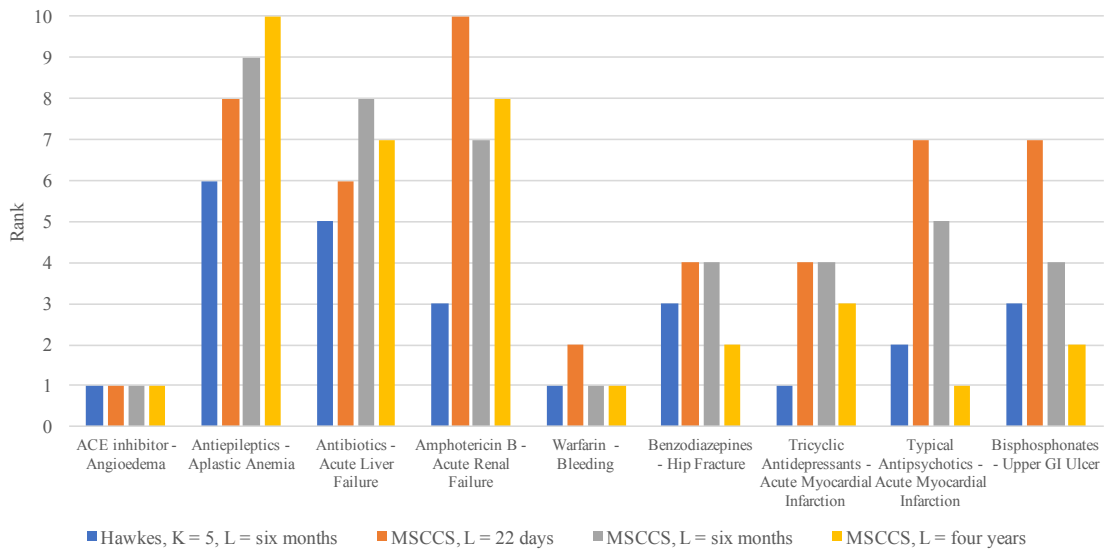


Figure 5.3: Rank of true ADR-causing drug among all ten drugs for each true ADR pair.

place is the best one can do. We observe that although MSCCS with L = four years performs reasonably well on the first pair and the last five pairs, it completely fails to discover the second true ADR pair. Actually, it even assigns a negative score to this true ADR pair, suggesting that the true causing drug inhibits the occurrence of the adverse outcome. On the other hand, MSCCS with L = 22 days attains better performance on the second and third pairs, but it cannot successfully learn the eighth and ninth true ADR pair due to the limitation of a short time-at-risk window. By using different influence functions, the Hawkes process model is able to better capture these long-term and short-term effects jointly and this results in an overall performance superior to MSCCS.

5.5 Discussion

We have proposed a log-linear Hawkes process model of adverse drug reactions with longitudinal event data. Compared with the leading approach, multiple

self-controlled case series, for ADR discovery with LEDs, the proposed method offers tremendous flexibility in modeling time-varying effects of various drugs on the occurrence of adverse health outcomes. Experimental results demonstrate the superior performance of the proposed method over MSCCS in various experiment settings.

Notice that in our experiments, the increase of the time-at-risk window and the number of influence functions used in the models does not necessarily correspond to the improvement of ADR discovery performance. A reasonable explanation is that with prolonged time-at-risk windows, long-term fluctuation of the baseline occurrence rate of an adverse health outcome also needs to be taken into consideration. However, in the current modeling framework, for efficiency we only use a patient-specific yet time-invariant parameterization to model the baseline occurrence rate of an adverse health outcome. Therefore, incorporating time-varying baseline (Kuang et al., 2016a) to distinguish between baseline fluctuation and time-varying drug effects would be an important future research direction. Other future directions include improving the efficiency of model fitting via parallelism and stochasticity as well as designing different kernels to facilitate the incorporation of different clinical hypotheses.

6 A MACHINE-LEARNING BASED DRUG REPURPOSING APPROACH USING BASELINE REGULARIZATION

6.1 Introduction

With the increasing availability of electronic health record (EHR) data, there is an emerging interest in using EHRs from various patients for computational drug repurposing (CDR). Specifically, in EHRs, drug prescriptions of various drugs are recorded throughout time for various patients. In the same time, numeric physical measurements, such as fasting blood glucose (FBG) level, blood pressure, and low density lipoprotein are also recorded. By designing machine learning algorithms that can establish relationships between the occurrences of prescriptions of some particular drugs and the increase or the decrease in the values of some particular numeric physical measurements, we might be able to identify drugs that can be potentially repurposed to control certain numeric physical measurements. This chapter describes such a machine learning algorithm called *baseline regularization* (Kuang et al., 2016a) for CDR, where we also learn lessons from Bao et al. (2017a) to handle the data irregularity issue for higher causal fidelity.

6.2 Materials

Figure 6.1 visualizes a set of electronic health records from two patients. Drug prescriptions of different types enter the EHRs of the two patients at different times. Fasting blood glucose (FBG) level measurements are also recorded at various times. In this chapter, we will consider how to identify drugs that can be potentially repurposed to control FBG level as an example to illustrate the use of baseline regularization. The idea is to formulate this problem as a machine learning problem by considering an FBG record as a response variable and using the drug prescriptions that occur before the FBG record as features to predict the value of the FBG record. If through the predictive model we notice that the prescription of a particular drug

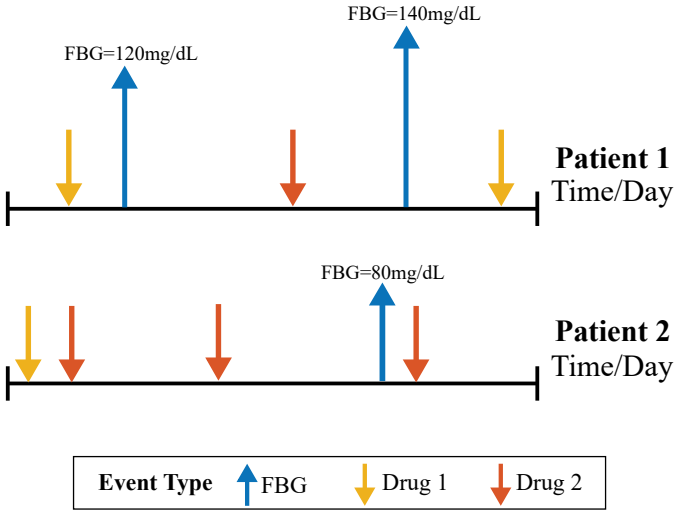


Figure 6.1: Visualization of electronic health records (EHRs) from two patients. Fasting blood glucose (FBG) level measurements as well as drug prescriptions of various drugs are observed for the two patients over time.

is associated with the decrease of FBG, then we can consider this drug as a potential candidate to be repurposed for glucose control. It should be noticed that while we are using FBG level control as an example for the ease of presentation, the proposed algorithm can also be used to identify drugs that can be potentially repurposed to control other numeric physical measurements.

6.2.1 Notation

Without loss of generality, we assume that only drug prescription records and FBG records are available for each patient. And we consider only patients with at least one FBG record throughout their observations. Let there be N patients and p drugs under consideration in total. Suppose that for the i^{th} patient, there are n_i drug prescription records and m_i FBG records in total, where $i \in \{1, 2, \ldots, N\}$. We can use a 2-tuple (x_{ij}, t_{ij}) to represent the j^{th} drug prescription record of the i^{th} patient, where $j \in \{1, 2, \ldots, n_i\}$, $x_{ij} \in \{1, 2, \ldots, p\}$ represents which drug among the p drugs is prescribed, and t_{ij} represents the timestamp of the drug prescription. Similarly, we can also use a 2-tuple (y_{ik}, τ_{ik}) to represent the k^{th} FBG measurement record

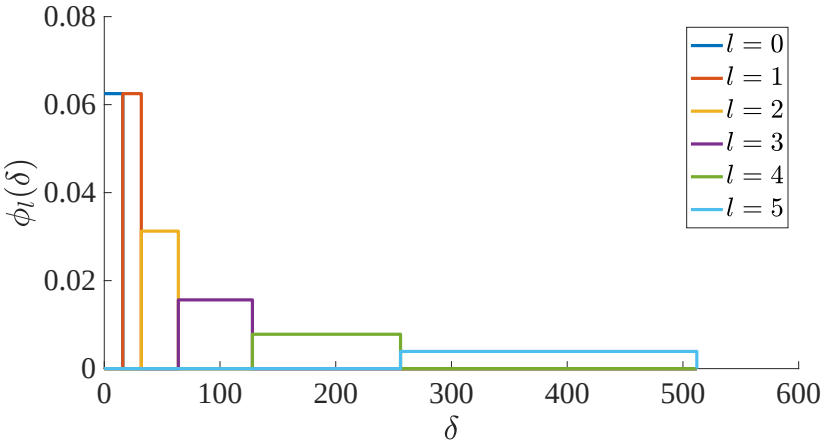


Figure 6.2: Dyadic influence functions for $S = 512$ and $L = 6$.

from the i^{th} patient, where $k \in \{1, 2, \ldots, m_i\}$, y_{ik} denotes the value of the FBG measurement, and τ_{ik} represents the measurement timestamp. Note that given i, $t_{i1} \leqslant t_{i2} \leqslant \cdots \leqslant t_{in_i}$ and $\tau_{i1} \leqslant \tau_{i2} \leqslant \cdots \leqslant \tau_{in_i}$. In this way, we can represent the EHR of each patient as a set of the aforementioned 2-tuples.

6.3 Methods

We first present how the potential influence of various drugs over time on the value of FBG measurements can be ascertained via the use of dyadic influence functions, directly from raw EHR data. We then present our baseline regularization model that combines the effects of time-varying patient-specific baselines and the effects from various drugs throughout time to predict FBG levels for CDR.

6.3.1 Dyadic Influence

We assume that drug prescriptions in the EHR of a patient have certain influences on the values of the FBG measurements that occur after the prescriptions. Since drug prescriptions occur throughout time for various patients, given an FBG measurement record, an intuition is that a drug prescription record that occurs long

before has less effect, if any, on the value of the FBG measurement in question, compared with a more recent drug prescription occurrence. Based on this intuition, for $t_{ij} \leqslant \tau_{ik}$, we represent the effect of a drug prescription (x_{ij}, t_{ij}) on an FBG measurement (y_{ik}, τ_{ik}) through a weighted sum of a pre-defined set of dyadic influence functions $\{\phi_l(\cdot)\}_{l=0}^{L-1}$ (Bao et al., 2017a). Specifically, let $S > 0$ and $L \in \mathbb{N}^+$ be given. For $l \in \{0, 1, 2, \ldots, L-1\}$, we define

$$\alpha_l \triangleq \begin{cases} 2^{L-1}/S, & l = 0 \\ 2^{L-l}/S, & l = 1, 2, \ldots, L-1 \end{cases};$$

and the half-closed-half-open intervals,

$$I_l \triangleq \begin{cases} [0, 1/\alpha_l), & l = 0 \\ [1/\alpha_l, 2/\alpha_l), & l = 1, 2, \ldots, L-1 \end{cases}.$$

Then we define

$$\phi_l(\delta) \triangleq \alpha_l \mathbb{I}(\delta \in I_l),$$

where $\delta = \tau_{ik} - t_{ij}$ is the time difference between the drug prescription and the FBG measurement, and $\mathbb{I}(\cdot)$ is the indicator function. Note that these $\phi_l(\cdot)$'s all integrate to one and are orthogonal to one another.

Figure 6.2 visualizes the set of dyadic influence functions when $S = 512$ and $L = 6$. As can be seen, when the time difference between two events δ increases, the influence decays in exponential order. For $\delta \geqslant S$, the previous drug prescription is assumed not to have any influence on the value of the FBG measurement in question. Dyadic influence functions provide a flexible approach to ascertain influences of various drug prescriptions in the past on the value of FBG measurement records. This is in contrast to the drug era construction that is prevalent in the pharmacovigilance literature (Nadkarni, 2010; Simpson et al., 2013; Ryan, 2015; Kuang et al., 2017c), where ad-hoc heuristics are used to generate a consecutive time period during which the value of an FBG measurement is assumed to be under unattenuated influence.

6.3.2 Baseline Regularization

Baseline regularization assumes that an observed FBG value is due to the influences of various drug prescriptions that occur in the past as well as a hidden, intrinsic baseline FBG value that represents the FBG level that would have been observed if the patient were not under any other influences. Specifically, baseline regularization considers solving the optimization problem in (6.1):

$$\begin{aligned}\hat{\mathbf{b}}, \hat{\boldsymbol{\beta}} \triangleq \arg\min_{\mathbf{b},\boldsymbol{\beta}} \frac{1}{2M} \sum_{i=1}^{N} \sum_{k=1}^{m_i} \left(y_{ik} - b_{ik} - \sum_{j=1}^{n_i} \sum_{q=1}^{p} \sum_{l=0}^{L-1} \beta_{ql} \phi_l(\tau_{ik} - t_{ij}) \cdot \mathbb{I}(x_{ij} = q) \right)^2 \\ + \lambda_1 \sum_{i=1}^{N} \sum_{k=1}^{m_i - 1} \left| b_{ik} - b_{i(k+1)} \right| + \lambda_2 \|\boldsymbol{\beta}\|_1,\end{aligned} \tag{6.1}$$

where $M = \sum_{i=1}^{N} m_i$ is the total number of FBG measurements under consideration, $\lambda_1 > 0$ and $\lambda_2 > 0$ are regularization parameters, and

$$\mathbf{b} \triangleq \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m_1} & \cdots & b_{N1} & b_{N2} & \cdots & b_{Nm_N} \end{bmatrix}^\top \text{ and}$$
$$\boldsymbol{\beta} \triangleq \begin{bmatrix} \beta_{10} & b_{11} & \cdots & \beta_{1(L-1)} & \cdots & \beta_{p0} & \beta_{p1} & \cdots & \beta_{p(L-1)} \end{bmatrix}^\top$$

are the parameters that we need to estimate. The baseline regularization problem is a regularized least square problem with a fused lasso penalty (controlled by λ_1) and a lasso penalty (controlled by λ_2).

The parameter $\mathbf{b}$ is a baseline parameter vector whose components represent the potentially different baseline FBG levels throughout time for different patients. Such time-varying and patient specific baselines are of great importance to provide flexibility to describe the intricate data generation process in reality. For example, diabetic patients tend to have higher FBG levels compared to a healthy person. Therefore, the fact that the baselines used are patient-specific helps to model such heterogeneity among different individuals in the data. Even for a particular patient, the FBG levels can also change dramatically over the years as the patient

ages. Therefore, the time-varying nature of the baseline parameters also helps to capture the heterogeneity of the FBG levels over time. The baseline parameter $\mathbf{b}$ is regularized by a fused lasso penalty, without which $\mathbf{b}$ is flexible enough to explain any given FBG level observations. The intuition of using a fused lasso penalty is to minimize the difference between two adjacent baseline parameters. Since baseline parameters represent the FBG values that would have been observed if the patient were not under other influences, it is reasonable to assume that these baseline values are usually relatively stable over a certain period of time, and hence we encourage such stability via the use of fused lasso penalties.

The parameter $\boldsymbol{\beta}$ represents the effects of every drug on the value of the FBG level depending on the time difference between the drug prescription and the FBG measurement. A lasso penalty is used to encourage sparsity over the effect parameter $\boldsymbol{\beta}$ as we assume that only a small portion of drugs can have some effect on the value of an FBG measurement during a certain period of time.

The least square objective is hence to minimize the differences between the observed FBG values and the values given by the model that take into consideration both the time-varying patient-specific baseline parameters that change stably and the sparse effect parameters that describe effects of various drugs during various periods of time.

For the q^{th} drug, let $\left\{\hat{\beta}_{q0}, \hat{\beta}_{q1}, \hat{\beta}_{q2}, \ldots, \hat{\beta}_{q(L-1)}\right\}$ be the set of effects learned from the baseline regularization model. We measure the overall effect of o_q on the FBG level as the average of the elements in the set: $o_q \triangleq \frac{1}{L}\sum_{l=0}^{L-1}\hat{\beta}_{ql}$.

Algorithm 2 Baseline Regularization

Require: $\mathbf{y}$, $\mathbf{Z}$, $\mathbf{D}$, λ_1,and λ_2.
Ensure: $\hat{\mathbf{b}}$ and $\hat{\boldsymbol{\beta}}$.
1: Initialize $\boldsymbol{\beta}^{(0)}$.
2: $u \leftarrow 0$.
3: **while** true **do**
4: $\quad \check{\mathbf{y}}^{(u+1)} \leftarrow \mathbf{y} - \mathbf{Z}\boldsymbol{\beta}^{(u)}$.
5: $\quad \mathbf{b}^{(u+1)} \leftarrow \arg\min_{\mathbf{b}} \frac{1}{2M}\|\check{\mathbf{y}}^{(u+1)} - \mathbf{b}\|_2^2 + \lambda_1\|\mathbf{D}\mathbf{b}\|_1$. ▷ $\mathbf{b}$-step
6: $\quad \tilde{\mathbf{y}}^{(u+1)} \leftarrow \mathbf{y} - \mathbf{b}^{(u+1)}$.
7: $\quad \boldsymbol{\beta}^{(u+1)} \leftarrow \arg\min_{\boldsymbol{\beta}} \frac{1}{2M}\|\tilde{\mathbf{y}}^{(u+1)} - \mathbf{Z}\boldsymbol{\beta}\|_2^2 + \lambda_2\|\boldsymbol{\beta}\|_1$. ▷ $\boldsymbol{\beta}$-step
8: $\quad$ **if** Stopping criteria met **then**
9: $\quad\quad \hat{\mathbf{b}} \leftarrow \mathbf{b}^{(u+1)}$ and $\hat{\boldsymbol{\beta}} \leftarrow \boldsymbol{\beta}^{(u+1)}$.
10: $\quad\quad$ **return** $\hat{\mathbf{b}}$ and $\hat{\boldsymbol{\beta}}$.
11: $\quad$ **else**
12: $\quad\quad u \leftarrow u + 1$.
13: $\quad$ **end if**
14: **end while**

6.3.3 Optimization for Baseline Regularization

The baseline regularization problem in (6.1) is a convex optimization problem. Furthermore, $\mathbf{b}$ and $\boldsymbol{\beta}$ are separable in the optimization problem. Therefore, we can perform a blockwise minimization procedure that alternates between the minimization of $\mathbf{b}$ and $\boldsymbol{\beta}$ to achieve optimality (Tseng, 2001). When $\mathbf{b}$ is fixed, the optimization problem with respect to $\boldsymbol{\beta}$ is a lasso linear regression problem (Tibshirani, 1996). When $\boldsymbol{\beta}$ is fixed, the optimization problem with respect to $\mathbf{b}$ is a blockwise fused lasso signal approximator problem (Tibshirani and Taylor, 2011). Both problems can be solved efficiently. The blockwise minimization algorithm is summarized in Algorithm 2. To see the two subproblems, let

$$z_{iql} \triangleq \sum_{j=1}^{n_i} \phi_l(\tau_{ik} - t_{ij}) \cdot \mathbb{I}(x_{ij} = q).$$

Then (6.1) can be rewritten as:

$$\hat{\mathbf{b}}, \hat{\boldsymbol{\beta}} \triangleq \arg\min_{\mathbf{b},\boldsymbol{\beta}} \frac{1}{2M}\|\mathbf{y} - \mathbf{b} - \mathbf{Z}\boldsymbol{\beta}\|_2^2 + \lambda_1\|\mathbf{D}\mathbf{b}\|_1 + \lambda_2\|\boldsymbol{\beta}\|_1, \tag{6.2}$$

where

$$\mathbf{y} \triangleq \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1m_1} & \cdots & y_{N1} & y_{N2} & \cdots & y_{Nm_N} \end{bmatrix}^\top,$$

$\mathbf{Z}$ is an $M \times (p \times L)$ data matrix whose i^{th} row is:

$$\begin{bmatrix} z_{i10} & z_{i11} & \cdots z_{i1(L-1)} & \cdots & z_{ip0} & z_{ip1} & \cdots z_{ip(L-1)} \end{bmatrix},$$

and $\mathbf{D}$ is the blockwise first difference matrix:

$$\mathbf{D} \triangleq \begin{bmatrix} \mathbf{D}_{m_1} & & & \\ & \mathbf{D}_{m_2} & & \\ & & \ddots & \\ & & & \mathbf{D}_{m_N} \end{bmatrix},$$

with an $(m-1) \times m$ first difference matrix defined as $\mathbf{D}_1 = 0$ and:

$$\mathbf{D}_m \triangleq \begin{bmatrix} -1 & 1 & & \\ & -1 & 1 & \\ & & \ddots & \\ & & -1 & 1 \end{bmatrix}.$$

Therefore, from (6.2), when $\boldsymbol{\beta}$ is fixed, let $\check{\mathbf{y}} \triangleq \mathbf{y} - \mathbf{Z}\boldsymbol{\beta}$; then the blockwise fused lasso signal approximator problem with respect to $\mathbf{b}$ is:

$$\arg\min_{\mathbf{b}} \frac{1}{2M}\|\check{\mathbf{y}} - \mathbf{b}\|_2^2 + \lambda_1\|\mathbf{D}\mathbf{b}\|_1.$$

On the other hand, from (6.2), when $\mathbf{b}$ is fixed, let $\tilde{\mathbf{y}} \triangleq \mathbf{y} - \mathbf{b}$, then the lasso linear regression problem with respect to $\boldsymbol{\beta}$ is:

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2M} \|\tilde{\mathbf{y}} - \mathbf{Z}\boldsymbol{\beta}\|_2^2 + \lambda_2 \|\boldsymbol{\beta}\|_1. \tag{6.3}$$

In Algorithm 2 the two most computationally-intensive steps are Step 5 and Step 7. The former involves solving a fused lasso signal approximator problem, whose solution can be computed exactly by the dynamic programming algorithm proposed in Johnson (2013). The latter involves solving a lasso linear regression problem, which is achieved by the cyclic coordinate descent algorithm with variable screening proposed in Friedman et al. (2010) and Tibshirani et al. (2012).

6.4 Results

To demonstrate the utility of baseline regularization, we run our algorithm on the Marshfield Clinic EHR to identify drugs that can be potentially used to control FBG level. We consider patients with at least one FBG measurement throughout their observations. This leads to a total number of 333,907 FBG measurements from 75,146 patients.

To ascertain influences from drug prescriptions, we choose S to be half a year and $L = 5$ for the dyadic influence function. We only consider drugs that have at least one drug prescription that is at most S amount of time prior to the occurrence of at least one FBG measurement, yielding a total number of 5147 different drugs for consideration. λ_1 and λ_2 are chosen such that roughly 200 drugs will be selected eventually by the model. This is because we do not know in advance whether the drugs returned by the algorithm could potentially control FBG level or not, and we need to examine the findings of the algorithm manually. Therefore, the regularization parameters need to be carefully chosen so that the number of drugs selected by the model can be feasibly examined. Table 6.1 reports the top thirty drugs ranked by their overall effects among the 180 drugs generated by the baseline regularization using $\lambda_1 = 86$ and $\lambda_2 = 2.841977 \times 10^{-4}$. For more information about

Table 6.1: Top thirty drugs selected by baseline regularization associated with FBG decrease.

INDX	CODE	DRUG NAME	SCORE
1	4132	GLUCOPHAGE	-82.388
2	7470	PIOGLITAZONE HCL	-36.869
3	8437	ROSIGLITAZONE MALEATE	-29.046
4	5786	METFORMIN	-18.867
5	4184	GLYBURIDE	-16.664
6	6382	NEEDLES INSULIN DISPOSABLE	-15.233
7	5787	METFORMIN HCL	-9.910
8	4806	INSULIN GLARGINE HUM.REC.ANLOG	-8.523
9	4497	HUM INSULIN NPH/REG INSULIN HM	-7.336
10	160	ACTOS	-6.006
11	7768	PREMARIN	-4.879
12	4106	GLIMEPIRIDE	-4.028
13	6656	NPH HUMAN INSULIN ISOPHANE	-3.613
14	4971	ISOSORBIDE MONONITRATE	-3.229
15	4561	HYDROCORTISONE	-3.084
16	4107	GLIPIZIDE	-3.007
17	9379	THIAMINE HCL	-2.968
18	1573	CAPTOPRIL	-2.871
19	5368	LIPITOR	-2.819
20	9152	SYRING W-NDL DISP INSUL 0.5ML	-2.380
21	1988	CIPROFLOXACIN HCL	-2.367
22	3937	FOSINOPRIL SODIUM	-2.252
23	5390	LISINOPRIL	-2.004
24	9994	VERAPAMIL HCL	-1.965
25	1216	BLOOD SUGAR DIAGNOSTIC	-1.900
26	7760	PREGABALIN	-1.708
27	6803	ONDANSETRON HCL	-1.678
28	4970	ISOSORBIDE DINITRATE	-1.575
29	6540	NITROGLYCERIN	-1.496
30	5571	MAGNESIUM	-1.266

choosing the regularization parameters, please see Section 6.5.

As shown in Table 6.1, the drugs in green are drugs that are prescribed to control blood sugar level. The drugs in white are not normally used to control blood sugar level. However, there might be some potentially interesting findings based on a literature review. For example, thiamine HCL is reported to reduce the adverse effect of hyperglycemia by inhibiting certain biological pathways (vinh quoc Luong and Nguyen, 2012), and deficiency of thiamine is observed in diabetic patients (Page et al., 2011). Ciprofloxacin HCL could lead to hypoglycemia, according to the medication guide from the Food and Drug Administration (FDA) (FDA, a). Lisinopril is also associated with hypoglycemia, according to the drug label from the FDA (FDA, d). Verapamil HCL is reported to decrease blood sugar level as well as to have some hope in preventing pancreatic β cell loss. Such a loss is considered a pathological characteristic for diabetes (Poudel and Kafle, 2017). Cases of hypoglycemia associated with the use of pregabalin have been reported (Abe et al., 2015; Raman, 2016). Premarin, fosinopril sodium, and hydrocortisone are potential false positives for our method, since they have been linked to hyperglycemia (DiabetesInControl, 2015). Drugs with mixed evidence are also found. For example, according to DiabetesInControl (2015), both Lipitor and captopril are linked to hyperglycemia. Studies that suggest otherwise are also seen in the literature (FDA, b; Girardin and Raccah, 1998; Neerati and Gade, 2011).

The baseline regularization algorithm is implemented with `R`. The blockwise fused lasso signal approximator problem is solved using a subroutine in the `R` package `glmgen` (Arnold et al., 2014). The lasso linear regression problem is solved using the `R` package `glmnet` (Friedman et al., 2009).

6.5 Notes

6.5.1 Splitting Patient Records

In (6.1), we try to control the differences between two adjacent baseline parameters via the use of the fused lasso penalty. Consider the pair b_{ik} and $b_{i(k+1)}$ that indicates

the baseline FBG levels corresponding to two adjacent physical measurements. Although the two measurements are adjacent to each other in time, the actual time difference between the two measurements could be large, i.e. $\tau_{ik} \ll \tau_{i(k+1)}$. In this case, it might not be reasonable to regularize the difference between the two baselines as the FBG level could go through substantial changes during such a long period of time. Therefore, we consider splitting the records from the same patient into various subsets within which the records are close to each other in time, and just regularize the differences between adjacent baselines within the same subset. It remains to determine how far apart two adjacent records should be for us to consider them belonging to distinct subsets. We take a data-driven approach to determine this threshold. In detail, we compute the time differences of all adjacent pairs of FBG measurements for all patients. We then use Tukey's method of outlier identification (Tukey, 1977) to determine the smallest outlier. The distribution of the differences is heavy-tailed, and most of the differences are small. Therefore, the smallest outlier is a relatively large time difference value, and we set this value as our threshold. After splitting the FBG records of a patient into various subsets, each subset of FBG records can be considered as data from an independent patient. Therefore, the previously established formulation of the baseline regularization model can be naturally extended to handle this situation by simply modifying $\mathbf{D}$ in (6.2) accordingly. The threshold value identified in our dataset is 4.1 years.

6.5.2 Model Selection

Since in CDR, we do not know a priori what drugs returned by the algorithm can actually decrease or increase FBG levels, we manually review the drug list to identify potential repurposing opportunities. Therefore, model selection for baseline regularization not only needs to identify a model that explains the data well but also needs to generate a drug list of moderate size so that subsequent reviewing efforts are feasible.

To determine an appropriate λ_1, we start from identifying the minimum λ_1^* such that all the baseline parameters are fused to its average in the following fused lasso

signal approximator problem, where we only use the baseline parameter $\mathbf{b}$ to model the FBG measurements $\mathbf{y}$:

$$\arg\min_{b} \frac{1}{2M}\|\mathbf{y}-\mathbf{b}\|_2^2+\lambda_1\|\mathbf{D}\mathbf{b}\|_1.$$

Define $\mathbf{T}_m$ as an $m\times m$ upper triangular matrix whose upper part and the diagonal are all ones, and whose entries are otherwise zeros. Then according to Wang et al. (2015),

$$\lambda_1^*=\max_{i\in\{1,2,\dots,N\}}\|\mathbf{T}_{m_i}\left(\mathbf{y}_i-\bar{y}_i\mathbf{1}_{m_i}\right)\|_\infty, \tag{6.4}$$

where $\mathbf{1}_m$ is an $m\times 1$ vector of all ones, and $\bar{y}_i$ is the mean of all the FBG measurements from the i^{th} patient. Upon the determination of λ_1^* in (6.4), we can choose $\lambda_1=\gamma\lambda_1^*$, where $\gamma\in(0,1)$ can vary to generate different models. The results reported in Table 6.1 are given by $\lambda_1=0.05\lambda_1^*$.

To determine an appropriate λ_2, we first solve for the pathwise solution to a continuous self-controlled case series (CSCCS) problem (Kuang et al., 2016c), which is a lasso linear regression problem assuming a fixed baseline parameter for each patient:

$$\arg\min_{\beta} \frac{1}{2M}\|\mathbf{y}-\mathbf{U}\bar{\mathbf{y}}-(\mathbf{X}-\mathbf{U}\bar{\mathbf{Z}})\boldsymbol{\beta}\|_2^2+\lambda_2\|\boldsymbol{\beta}\|_1,$$

where

$$\mathbf{U}\triangleq\begin{bmatrix}\mathbf{1}_{m_1} & & & \\ & \mathbf{1}_{m_2} & & \\ & & \ddots & \\ & & & \mathbf{1}_{m_N}\end{bmatrix},\quad \bar{\mathbf{y}}\triangleq(\mathbf{U}^\top\mathbf{U})^{-1}\mathbf{U}^\top\mathbf{y},\quad \bar{\mathbf{Z}}\triangleq(\mathbf{U}^\top\mathbf{U})^{-1}\mathbf{U}^\top\mathbf{Z}.$$

In our experiments, we are aiming at selecting about 200 drugs in the end. Therefore, from the solution path, we choose an λ_2 whose solution selects about 250 drugs and we use this λ_2 for the baseline regularization problem. The solution to the CSCCS problem can also be used to initialize $\boldsymbol{\beta}^{(0)}$ in baseline regularization in Algorithm 2. Given the same λ_2, we notice that the baseline regularization problem

usually will select fewer drugs compared to the corresponding CSCCS problem. Intuitively, this is because the introduction of time-varying and patient-specific baseline parameters in the baseline regularization problem help to explain the changes in the FBG measurements better. Therefore, fewer drugs are needed in order to explain the changes of FBG levels in the dataset, yielding a sparser drug effect parameterization.

When multiple configurations of λ_1's and λ_2's are provided, we can use Akaike information criterion (AIC) or Bayesian information criterion (BIC) for model selection. The degree of freedom of the baseline regularization model needed in the calculation is the summation of the degree of freedom of the baseline parameter $\mathbf{b}$ and the degree of freedom of the drug effect parameter $\boldsymbol{\beta}$. The former is the total number of piecewise constant segments of $\mathbf{b}$ and the latter is the number of nonzero entries of $\boldsymbol{\beta}$.

Since the dimension of the parameterization in baseline regularization is larger than the sample size of the data, caution needs to be paid when we choose regularization parameters. Essentially, we would like to choose large λ_1 and λ_2 to impose strong regularization to avoid overfitting. The degree of freedom of the learned model also needs to be monitored and controlled so that it is smaller than the sample size of the data.

6.5.3 Stopping Criteria

Since the baseline regularization problem is a convex optimization problem, we can verify the convergence of the optimization procedure in Algorithm 2 by checking the violation of the Karush-Kuhn-Tucker (KKT) conditions of the current iterate. Since when $\boldsymbol{\beta}^{(u)}$ is given, the update to $\mathbf{b}^{(u+1)}$ can be carried out exactly by Step 4 and Step 5 of Algorithm 2, we are interested in knowing the violation due to $\mathbf{b}^{(u+1)}$ and $\boldsymbol{\beta}^{(u)}$ via the KKT conditions of (6.3):

$$\mathbf{s}^{(u)} = \frac{1}{n\lambda_2}\mathbf{Z}^\top \left(\mathbf{y} - \mathbf{b}^{(u+1)} - \mathbf{Z}\boldsymbol{\beta}^{(u)}\right),$$

where $\mathbf{s}^{(u)}$ is the subgradient of $\|\boldsymbol{\beta}\|_1$. If $\mathbf{b}^{(u+1)}$ and $\boldsymbol{\beta}^{(u)}$ are optimal, then

$$\hat{s}_d \begin{cases} = 1, & \beta_d^{(u)} > 0 \\ = -1, & \beta_d^{(u)} < 0 \\ \in [-1, 1], & \beta_d^{(u)} = 0 \end{cases}, \tag{6.5}$$

where $\hat{s}_d$ and $\beta_d^{(u)}$ are the d^{th} components of $\hat{\mathbf{s}}$ and $\boldsymbol{\beta}^{(u)}$, respectively. By measuring how much $\mathbf{s}^{(u)}$ violates the specification of $\hat{\mathbf{s}}$ in (6.5) via $\|\mathbf{v}^{(u)}\|_2$, where the d^{th} component of $\mathbf{v}^{(u)}$ is

$$v_d^{(u)} \triangleq \begin{cases} s_d^{(u)} - 1, & \beta_d^{(u)} > 0 \\ s_d^{(u)} + 1, & \beta_d^{(u)} < 0 \\ \max\{0, |s_d^u| - 1\}, & \beta_d^{(u)} = 0 \end{cases},$$

we know about how far away the current solution is to optimality. Such a measurement can be used as a stopping criterion. In our experiment, we set $\|\mathbf{v}^{(u)}\|_2 \leqslant 0.01$ as our stopping criterion.

6.6 Conclusion

We have presented an algorithm to predict the effects of drugs on numeric physical measurements in the EHR such as fasting blood glucose. Drugs with a strong effect to decrease the measurement are potential repurposing targets. Our method inherits from self-controlled case series (Kuang et al., 2016c) the ability to take into account inter-patient variation. By addition of a time-varying baseline it can also address intra-patient variation over time. And by use of dyadic influence functions it can avoid the need to decide drug eras and can model different effect times for different drugs.

Part IV

Interplay

Understanding the potential causal interplay among a broad spectrum of health-related factors encoded in clinical data is our goal in developing machine learning models and algorithms with high causal fidelity for EHR. To this end, an effective approach is to view the various variables collected in EHRs as following a multivariate distribution, modeled by a graphical model. Since EHR data can usually be represented as binary variables or count variables, undirected graphical models that represent multivariate binary/count distribution are of our particular interest in Part IV. Note that considering the learning of undirected graphical models is without loss of generality, because an undirected graphical model can serve as a template to construct Bayesian networks, as mentioned in Chapter 2.

In Chapter 7, we consider modeling LED via Poisson square root graphical models. We combine lessons learned previously, from addressing the inhomogeneity and irregularity challenges, with graphical models to meticulously model the interplay among various clinical event types, yielding improved performance for ADR discovery and providing evidence for the causal fidelity of our approaches.

In the aforementioned chapter, we take a nodewise regression approach to graphical model learning. While efficient, such an approach is susceptible to agnostic data generation-i.e. data that are not generated according to a graphical model-compared to maximum likelihood estimation. However, maximum likelihood learning of graphical models for binary and count data are notoriously difficult. To make progress, in Chapter 8, we consider a sampling-based maximum likelihood learning paradigm in the context of Ising models and derive more efficient procedure for the stochastic learning of Ising models. Furthermore, in Chapter 9, we deliver more efficient binary graphical model learning by deriving the first screening rule for maximum likelihood estimation of ℓ_1-regularized Ising models.

7 TEMPORAL POISSON SQUARE ROOT GRAPHICAL MODELS

7.1 Introduction

Longitudinal event data (LED) and the analytics challenges therein are ubiquitous now. In business analytics, purchasing events of different items from millions of customers are collected, and retailers are interested in how a distinct market action or the sales of one particular type of item could boost or hinder the sales of another type (Han et al., 2011). In search analytics, web search keywords from billions of web users are usually mapped into various topics (e.g. travel, education, weather), and search engine providers are interested in the interplay among these search topics for a better understanding of user preferences (Gunawardana et al., 2011). In health analytics, electronic health records (EHRs) contain clinical encounter events from millions of patients collected over decades, including drug prescriptions, biomarkers, and condition diagnoses, among others. Unraveling the relationships between different drugs and different conditions is vital to answering some of the most pressing medical and scientific questions such as drug-drug interaction detection (Tatonetti et al., 2012), comorbidity identification, adverse drug reaction (ADR) discovery (Simpson et al., 2013; Bao et al., 2017a; Kuang et al., 2017c), computational drug repositioning (Kuang et al., 2016a,c), and precision medicine (Liu et al., 2013a, 2014a).

All these analytics challenges raise the statistical modeling question: *can we offer a comprehensive perspective about the potential causal relationships among the occurrences of all possible pairs of event types in longitudinal event data?* In this chapter, we propose a solution via temporal Poisson square root graphical models (TPSQRs), a generalization of Poisson square root graphical models (PSQRs, Inouye et al. 2016) made in order to represent multivariate distributions among count variables evolving temporally in LED.

The reason why conventional undirected graphical models (UGMs) are not readily applicable to LED is the lack of mechanisms to address the *temporality* and *irregularity* in the data. Conventional UGMs (Liu and Page, 2013a; Liu et al., 2014b,

2015; Yang et al., 2015a; Liu et al., 2016; Kuang et al., 2017a; Geng* et al., 2018a) focus on estimating the co-occurrence relationships among various variables rather than their temporal relationships, that is, how the occurrence of one type of event may affect the future occurrence of another type. Furthermore, existing temporal variants of UGMs (Kolar et al., 2010; Yang et al., 2015b) usually assume that data are regularly sampled, and observations for all variables are available at each time point. Neither assumption is true, due to the irregularity of LED.

In contrast to these existing UGM models, a TPSQR models temporal relationships, by *data aggregation*; a TPSQR extracts a sequence of time-stamped summary count statistics of distinct event types that preserves the relative temporal order in the raw data for each subject. A PSQR is then used to model the joint distribution among these summary count statistics for each subject. Different PSQRs for different subjects are assumed to share the same *template parameterization* and hence can be learned jointly by estimating the template in a pseudo-likelihood fashion. To address the challenge in temporal irregularity, we compute the exact time difference between each pair of time-stamped summary statistics, and decide whether a difference falls into a particular predefined time interval, hence transforming the irregular time differences into regular timespans. We then incorporate the effects of various timespans into the template parameterization as well as PSQR constructions from the template.

By addressing temporality and irregularity of LED in this fashion, TPSQR is also different from many point process models (Gunawardana et al., 2011; Weiss et al., 2012; Weiss and Page, 2013; Du et al., 2016), which usually strive to pinpoint the exact occurrence times of events, and offer generative mechanisms to event trajectories. TPSQR, on the other hand, adopts a coarse resolution approach to temporal modeling via the aforementioned data aggregation and time interval construction. As a result, TPSQR focuses on estimating stable relationships among occurrences of different event types, and does not model the precise event occurrence timing. This behavior is especially meaningful in application settings such as ADR discovery, where the importance of identifying the occurrence of an adverse condition caused by the prescription of a drug usually outweighs knowing about the exact time point

of the occurrence of the ADR, due to the high variance of the onset time of ADRs (Schuemie et al., 2016).

Since TPSQR is a generalization of PSQR, many desirable properties of PSQR are inherited by TPSQR. For example, TPSQR, like PSQR, is capable of modeling both positive and negative dependencies between covariates. Such flexibility cannot usually be taken for granted when modeling a multivariate distribution over count data due to the potential dispersion of the partition function of a graphical model (Yang et al., 2015a). TPSQR can be learned by solving the pseudo-likelihood problem for PSQR. For efficiency and scalability, we use Poisson pseudo-likelihood to approximately solve the original pseudo-likelihood problem induced by a PSQR, and we show that the Poisson pseudo-likelihood approximation can recover the structure of the underlying PSQR under mild assumptions. Finally, we demonstrate the utility of TPSQRs using Marshfield Clinic EHRs with millions of drug prescription and condition diagnosis events for the task of adverse drug reaction (ADR) detection. Our contributions are three-fold:

- TPSQR is a generalization of PSQR made in order to represent the multivariate distributions among count variables evolving temporally in LED. TPSQR can accommodate both positive and negative dependencies among covariates, and can be learned efficiently via the pseudo-likelihood problem for PSQR.

- In terms of advancing the state-of-the-art of PSQR estimation, we propose Poisson pseudo-likelihood approximation in lieu of the original more computationally-intensive conditional distribution induced by the joint distribution of a PSQR. We show that under mild assumptions, the Poisson pseudo-likelihood approximation procedure is sparsistent (Ravikumar et al., 2007) with respect to the underlying PSQR. Our theoretical results not only justify the use of the more efficient Poisson pseudo-likelihood over the original conditional distribution for better estimation efficiency of PSQR but also establish a formal correspondence between the more intuitive but less stringent local Poisson graphical models (Allen and Liu, 2013) and the more rigorous but less convenient PSQRs.

- We apply TPSQR to Marshfield Clinic EHRs to determine the relationships between the occurrences of various drugs and the occurrences of various conditions, and offer more accurate estimations for adverse drug reaction (ADR) discovery, a challenging task in health analytics due to the (thankfully) rare and weak ADR signals encoded in the data, whose success is crucial to improving healthcare both financially and clinically (Sultana et al., 2013).

7.2 Background

We show how to deal with the challenges in temporality and irregularity mentioned in Section 7.1 via the use of data aggregation and an influence function for LED. We then define the template parameterization that is central to the modeling of TPSQRs.

7.2.1 Longitudinal Event Data

Longitudinal event data are time-stamped events of finitely many types collected across various subjects over time. Figure 7.1 visualizes the LED for two subjects. As shown in Figure 7.1, the occurrences of different event types are represented as arrows in different colors. No two events for one subject occur at the exact same time. We are interested in modeling the relationships among the occurrences of different types of events via TPSQR.

7.2.2 Data Aggregation

To enable PSQRs to cope with the temporality in LED, TPSQRs start by extracting relative-temporal-order-preserved summary count statistics from the raw LED via data aggregation, to cope with the high volume and frequent consecutive replications of events of the same type that are commonly observed in LED. Take Subject 1 in Figure 7.1 as an illustrative example; we divide the raw data of Subject 1 into four timespans by the dashed lines. Each of the four timespans contains only

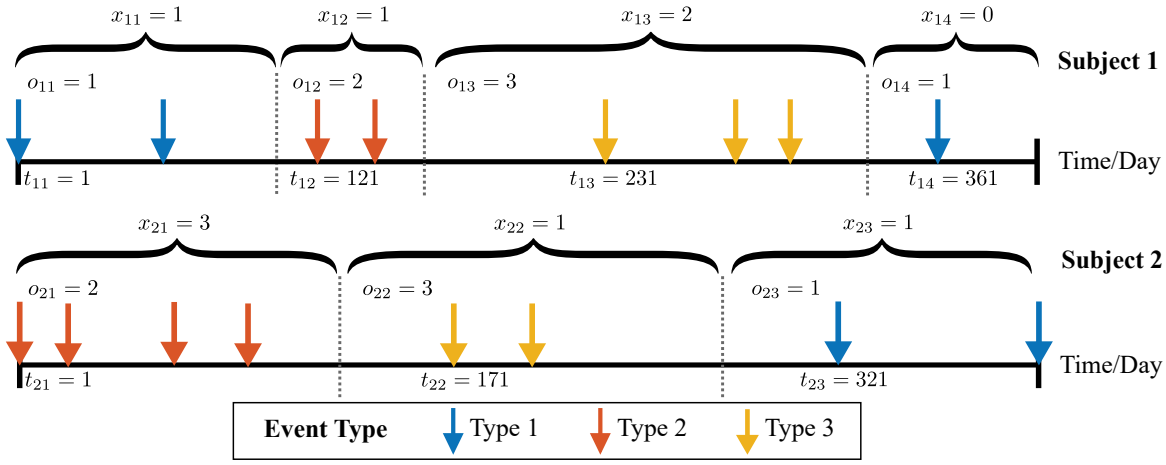


Figure 7.1: Visualization of longitudinal event data from two subjects. Curly brackets denote the timespans during which events of only one type occur. x_{ij}'s represent the number of subsequent occurrences after the first occurrence. o_{ij}'s are the types of events in various timespans.

events of the same type. We use three statistics to summarize each timespan: the time stamp of the first occurrence of the event in each timespan: $t_{11} = 1$, $t_{12} = 121$, $t_{13} = 231$, and $t_{14} = 361$; the event type in each timespan: $o_{11} = 1$, $o_{12} = 2$, $o_{13} = 3$, and $o_{14} = 1$; and the counts of subsequent occurrences in each timespan: $x_{11} = 1$, $x_{12} = 1$, $x_{13} = 2$, and $x_{14} = 0$. Note that the reason $x_{14} = 0$ is that there is only one occurrence of event type 1 in total during timespan 4 of subject 1. Therefore, the number of subsequent occurrence after the first and only occurrence is 0.

Let there be N independent subjects and p types of events in a given LED $\mathbb{X}$. We denote by n_i the number of timespans during which only one type of event occurs to subject i, where $i \in \{1, 2, \cdots, N\}$. The j^{th} timespan of the i^{th} subject can be represented by the vector $\mathbf{s}_{ij} := \begin{bmatrix} t_{ij} & o_{ij} & x_{ij} \end{bmatrix}^\top$, where $j \in \{1, 2, \cdots, n_i\}$, and ":=" represents "defined as." $t_{ij} \in [0, +\infty)$ is the time stamp at which the first event occurs during the timespan $\mathbf{s}_{ij}$. Furthermore, $t_{11} < t_{12} < \cdots < t_{1n_i}$. $o_{ij} \in \{1, 2, \cdots, p\}$ represents the event type in $\mathbf{s}_{ij}$. Furthermore, $o_{ij} \neq o_{i(j+1)}$, $\forall i \in \{1, 2, \cdots, N\}$ and $\forall j < n_i$. $x_{ij} \in \mathbb{N}$ is the number of subsequent occurrences of events of the same type in $\mathbf{s}_{ij}$.

7.2.3 Influence Function

Let $\mathbf{s}_{ij}$ and $\mathbf{s}_{ij'}$ be given, where $j < j' \leqslant n_i$. To handle the irregularity of the data, we map the time difference $t_{ij'} - t_{ij}$ to a one-hot vector that represents the activation of a time interval using an *influence function* $\boldsymbol{\phi}(\cdot)$, a common mechanism widely used in point process models and signal processing. In detail, let $L + 1$ user-specified time-threshold values be given, where $0 = \tau_0 < \tau_1 < \tau_2 < \cdots < \tau_L$. $\boldsymbol{\phi}(\tau)$ is a $L \times 1$ one hot vector whose l^{th} component is defined as:

$$[\boldsymbol{\phi}(\tau)]_l := \begin{cases} 1, & \tau_{l-1} \leqslant \tau < \tau_l \\ 0, & \text{otherwise} \end{cases}, \tag{7.1}$$

where $l \in \{1, 2, \cdots, L\}$. In our case, we let $\tau := t_{ij'} - t_{ij}$ to construct $\boldsymbol{\phi}(\tau)$ according to (7.1). Widely used influence functions in signal processing include the dyadic wavelet function and the Haar wavelet function (Mallat, 2008); both are piecewise constant and hence share similar representation to (7.1).

7.2.4 Template Parameterization

Template parameterization provides the capability of TPSQRs to represent the effects of all possible (ordered) pairs of event types on all time scales. Specifically, let an ordered pair $(k, k') \in \{1, 2, \cdots, p\}^2$ be given. Let $0 = \tau_0 < \tau_1 < \tau_2 < \cdots < \tau_K$ also be given. For the ease of presentation, we assume that $k \neq k'$, which can be easily generalized to $k = k'$. Considering a particular patient, we are interested in knowing the effect of an occurrence of a type k event towards a subsequent occurrence of a type k' event, when the time between the two occurrences falls in the l^{th} time window specified via (7.1). Enumerating all L time windows, we have:

$$\boldsymbol{w}_{kk'} := \begin{bmatrix} w_{kk'1} & w_{kk'2} & \cdots & w_{kk'L} \end{bmatrix}^\top. \tag{7.2}$$

Note that since (k, k') is ordered, $\boldsymbol{w}_{k'k}$ is different from $\boldsymbol{w}_{kk'}$. We further define $\mathbf{W}$ as a $(p-1)p \times L$ matrix that stacks up all $\boldsymbol{w}_{kk'}^\top$'s. In this way, $\mathbf{W}$ includes all

possible pairwise temporally bidirectional relationships among the p variables on different time scales, offering holistic representation power. To represent the intrinsic prevalence effect of the occurrences of events of various types, we further define $\boldsymbol{\omega} := \begin{bmatrix} \omega_1 & \omega_2 & \cdots & \omega_p \end{bmatrix}^\top$. We call $\boldsymbol{\omega}$ and $\mathbf{W}$ the template parameterization, from which we will generate the parameters of various PSQRs as shown in Section 7.3.

7.3 Modeling

Let $\mathbf{s}_{ij}$'s be given where $j \in \{1, 2, \cdots, n_i\}$; we demonstrate the use of the influence function and template parameterization to construct a PSQR for subject i.

Let $\mathbf{t}_i := \begin{bmatrix} t_{i1} & t_{i2} & \cdots & t_{in_i} \end{bmatrix}^\top$, $\mathbf{o}_i := \begin{bmatrix} o_{i1} & o_{i2} & \cdots & o_{in_i} \end{bmatrix}^\top$, and $\mathbf{x}_i := \begin{bmatrix} x_{i1} & x_{i2} & \cdots & x_{in_i} \end{bmatrix}^\top$. Given $\mathbf{t}_i$ and $\mathbf{o}_i$, a TPSQR aims at modeling the joint distribution of counts $\mathbf{x}_i$ using a PSQR. Specifically, under the template parameterization $\boldsymbol{\omega}$ and $\mathbf{W}$, we first define a symmetric parameterization $\boldsymbol{\Theta}^{(i)}$ using $\mathbf{t}_i$ and $\mathbf{o}_i$. The component of $\boldsymbol{\Theta}^{(i)}$ at the j^{th} row and the j'^{th} column is:

$$\theta^{(i)}_{jj'} := [\boldsymbol{\Theta}^{(i)}]_{jj'} := \begin{cases} \omega_{o_{ij}}, & j = j' \\ \boldsymbol{w}^\top_{o_{ij}o_{ij'}} \boldsymbol{\phi}(|t_{ij'} - t_{ij}|), & j < j' \\ [\boldsymbol{\Theta}^{(i)}]_{j'j}, & j > j' \end{cases}. \tag{7.3}$$

We then can use $\boldsymbol{\Theta}^{(i)}$ to parameterize a PSQR that gives a joint distribution over $\mathbf{x}_i$ as:

$$P\left(\boldsymbol{x}_i; \boldsymbol{\Theta}^{(i)}\right) := \exp\left[\sum_{j=1}^{n_i} \theta^{(i)}_{jj} \sqrt{x_{ij}} + \sum_{j=1}^{n_i - 1} \sum_{j'>j}^{n_i} \theta^{(i)}_{jj'} \sqrt{x_{ij} x_{ij'}} - \sum_{j=1}^{n_i} \log(x_{ij}!) - A_{n_i}\left(\boldsymbol{\Theta}^{(i)}\right) \right]. \tag{7.4}$$

In (7.4), $A_{n_i}(\boldsymbol{\Theta}^{(i)})$ is a normalization constant called the log-partition function that ensures the legitimacy of the probability distribution in question:

$$A_{n_i}\left(\boldsymbol{\Theta}^{(i)}\right) := \log \sum_{\boldsymbol{x} \in \mathbb{N}^{n_i}} \exp\left[\sum_{j=1}^{n_i} \theta^{(i)}_{jj} \sqrt{x_j} + \sum_{j=1}^{n_i - 1} \sum_{j'>j}^{n_i} \theta^{(i)}_{jj'} \sqrt{x_j x_{j'}} - \sum_{j=1}^{n_i} \log(x_j!) \right]. \tag{7.5}$$

Note that in (7.5) we emphasize the dependency of the partition function upon the dimension of $\mathbf{x}$ using the subscript n_i, and $\mathbf{x} := \begin{bmatrix} x_1 & x_1 & \cdots & x_{n_i} \end{bmatrix}$.

To model the joint distribution of $\mathbf{x}_i$, TPSQR directly uses $\boldsymbol{\Theta}^{(i)}$, which is extracted from $\boldsymbol{\omega}$ and $\mathbf{W}$ via (7.3) depending on the individual and temporal irregularity of the data characterized by $\mathbf{t}_i$ and $\mathbf{o}_i$. Therefore, $\boldsymbol{\omega}$ and $\mathbf{W}$ serve as a template for constructing $\boldsymbol{\Theta}^{(i)}$'s, and hence provide a "template parameterization." Since there are N subjects in total in the dataset, and each $\boldsymbol{\Theta}^{(i)}$ offers a personalized PSQR for one subject, TPSQR is capable of learning a collection of interrelated PSQRs due to the use of the template parameterization. Recall the well-rounded representation power of a template shown in Section 7.2.4; learning the template parameterization via TPSQR can hence offer a comprehensive perspective about the relationships for all possible temporally ordered pairs of event types.

Furthermore, since TPSQR is a generalization of PSQR, it inherits many desirable properties enjoyed by PSQR. A most prominent property is its capability of accommodating both positive and negative dependencies between variables. Such flexibility in general cannot be taken for granted when modeling multivariate count data. For example, a Poisson graphical model (Yang et al., 2015a) can only represent negative dependencies due to the diffusion of its log-partition function when positive dependencies are involved. Yet for example one drug (e.g., the blood thinner Warfarin) can have a positive influence on some conditions (e.g., bleeding) and a negative influence on others (e.g., stroke). We refer interested readers to Allen and Liu 2013; Yang et al. 2013; Inouye et al. 2015; Yang et al. 2015a; Inouye et al. 2016 for more details of PSQRs and other related Poisson graphical models.

7.4 Estimation

In this section, we present the pseudo-likelihood estimation problem for TPSQR. We then point out that solving this problem can be inefficient, which leads to the proposed Poisson pseudo-likelihood approximation to the original pseudo-likelihood problem.

7.4.1 Pseudo-Likelihood for TPSQR

We now present our estimation approach for TPSQR based on pseudo-likelihood. We start from considering the pseudo-likelihood for a given i^{th} subject. By (7.4), the log probability of x_{ij} conditioned on $\mathbf{x}_{i,-j}$, which is an $(n_i-1)\times 1$ vector constructed by removing the j^{th} component from $\mathbf{x}_i$, is given as:

$$\log P\left(x_{ij}|\mathbf{x}_{i,-j};\theta_j^{(i)}\right) = -\log(x_{ij}!)+\left(\theta_{jj}^{(i)}+\theta_{j,-j}^{(i)\top}\sqrt{\mathbf{x}_{i,-j}}\right)\sqrt{x_{ij}}-\tilde{A}_{n_i}\left(\theta_j^{(i)}\right), \quad (7.6)$$

where $\theta_j^{(i)}$ is the j^{th} column of $\mathbf{\Theta}^{(i)}$ and hence

$$\begin{aligned}\theta_j^{(i)} &:= \begin{bmatrix}\theta_{1j}^{(i)} & \cdots & \theta_{j-1,j}^{(i)} & \theta_{jj}^{(i)} & \theta_{j+1,j}^{(i)} & \cdots & \theta_{n_i,j}^{(i)}\end{bmatrix}^\top \\ &:= \begin{bmatrix}\theta_{1j}^{(i)} & \cdots & \theta_{j-1,j}^{(i)} & \theta_{jj}^{(i)} & \theta_{j,j+1}^{(i)} & \cdots & \theta_{j,n_i}^{(i)}\end{bmatrix}^\top. \end{aligned} \quad (7.7)$$

In (7.7), by the symmetry of $\mathbf{\Theta}^{(i)}$, we rearrange the index after $\theta_{jj}^{(i)}$ to ensure that the row index is no larger than the column index so that the parameterization is consistent with that in (7.4). We will adhere to this convention in the subsequent presentation. Furthermore, $\theta_{j,-j}^{(i)}$ is an $(n_i-1)\times 1$ vector constructed from $\theta_j^{(i)}$ by excluding its j^{th} component, and $\sqrt{\mathbf{x}_{i,-j}}$ is constructed by taking the square root of each component of $\mathbf{x}_{i,-j}$. Finally,

$$\tilde{A}_{n_i}\left(\theta_j^{(i)}\right) := \log \sum_{\mathbf{x}\in\mathbb{R}^{n_i}} \exp\left[\left(\theta_{jj}^{(i)}+\theta_{j,-j}^{(i)\top}\sqrt{\mathbf{x}_{i,-j}}\right)\sqrt{x_{ij}}-\log(x_{ij}!)\right], \quad (7.8)$$

which is a quantity that involves summing up infinitely many terms, and in general cannot be further simplified, leading to potential intractability in computing (7.8).

With the conditional distribution in (7.6) and letting $M := \sum_{i=1}^N n_i$, the pseudo-likelihood problem for TPSQR is given as:

$$\max_{\omega,W} \frac{1}{M}\sum_{i=1}^{N}\sum_{j=1}^{n_i}\log P\left(x_{ij}\,|\,\mathbf{x}_{i,-j};\theta_j^{(i)}\right). \quad (7.9)$$

(7.9) is the maximization over all the conditional distributions of all the count variables for all N personalized PSQRs generated by the template. *Therefore, it can be viewed as a pseudo-likelihood estimation problem directly for* $\boldsymbol{\omega}$ *and* $\boldsymbol{W}$. However, solving the pseudo-likelihood problem in (7.9) involves the predicament of computing the potentially intractable (7.8), which motivates us to use Poisson pseudo-likelihood as an approximation to (7.9).

7.4.2 Poisson Pseudo-Likelihood

Using the parameter vector $\theta_j^{(i)}$, we define the conditional distribution of x_{ij} given by $\boldsymbol{x}_{i,-j}$ via the Poisson distribution as:

$$\hat{P}\left(x_{ij} \mid \boldsymbol{x}_{i,-j}; \theta_j^{(i)}\right) \propto \exp\left[\left(\theta_{jj}^{(i)} + \theta_{-j}^{(i)\top}\boldsymbol{x}_{i,-j}\right) x_{ij} - \exp\left(\theta_{jj}^{(i)} + \theta_{-j}^{(i)\top}\boldsymbol{x}_{i,-j}\right)\right]. \tag{7.10}$$

Notice the similarity between (7.6) and (7.10). We can define the sparse Poisson pseudo-likelihood problem similar to the original pseudo-likelihood problem by replacing $\log P\left(x_{ij} \mid \boldsymbol{x}_{i,-j}; \theta_j^{(i)}\right)$ with $\log \hat{P}\left(x_{ij} \mid \boldsymbol{x}_{i,-j}; \theta_j^{(i)}\right)$:

$$\max_{\boldsymbol{\omega},\boldsymbol{W}} \frac{1}{M}\sum_{i=1}^{N}\sum_{j=1}^{n_i} \log \hat{P}\left(x_{ij} \mid \boldsymbol{x}_{i,-j}; \theta_j^{(i)}\right) - \lambda\|\boldsymbol{W}\|_{1,1}, \tag{7.11}$$

where $\lambda \geqslant 0$ is the regularization parameter, and the penalty

$$\|\boldsymbol{W}\|_{1,1} := \sum_{i=1}^{(p-1)p}\sum_{j=1}^{L} |[\boldsymbol{W}]_{ij}|$$

is used to encourage sparsity over the template parameterization $\boldsymbol{W}$ that determines the interactions between the occurrences of two distinct event types. As mentioned at the end of Section 7.4.1, TPSQR learning is equivalent to learning a PSQR over the template parameterization. Therefore, the sparsity penalty induced here is helpful to recover the structure of the underlying graphical model.

The major advantage of approximating the original pseudo-likelihood problem

with Poisson pseudo-likelihood is the gain in computational efficiency. Based on the construction in Geng et al. 2017, (7.11) can be formulated as an L_1-regularized Poisson regression problem, which can be solved much more efficiently via many sophisticated algorithms and their implementations (Friedman et al., 2010; Tibshirani et al., 2012) compared to solving the original problem that involves the potentially challenging computation for (7.8). Furthermore, in the subsequent section, we will show that even though the Poisson pseudo-likelihood is an approximation procedure to the pseudo-likelihood of PSQR, under mild assumptions Poisson pseudo-likelihood is still capable of recovering the structure of the underlying PSQR.

7.4.3 Sparsistency Guarantee

For the ease of presentation, in this section we will reuse much of the notation that appears previously. The redefinitions introduced in this section only apply to the contents in this section and the related proofs in the Appendix. Recall at the end of Section 7.4.1, the pseudo-likelihood problem of TPSQR can be viewed as learning a PSQR parameterized by the template. Therefore, without loss of generality, we will consider a PSQR over p count variables $\mathbf{X} = \mathbf{x} \in \mathbb{N}^p$ parameterized by a $p \times p$ symmetric matrix $\mathbf{\Theta}^*$, where $\mathbf{X} := \begin{bmatrix} X_1 & X_2 & \cdots & X_p \end{bmatrix}^\top$ is the multivariate random variable, and $\mathbf{x}$ is an assignment to $\mathbf{X}$. We use $\|\cdot\|_\infty$ to represent the infinity norm of a vector or a matrix. Let $\mathbb{X} := \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ be a dataset with n independent and identically distributed (i.i.d.) samples generated from the PSQR. Then the joint probability distribution over $\mathbf{x}$ is:

$$P(\mathbf{x}; \mathbf{\Theta}^*) := \exp\left[\sum_{j=1}^{p} \theta^*_{jj}\sqrt{x_{ij}} + \sum_{j=1}^{p-1}\sum_{j'>j}^{p} \theta^*_{jj'}\sqrt{x_{ij}x_{ij'}} - \sum_{j=1}^{p}\log(x_{ij}!) - A\left(\mathbf{\Theta}^*\right)\right],$$

where $A(\mathbf{\Theta}^*)$ is the log-partition function, and the corresponding Poisson pseudo-likelihood problem is:

$$\hat{\mathbf{\Theta}} := \arg\min_{\mathbf{\Theta}} F(\mathbf{\Theta}) + \lambda \|\mathbf{\Theta}\|_{1,\text{off}}, \tag{7.12}$$

where

$$F(\mathbf{\Theta}) := \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p}\Big[-(\theta_{jj} + \theta_{j,-j}^{\top}\mathbf{x}_{i,-j})x_{ij} + \exp(\theta_{jj} + \theta_{j,-j}^{\top}\mathbf{x}_{i,-j})\Big], \tag{7.13}$$

and $\|\mathbf{\Theta}\|_{1,\text{off}}$ represents imposing L_1 penalty over all but the diagonal components of $\mathbf{\Theta}$.

Sparsistency (Ravikumar et al., 2007) addresses whether $\hat{\mathbf{\Theta}}$ can recover the structure of the underlying $\mathbf{\Theta}^*$ with high probability using n i.i.d. samples. In what follows, we will show that $\hat{\mathbf{\Theta}}$ is indeed sparsistent under mild assumptions.

We use $\mathbb{E}[\cdot]$ to denote the expectation of a random variable under $P(\boldsymbol{x}; \mathbf{\Theta}^*)$. The first assumption is about the boundedness of $\mathbb{E}[\mathbf{X}]$, and the boundedness of the partial second order derivatives of a quantity related to the log-partition $A(\mathbf{\Theta}^*)$. This assumption is standard in the analysis of pseudo-likelihood methods (Yang et al., 2015a).

Assumption 1. $\|\mathbb{E}[\mathbf{X}]\|_\infty \leqslant C_1$ *for some* $C_1 > 0$. *Let*

$$B(\mathbf{\Theta}, \mathbf{b}) := \log \sum_{\mathbf{x}\in\mathbb{N}^p} \exp\left[\sum_{j=1}^{p}\theta_{jj}\sqrt{x_j} + \mathbf{b}^{\top}\mathbf{x} + \sum_{j=1}^{p-1}\sum_{j'>j}^{p}\theta_{jj'}\sqrt{x_j x_{j'}} - \sum_{j=1}^{p}\log(x_j!)\right].$$

For some $C_2 > 0$, *and* $\forall k \in [0,1]$,

$$\forall j \in \{1, 2, \cdots, p\}, \quad \frac{\partial^2 B(\mathbf{\Theta}, \mathbf{0} + k\boldsymbol{e}_j)}{\partial^2 b_j} \leqslant C_2,$$

where $\boldsymbol{e}_j$ *is the one-hot vector with the* j^{th} *component as* 1.

The following assumption characterizes the boundedness of the conditional

distributions given by the PSQR under $\boldsymbol{\Theta}^*$ and by the Poisson approximation using the same $\boldsymbol{\Theta}^*$.

Assumption 2. *Let* $\lambda_{ij}^* := \exp\left(\theta_{jj}^* + \boldsymbol{\theta}_{j,-j}^{*\top} \mathbf{x}_{i,-j}\right)$ *be the mean parameter of a Poisson distribution. Then* $\forall i \in \{1, 2 \cdots, n\}$ *and* $\forall j \in \{1, 2, \cdots, p\}$*, for some* $C_3 > 0$ *and* $C_4 > 0$*, we have that* $\mathbb{E}\left[X_j \mid \mathbf{x}_{i,-j}\right] \leqslant C_3$ *and* $|\lambda_{ij}^* - \mathbb{E}\left[X_j \mid \mathbf{x}_{i,-j}\right]| \leqslant C_4$*.*

The third assumption is the mutual incoherence condition vital to the sparsistency of sparse statistical learning with L_1-regularization. Also, with a slight abuse of notation, in the remaining of Section 7.4.3 as well as in the corresponding proofs, we should view $\boldsymbol{\Theta}$ as a vector generated by stacking up $\theta_{jj'}$'s, where $j \leqslant j'$, whenever it is clear from context.

Assumption 3. *Let* $\boldsymbol{\Theta}^*$ *be given. Define the index sets*

$$
\begin{aligned}
A &:= \left\{(j, j') \mid \theta_{jj'}^* \neq 0, j \neq j', j, j' \in \{1, 2, \cdots, p\}\right\}, \\
D &:= \{(j, j) \mid j \in \{1, 2, \cdots, p\}\}, \quad S := A \cup D, \\
I &:= \left\{(j, j') \mid \theta_{jj'}^* = 0, j \neq j', j, j' \in \{1, 2, \cdots, p\}\right\}.
\end{aligned}
$$

Let $\mathbf{H} := \boldsymbol{\nabla}^2 F(\boldsymbol{\Theta}^*)$*. Then for some* $0 < \alpha < 1$ *and* $C_5 > 0$*, we have* $\left\|\mathbf{H}_{IS}\mathbf{H}_{SS}^{-1}\right\|_\infty \leqslant 1 - \alpha$ *and* $\left\|\mathbf{H}_{SS}^{-1}\right\|_\infty \leqslant C_5$*, where we use the index sets as subscripts to represent the corresponding components of a vector or a matrix.*

The final assumption characterizes the second-order Taylor expansion of $F(\boldsymbol{\Theta}^*)$ at a certain direction $\boldsymbol{\Delta}$.

Assumption 4. *Let* $\mathbf{R}(\boldsymbol{\Delta})$ *be the second-order Taylor expansion remainder of* $\boldsymbol{\nabla} F(\boldsymbol{\Theta})$ *around* $\boldsymbol{\Theta} = \boldsymbol{\Theta}^*$ *at direction* $\boldsymbol{\Delta} := \boldsymbol{\Theta} - \boldsymbol{\Theta}^*$ *(i.e.* $\boldsymbol{\nabla} F(\boldsymbol{\Theta}) = \boldsymbol{\nabla} F(\boldsymbol{\Theta}^*) + \boldsymbol{\nabla}^2 F(\boldsymbol{\Theta}^*)(\boldsymbol{\Theta} - \boldsymbol{\Theta}^*) + \mathbf{R}(\boldsymbol{\Delta})$*), where* $\|\boldsymbol{\Delta}\|_\infty \leqslant r := 4C_5\lambda \leqslant \frac{1}{C_5 C_6}$ *with* $\boldsymbol{\Delta}_I = \mathbf{0}$*, and for some* $C_6 > 0$*. Then* $\|\mathbf{R}(\boldsymbol{\Delta})\|_\infty \leqslant C_6 \|\boldsymbol{\Delta}\|_\infty^2$*.*

With these mild assumptions, the sparsistency result is stated in Theorem 7.1.

Theorem 7.1. *Suppose that Assumption 1 - 4 are all satisfied. Then, with probability of at least* $1 - \left((\exp(C_1 + C_2/2) + 8)\, p^{-2} + p^{-1/C_2}\right)$, $\hat{\mathbf{\Theta}}$ *shares the same structure with* $\mathbf{\Theta}^*$, *if for some constant* $C_7 > 0$,

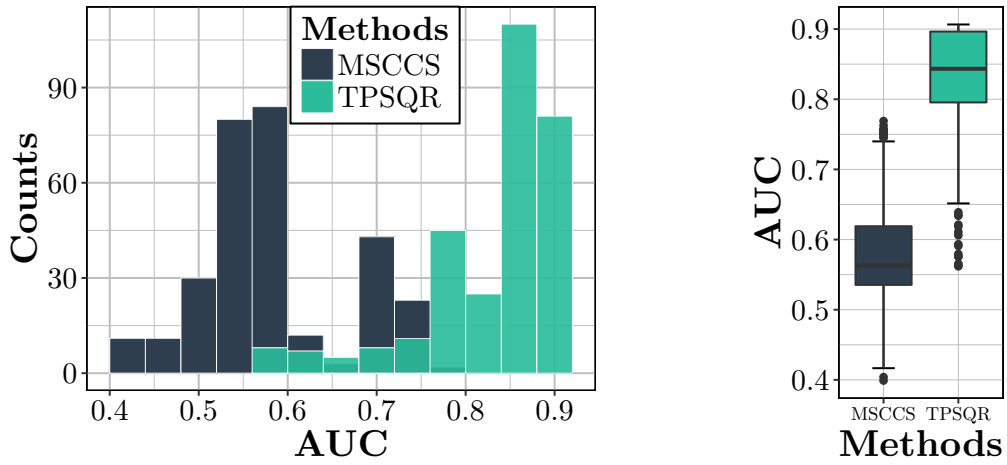
$$\lambda \geqslant \frac{8}{\alpha}\left[C_3(3\log p + \log n) + (3\log p + \log n)^2\right]\sqrt{\frac{\log p}{n}} + 8C_4\left(C_1 + \sqrt{\frac{2\log p}{n}}\right)\alpha,$$

$$\lambda \leqslant C_7\sqrt{\frac{\log^5 p}{n}}, r \leqslant \left\|\mathbf{\Theta}^*_S\right\|_\infty, \textit{ and } n \geqslant \left(64C_7C_5^2C_6/\alpha\right)^2\log^5 p.$$

We defer the proof of Theorem 7.1 to the Appendix. Note that $\log^5 p$ in Theorem 7.1 represents a higher sample complexity compared to similar results in the analysis of Ising models (Ravikumar et al., 2010). Such a higher sample complexity intuitively makes sense since the multivariate count variables that we deal with are unbounded and usually heavy-tailed, and we are also considering the Poisson pseudo-likelihood approximation to the original pseudo-likelihood problem induced by PSQRs. The fact that Poisson pseudo-likelihood is a sparsistent procedure for learning PSQRs not only provides an efficient approach to learn PSQRs with strong theoretical guarantees, but also establishes a formal correspondence between local Poisson graphical models (LPGMs, Allen and Liu 2013) and PSQRs. This is because Poisson pseudo-likelihood is also a sparsistent procedure for LPGMs. Compared to PSQRs, LPGMs are more intuitive yet less stringent theoretically due to the lack of a joint distribution defined by the model. Fortunately, with the guarantees in Theorem 7.1, we are able to provide some reassurance for the use of LPGMs in terms of structure recovery.

7.5 Adverse Drug Reaction Discovery

To demonstrate the capability of TPSQRs to capture temporal relationships between different pairs of event types in LED, we use ADR discovery from EHR as an example. ADR discovery is the task of finding unexpected and negative incidents caused by drug prescriptions. In EHR, time-stamped drug prescriptions as well as



(a) Overlapping Histograms (b) Boxplot

Figure 7.2: Overall performance of TPSQR and MSCCS measured by AUC among 300 different experimental configurations for each of the two methods.

condition diagnoses are collected from millions of patients. These prescriptions of different drugs and diagnoses of different conditions can hence be viewed as various event types in LED. Therefore, using TPSQR, we can model whether the occurrences of a particular drug k could elevate the possibility of the future occurrences of a condition k' on different time scales by estimating $w_{kk'}$ defined in (7.2). If an elevation is observed, we can consider the drug k as a potential candidate to cause condition k' as an adverse drug reaction.

Postmarketing ADR surveillance from EHR is a multi-decade research and practice effort that is of utmost importance to the pharmacovigilance community (Bate et al., 2018), with substantial financial and clinical implication for health care delivery (Sultana et al., 2013). Various ADR discovery methods have been proposed over the years (Harpaz et al., 2012), and a benchmark task is created by the Observational Medical Outcome Partnership (OMOP, Simpson 2011) to evaluate the ADR signal detection performance of these methods. The OMOP task is to identify the ADRs in 50 drug-condition pairs, coming from a selective combination of ten different drugs and nine different conditions. Among the 50 pairs, 9 of them are confirmed ADRs, while the remaining 41 of them are negative controls.

A most successful ADR discovery method using EHR is the multiple self-controlled case series (MSCCS, Simpson et al. 2013), which has been deployed in real-world ADR discovery related projects (Hripcsak et al., 2015). A reason for the success of MSCCS is its introduction of fixed effects to address the heterogeneity among different subjects (e.g. patients in poorer health might tend to be more likely to have a heart attack compared to a healthy person, which might confound the effects of various drugs when identifying drugs that could cause heart attacks as an ADR).

Therefore, when using TPSQR, we will also introduce fixed effects to equip TPSQRs with the capability of addressing subject heterogeneity. Specifically, we consider learning a variant of (7.11):

$$\max_{\boldsymbol{\alpha},\omega,\mathbf{W}} \frac{1}{M}\sum_{i=1}^{N}\sum_{j=1}^{n_i}\alpha_{io_{ij}}+\log \hat{P}\left(x_{ij} \mid \mathbf{x}_{i,-j}; \theta_j^{(i)}\right)-\lambda\|\mathbf{W}\|_{1,1},$$

where $\boldsymbol{\alpha}$ is the fixed effect parameter vector constructed by $\alpha_{io_{ij}}$'s that depicts the belief that different patients could have different baseline risks of experiencing different types of events.

7.6 Experiments

In what follows, we will compare the performances of TPSQR, MSCCS, and Hawkes process (Bao et al., 2017a) in the OMOP task. The experiments are conducted using Marshfield Clinic EHRs with millions of drug prescription and condition diagnosis events from 200,000 patients.

7.6.1 Experimental Configuration

Minimum Duration: clinical encounter sequences from different patients might span across different time lengths. Some have decades of observations in their records while other might have records only last a few days. We therefore consider

minimum duration of the clinic encounter sequence as a threshold to determine whether we admit a patient to the study or not. In our experiments, we consider two minimum duration thresholds: *0.5 year* and *1 year*.

Maximum Time Difference: for TPSQR, in (7.1), τ_L determines the maximum time difference between the occurrences of two events within which the former event might have nonzero influence on the latter event. We call τ_L the maximum time difference to characterize how distant in the past we would like to take previous occurrences into consideration when modeling future occurrences. In our experiments, we consider three maximum time differences: *0.5 year*, *1 year*, and *1.5 years*. $L = 3$ and the corresponding influence functions are chosen according to Bao et al. 2017a. In MSCCS, a configuration named *risk window* serves a similar purpose to the maximum difference in TPSQR. We choose three risk windows according to Kuang et al. 2017c so as to ensure that the both TPSQR and MSCCS have similar capability in considering the event history on various time scales.

Regularization Parameter: we use L_1-regularization for TPSQR since it encourages sparsity, and the sparsity patterns learned correspond to the structures of the graphical models. We use L_2-regularization for MSCCS since it yields outstanding empirical performance in previous studies (Simpson et al., 2013; Kuang et al., 2017c). 50 regularization parameters are chosen for both TPSQR and MSCCS.

To sum up, there are $2 \times 3 \times 50 = 300$ experimental configurations respectively for TPSQR and MSCCS.

7.6.2 Overall Performance

For each of the 300 experimental configurations for TPSQR and MSCCS, we perform the OMOP task using our EHR data. Both TPSQR and MSCCS can be implemented by the `R` package `glmnet` (Friedman et al., 2010; Tibshirani et al., 2012). We then use Area Under the Curve (AUC) for the receiver operating characteristic curve to evaluate how well TPSQR and MSCCS can distinguish actual ADRs from negative controls under this particular experimental configuration. The result is 300 AUCs corresponding to the total number of experimental configurations for each of the

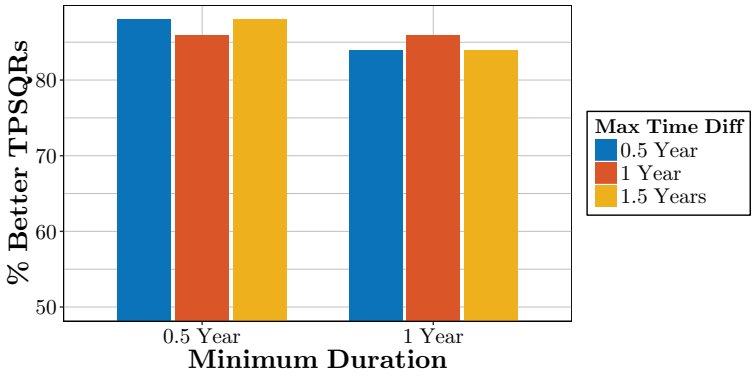


Figure 7.3: Percentage of better TPSQR models under various minimum duration and maximum time difference designs compared to the best MSCCS model.

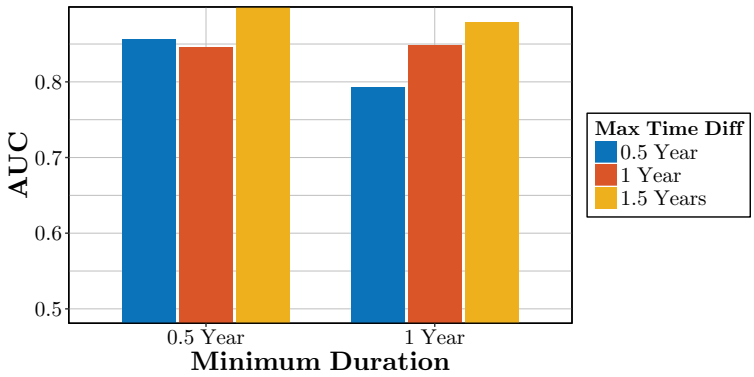


Figure 7.4: AUC of TPSQR models selected by AIC for given minimum duration and maximum time difference designs.

two methods. For TPSQR, since the effect of drug k on condition k' is estimated over different time scales via $\boldsymbol{w}_{kk'}$, the score corresponding to this drug-condition pair used to calculate the AUC is computed by the average over all the components of $\boldsymbol{w}_{kk'}$. For MSCCS, AUC is computed according to Kuang et al. 2017c. Figure 7.2 presents the histogram of these two sets of 300 AUCs. The contrast in the performances between TPSQR and MSCCS is obvious. The distribution of TPSQR shifts substantially towards higher AUC values compared to the distribution of MSCCS. Therefore, the overall performance of TPSQR is superior to that of MSCCS in the OMOP task under various experimental configurations in question. As a matter

of fact, the top performing TPSQR model reaches an AUC of 0.91, as opposed to 0.77 for MSCCS. Furthermore, the majority of TPSQRs have higher AUCs even compared to the MSCCS model that has the best AUC. We also contrast the performance of TPSQR with the Hawkes process method in Bao et al. 2017a, whose best AUC is 0.84 under the same experiment configurations.

7.6.3 Sensitivity Analysis and Model Selection

To see how sensitive the performance of TPSQR is for different choices of experimental configurations, we compute the percentage of TPSQRs with a given minimum duration and a given maximum time difference design that are better than the best MSCCS model (with an AUC of 0.77). The results are summarized in Figure 7.3. As can be seen, the percentage of better TPSQRs is consistently above 80% under various scenarios, suggesting the robustness of TPSQRs to various experimental configurations. Given a fixed minimum duration and a fixed maximum time difference, we conduct model selection for TPQSRs by the Akaike information criterion (AIC) over the regularization parameters. The AUC of the selected models are summarized in Figure 7.4. Note that under various fixed minimum duration and maximum time difference designs, AIC is capable of selecting models with high AUCs. In fact, all the models selected by AIC have higher AUCs than the best performer of MSCCS. This phenomenon demonstrates that the performance of TPSQR is consistent and robust with respect to the various choices of experimental configurations.

7.7 Conclusion

We propose TPSQRs, a generalization of PSQRs for the temporal relationships between different event types in LED. We propose the use of Poisson pseudo-likelihood approximation to solve the pseudo-likelihood problem arising from PSQRs. The approximation procedure is extremely efficient to solve, and is sparsis-

tent in recovering the structure of the underlying PSQR. The utility of TPSQR is demonstrated using Marshfield Clinic EHRs for adverse drug reaction discovery.

7.8 Appendix

We prove Theorem 7.1 in this section. Since the proof is technical and lengthy, for the clarity of presentation, we organize the proof as follows. To begin with, in Section 7.8.1, we review two standard concentration inequalities, the Chernoff inequality and the Hoeffding inequality, which will be used to prove some technical lemmas. We then present and prove these technical lemmas in Section 7.8.2. These technical lemmas are subsequently used to validate some auxiliary results, which are presented in Section 7.8.3. Finally, we prove Theorem 7.1 based on these auxiliary results.

7.8.1 Concentration Inequalities

Lemma 7.2 (Hoeffding Inequality)**.** *Let* $X_1, X_2, \cdots, X_n$ *be* n *i.i.d. random variables drawn from the distribution* $\mathcal{D}$*, with* $0 \leqslant X_i \leqslant a, \forall i \in \{1, 2, \cdots, n\}$*. Let* $\bar{X} := \frac{1}{n}\sum_{i=1}^{n} X_i$*. Then, for any* $t > 0$*,*

$$P(|\bar{X} - \mathbb{E}[\bar{X}]| \geqslant t) \leqslant 2\exp\left(-\frac{2nt^2}{a^2}\right).$$

Lemma 7.3 (Chernoff Inequality)**.** *Let* $X_1, X_2, \cdots, X_n$ *be* n *random variables and let* $X := \sum_{i=1}^{n} X_i$*. Then, for any* $t > 0$*,*

$$P(X \geqslant \epsilon) \leqslant \exp(-t\epsilon)\mathbb{E}\left[\exp\left(\sum_{i=1}^{n} tX_i\right)\right]. \tag{7.14}$$

Furthermore, if X_i*'s are independent, then*

$$P(X \geqslant \epsilon) \leqslant \min_{t>0} \exp(-t\epsilon) \prod_{i=1}^{n} \mathbb{E}\left[\exp(tX_i)\right]. \tag{7.15}$$

7.8.2 Technical Lemmas

We use $\|\cdot\|_{\max}$ to represent the max norm of a matrix, which is equal to the maximum of the absolute value of all the elements in the matrix.

Lemma 7.4. *Let $\mathbb{X}$ be given. Suppose that $0 < \max_{i,i' \in \{1,2,\cdots,n\}} \|\mathbf{x}_i \mathbf{x}_{i'}^\top\|_{\max} < \epsilon^2$. Then,*

$$P\left(\max_{j \neq j', j \neq j', j, j' \in \{1,2,\cdots,p\}} |\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] - \mathbb{E}[X_j X_{j'}]| \geqslant \epsilon^2 \sqrt{\frac{\log p}{n}}\right) \leqslant 2\exp(-2\log p).$$

Proof. Since $0 < \max_{i,i' \in \{1,2,\cdots,n\}} \|\mathbf{x}_i \mathbf{x}_{i'}^\top\|_{\max} < \epsilon^2$, we let $a = \epsilon^2$ and $t = \epsilon^2 \sqrt{\frac{\log p}{n}}$ in Lemma 7.2 to yield the result. □

Lemma 7.5. *Let $\mathbb{X}$ be given. Suppose that $0 < \max_{i \in \{1,2,\cdots,n\}} \|\mathbf{x}_i\|_\infty < \epsilon$. Then,*

$$P\left(\max_{j \in \{1,2,\cdots,p\}} |\mathbb{E}_{\mathbb{X}}[X_j] - \mathbb{E}[X_j]| \geqslant \epsilon \sqrt{\frac{\log p}{n}}\right) \leqslant 2\exp(-2\log p).$$

Proof. Since $0 < \max_{i \in \{1,2,\cdots,n\}} \|\mathbf{x}_i\|_\infty < \epsilon$, we let $a = \epsilon$ and $t = \epsilon \sqrt{\frac{\log p}{n}}$ in Lemma 7.2 to yield the result. □

Lemma 7.6. *Let $\mathbb{X}$ be given. Suppose that $0 < \max_{i \in \{1,2,\cdots,n\}} \|\mathbf{x}_i\|_\infty < \epsilon$. Then,*

$$\begin{aligned} &P\left(\max_{j, j' \in \{1,2,\cdots,p\}} |\mathbb{E}_{\mathbb{X}}[\mathbb{E}[X_j X_{j'} \mid \mathbf{X}_{-j}]] - \mathbb{E}[\mathbb{E}[X_j X_{j'} \mid \mathbf{X}_{-j}]]| \geqslant C_3 \epsilon \sqrt{\frac{\log p}{n}}\right) \\ \leqslant &2\exp(-2\log p). \end{aligned}$$

Proof. Since $0 < \max_{i \in \{1,2,\cdots,n\}} \|\mathbf{x}_i\|_\infty < \epsilon$ and $\mathbb{E}[X_j \mid \mathbf{x}_{i,-j}] \leqslant C_3$ by Assumption 2, we have that $0 < \mathbb{E}[X_j X_{j'} \mid \mathbf{x}_{i,-j}] \leqslant C_3 \epsilon$. Therefore, we let $a = C_3 \epsilon$ and $t = C_3 \epsilon \sqrt{\frac{\log p}{n}}$ in Lemma 7.2 to yield the result. □

Remark

The subtlety of the definitions of C_3 and C_4 in Assumption 2, as well as the notion of ϵ in Lemma 7.4, Lemma 7.5, and Lemma 7.6 should be noted. Formally, the n data points $x_1, x_2, \cdots, x_n$ in $\mathbb{X}$ can be viewed as assignments to the corresponding random variables $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \cdots, \mathbf{X}^{(n)}$ following the PSQR parameterized by $\boldsymbol{\Theta}^*$. In Assumption 2, we are interested in a set $\mathcal{X} \subseteq \mathbb{N}^p$, such that $\forall i \in \{1, 2, \cdots, n\}$ and $\forall j \in \{1, 2, \cdots, p\}$,

$$\max_{\mathbf{X}^{(i)} \in \mathcal{X}} \mathbb{E}\left[X_j \mid \mathbf{X}_{-j}^{(i)}\right] \leqslant C_3 \quad \text{and} \quad \max_{\mathbf{X}^{(i)} \in \mathcal{X}} |\lambda_{ij}^* - \mathbb{E}\left[X_j \mid \mathbf{X}_{-j}^{(i)}\right]| \leqslant C_4.$$

In Lemma 7.4, Lemma 7.5, and Lemma 7.6, we are interested in a set $\mathcal{X} \subseteq \mathbb{N}^p$, such that $\forall i, i' \in \{1, 2, \cdots, n\}$, where $i \neq i'$,

$$0 < \max_{\mathbf{X}^{(i)}, \mathbf{X}^{(i')} \in \mathcal{X}} \|\mathbf{X}^{(i)} \mathbf{X}^{(i')\top}\|_{\max} < \epsilon^2 \quad \text{and} \quad 0 < \max_{\mathbf{X}^{(i)} \in \mathcal{X}} \|\mathbf{X}^{(i)}\|_\infty < \epsilon.$$

Also, implicitly, we have that $x_i \in \mathcal{X}$, $\forall i \in \{1, 2, \cdots, n\}$.

Lemma 7.7. *Let $\mathbb{X}$ be given. Then,*

$$P\left(\max_{j \in \{1,2,\cdots,p\}} |\mathbb{E}_{\mathbb{X}}[\mathbb{E}[X_j \mid \mathbf{X}_{-j}]] - \mathbb{E}[\mathbb{E}[X_j \mid \mathbf{X}_{-j}]]| \geqslant C_3 \sqrt{\frac{\log p}{n}}\right) \leqslant 2 \exp(-2 \log p).$$

Proof. Since $\mathbb{E}[X_j \mid x_{i,-j}] \leqslant C_3$ by Assumption 2, we have that $0 < \mathbb{E}[X_j \mid x_{i,-j}] \leqslant C_3$. Therefore, we let $a = C_3$ and $t = C_3 \sqrt{\frac{\log p}{n}}$ in Lemma 7.2 to yield the result. □

Lemma 7.8. *Let $\mathbf{X}$ be a random vector drawn from a PSQR distribution parameterized by $\boldsymbol{\Theta}^*$. Suppose that $\{x_1, x_2, \cdots x_n\}^\top$ is the set of n i.i.d. samples of $\mathbf{X}$. Given $j \in \{1, 2, \cdots, p\}$, $\epsilon_1 := 3 \log p + \log n$, and $\epsilon_2 := C_1 + \sqrt{\frac{2 \log p}{n}}$,*

$$P(X_j \geqslant \epsilon_1) \leqslant \exp(C_1 + C_2/2 - \epsilon_1), \text{ and } P\left(\frac{1}{n} \sum_{i=1}^{n} x_{ij} \geqslant \epsilon_2\right) \leqslant \exp\left[-\frac{n(\epsilon_2 - C_1)^2}{2C_2}\right].$$

Proof. We start with proving the first inequality. To this end, consider the following equation due to Taylor expansion:

$$\begin{aligned}\log \mathbb{E}\left[\exp(X_j)\right] =& B(\boldsymbol{\Theta}^*, \mathbf{0} + \mathbf{e}_j) - B(\boldsymbol{\Theta}^*, \mathbf{0}) = \boldsymbol{\nabla}^\top B(\boldsymbol{\Theta}^*, \mathbf{0})\mathbf{e}_j + \frac{1}{2}\mathbf{e}_j^\top \boldsymbol{\nabla}^2 B(\boldsymbol{\Theta}^*, k\mathbf{e}_j)\mathbf{e}_j \\ =& \mathbb{E}[X_j] + \frac{1}{2}\frac{\partial^2}{\partial b_j^2} B(\boldsymbol{\Theta}^*, \mathbf{0} + k\mathbf{e}_j) \leqslant C_1 + C_2/2,\end{aligned} \tag{7.16}$$

where $k \in [0, 1]$, $\mathbf{e}_j$ is a vector whose j^{th} component is one and zeros elsewhere, and the last inequality is due to Assumption 1. Then, let $t = 1$ and $X = X_j$ in Lemma 7.3,

$$P\left(X_j \geqslant \epsilon_1\right) = \exp(-\epsilon_1)\mathbb{E}\left[\exp(X_j)\right] \leqslant \exp(C_1 + C_2/2 - \epsilon_1).$$

Now, we prove the second bound. For any $0 < a < 1$ and some $k \in [0, 1]$, with Taylor expansion,

$$\begin{aligned}\log \mathbb{E}\left[\exp(aX_i)\right] =& B(\boldsymbol{\Theta}^*, \mathbf{0} + a\mathbf{e}_j) - B(\boldsymbol{\Theta}^*, \mathbf{0}) \\ =& a\boldsymbol{\nabla}^\top B(\boldsymbol{\Theta}^*, \mathbf{0})\mathbf{e}_j + \frac{a^2}{2}\mathbf{e}_j^\top \boldsymbol{\nabla}^2 B(\boldsymbol{\Theta}^*, \mathbf{0} + ak\boldsymbol{e}_j)\mathbf{e}_j \\ =& a\mathbb{E}(X_j) + \frac{a^2}{2}\frac{\partial^2}{\partial b_j^2} B(\boldsymbol{\Theta}^*, \mathbf{0} + ak\mathbf{e}_j) \leqslant aC_1 + \frac{a^2}{2}C_2,\end{aligned} \tag{7.17}$$

where the last inequality is due to Assumption 1. Then, following the proof technique above, we have

$$\begin{aligned}P\left(\frac{1}{n}\sum_{i=1}^n X_i \geqslant \epsilon_2\right) =& P\left(\sum_{i=1}^n X_i \geqslant n\epsilon_2\right) \leqslant \min_{t>0} \exp(-tn\epsilon_2) \prod_{i=1}^n \mathbb{E}\left[\exp\left(tX_i\right)\right] \\ \leqslant& \min_{t>0} \exp(-tn\epsilon_2) \prod_{i=1}^n \exp\left(C_1 t + \frac{C_2}{2}t^2\right) \\ =& \min_{t>0} \exp\left[(C_1 - \epsilon_2)nt + \frac{nC_2}{2}t^2\right] \leqslant \exp\left[-\frac{n(\epsilon_2 - C_1)^2}{2C_2}\right],\end{aligned}$$

where the minimum is obtained when $t = \frac{\epsilon_2 - C_1}{C_2}$, and we have used the fact that $\epsilon_2 > C_1$. □

7.8.3 Auxiliary Results

Lemma 7.9. *Let* $r := 4C_5\lambda$. *Then with probability of at least*

$$1 - \left((\exp(C_1 + C_2/2) + 8)\, p^{-2} + p^{-1/C_2} \right),$$

the following two inequalities simultaneously hold:

$$\begin{aligned} \|\nabla F(\boldsymbol{\Theta}^*)\|_\infty \leqslant & 2\left[C_3(3\log p + \log n) + (3\log p + \log n)^2\right]\sqrt{\frac{\log p}{n}} \\ & + 2C_4\left(C_1 + \sqrt{\frac{2\log p}{n}}\right), \end{aligned} \tag{7.18}$$

$$\|\tilde{\boldsymbol{\Theta}}_S - \boldsymbol{\Theta}^*_S\|_\infty \leqslant r. \tag{7.19}$$

Proof. We prove (7.18) and (7.19) in turn.

Proof of (7.18)

To begin with, we prove (7.18). By the definition of F in (7.13), for $j < j'$, the derivative of $F(\boldsymbol{\Theta}^*)$ is:

$$\begin{aligned} \frac{\partial F(\boldsymbol{\Theta}^*)}{\partial \theta_{jj'}} =& \frac{1}{n}\sum_{i=1}^n \left[-x_{ij'}x_{ij} + \lambda^*_{ij}x_{ij'} - x_{ij}x_{ij'} + \lambda^*_{ij'}x_{ij}\right] \\ =& -2\mathbb{E}_{\mathbb{X}}[X_jX_{j'}] + \frac{1}{n}\sum_{i=1}^n \lambda^*_{ij}x_{ij'} + \frac{1}{n}\sum_{i=1}^n \lambda^*_{ij'}x_{ij}. \end{aligned} \tag{7.20}$$

and

$$\frac{\partial}{\partial \theta_{jj}} F(\boldsymbol{\Theta}^*) = \frac{1}{n}\sum_{i=1}^n \left[-x_{ij} + \lambda^*_{ij}\right] = -\mathbb{E}_{\mathbb{X}}[X_j] + \frac{1}{n}\sum_{i=1}^n \lambda^*_{ij}, \tag{7.21}$$

where $\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] := \frac{1}{n}\sum_{i=1}^{n} x_{ij}x_{ij'}$ and $\mathbb{E}_{\mathbb{X}}[X_j] := \frac{1}{n}\sum_{i=1}^{n} x_{ij}$ are the expectations of $X_j X_{j'}$ and X_j over the empirical distribution given by the dataset $\mathbb{X}$.

Then, by defining $\mathbb{E}[X_j X_{j'}]$ as the expectation of the multiplication of two components of an multivariate square root Poisson random vector whose distribution is parameterized by $\boldsymbol{\Theta}^*$, and by Assumption 2, (7.20) can be controlled via

$$
\begin{aligned}
&|\frac{\partial}{\partial \theta_{jj'}} F(\boldsymbol{\Theta}^*)| \\
=&|\frac{1}{n}\sum_{i=1}^{n} \lambda^*_{ij} x_{ij'} - \mathbb{E}[X_j X_{j'}] + \frac{1}{n}\sum_{i=1}^{n} \lambda^*_{ij'} x_{ij} - \mathbb{E}[X_j X_{j'}] \\
&+2\mathbb{E}[X_j X_{j'}] - 2\mathbb{E}_{\mathbb{X}}[X_j X_{j'}]| \\
\leqslant&|\frac{1}{n}\sum_{i=1}^{n} \lambda^*_{ij} x_{ij'} - \mathbb{E}[X_j X_{j'}]| + |\frac{1}{n}\sum_{i=1}^{n} \lambda^*_{ij'} x_{ij} - \mathbb{E}[X_j X_{j'}]| \\
&+2|\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] - \mathbb{E}[X_j X_{j'}]| \\
=&|\frac{1}{n}\sum_{i=1}^{n} \left(\mathbb{E}[X_j \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}] + \lambda^*_{ij} - \mathbb{E}[X_j \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}]\right) x_{ij'} - \mathbb{E}[X_j X_{j'}]| \\
&+|\frac{1}{n}\sum_{i=1}^{n} \left(\mathbb{E}[X_{j'} \mid \mathbf{X}_{-j'} = \mathbf{x}_{i,-j'}] + \lambda^*_{ij'} - \mathbb{E}[X_{j'} \mid \mathbf{X}_{-j'} = \mathbf{x}_{i,-j'}]\right) x_{ij} - \mathbb{E}[X_j X_{j'}]| \\
&+2|\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] - \mathbb{E}[X_j X_{j'}]| \\
\leqslant&|\frac{1}{n}\sum_{i=1}^{n} \left(\mathbb{E}[X_j \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}]\right) x_{ij'} - \mathbb{E}[X_j X_{j'}]| + \frac{1}{n}\sum_{i=1}^{n} |\lambda^*_{ij} - \mathbb{E}[X_j \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}]| x_{ij'} \\
&+|\frac{1}{n}\sum_{i=1}^{n} \left(\mathbb{E}[X_{j'} \mid \mathbf{X}_{-j'} = \mathbf{x}_{i,-j'}]\right) x_{ij} - \mathbb{E}[X_j X_{j'}]| + \frac{1}{n}\sum_{i=1}^{n} |\lambda^*_{ij'} - \mathbb{E}[X_{j'} \mid \mathbf{X}_{-j'} = \mathbf{x}_{i,-j'}]| x_{ij} \\
&+2|\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] - \mathbb{E}[X_j X_{j'}]| \\
\leqslant&|\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[X_j \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}] x_{ij'} - \mathbb{E}[X_j X_{j'}]| + |\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[X_{j'} \mid \mathbf{X}_{-j'} = \mathbf{x}_{i,-j'}] x_{ij} - \mathbb{E}[X_j X_{j'}]| \\
&+2|\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] - \mathbb{E}[X_j X_{j'}]| + C_4(\mathbb{E}_{\mathbb{X}}[X_j] + \mathbb{E}_{\mathbb{X}}[X_{j'}]) \\
=&|\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[X_j X_{j'} \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}] - \mathbb{E}[X_j X_{j'}]| + |\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[X_j X_{j'} \mid \mathbf{X}_{-j'} = \mathbf{x}_{i,-j'}] - \mathbb{E}[X_j X_{j'}]|
\end{aligned}
$$

$$
\begin{aligned}
&+2|\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] - \mathbb{E}[X_j X_{j'}]| + C_4(\mathbb{E}_{\mathbb{X}}[X_j] + \mathbb{E}_{\mathbb{X}}[X_{j'}]) \\
&=2|\mathbb{E}_{\mathbb{X}}[\mathbb{E}[X_j X_{j'} \mid \mathbf{X}_{-j}]] - \mathbb{E}[X_j X_{j'}]| + 2|\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] - \mathbb{E}[X_j X_{j'}]| + C_4(\mathbb{E}_{\mathbb{X}}[X_j] + \mathbb{E}_{\mathbb{X}}[X_{j'}]) \\
&=2|\mathbb{E}_{\mathbb{X}}[\mathbb{E}[X_j X_{j'} \mid \mathbf{X}_{-j}]] - \mathbb{E}[\mathbb{E}[X_j X_{j'} \mid \mathbf{X}_{-j}]]| + 2|\mathbb{E}_{\mathbb{X}}[X_j X_{j'}] - \mathbb{E}[X_j X_{j'}]| \\
&+C_4(\mathbb{E}_{\mathbb{X}}[X_j] + \mathbb{E}_{\mathbb{X}}[X_{j'}]),
\end{aligned}
$$

where we have used the law of total expectation in the last equality.

Similarly, (7.21) can be controlled via

$$
\begin{aligned}
|\frac{\partial}{\partial \theta_{jj}} F(\mathbf{\Theta}^*)| =& |-\mathbb{E}_{\mathbb{X}}[X_j] + \frac{1}{n}\sum_{i=1}^{n} \lambda_{ij}^*| = |-\mathbb{E}_{\mathbb{X}}[X_j] \\
&+\frac{1}{n}\sum_{i=1}^{n} \left(\mathbb{E}[X_j \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}] + \lambda_{ij}^* - \mathbb{E}[X_j \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}]\right)| \\
=& |-\mathbb{E}_{\mathbb{X}}[X_j] + \mathbb{E}[X_j] - \mathbb{E}[X_j] + \mathbb{E}_{\mathbb{X}}[\mathbb{E}[X_j \mid \mathbf{X}_{-j}]] \\
&+\frac{1}{n}\sum_{i=1}^{n} \left(\lambda_{ij}^* - \mathbb{E}[X_j \mid \mathbf{X}_{-j} = \mathbf{x}_{i,-j}]\right)| \\
\leqslant& |\mathbb{E}_{\mathbb{X}}[\mathbb{E}[X_j \mid \mathbf{X}_{-j}]] - \mathbb{E}[X_j]| + |\mathbb{E}_{\mathbb{X}}[X_j] - \mathbb{E}[X_j]| + C_4 \\
=& |\mathbb{E}_{\mathbb{X}}[\mathbb{E}[X_j \mid \mathbf{X}_{-j}]] - \mathbb{E}[\mathbb{E}[X_j \mid \mathbf{X}_{-j}]]| + |\mathbb{E}_{\mathbb{X}}[X_j] - \mathbb{E}[X_j]| + C_4.
\end{aligned}
$$

We define four events:

$$
\begin{aligned}
E_1 &:= \left\{ \max_{j \neq j', j, j' \in \{1,2,\cdots,p\}} |\frac{\partial}{\partial \theta_{jj'}} F(\mathbf{\Theta}^*)| \geqslant 2(C_3\epsilon_1 + \epsilon_1^2)\sqrt{\frac{\log p}{n}} + 2C_4\epsilon_2 \right\}, \\
E_2 &:= \left\{ \max_{j \in \{1,2,\cdots,p\}} |\frac{\partial}{\partial \theta_{jj}} F(\mathbf{\Theta}^*)| \geqslant (C_3 + \epsilon_1)\sqrt{\frac{\log p}{n}} + C_4/n \right\}, \\
E_3 &:= \left\{ 0 < \max_{i \in \{1,2,\cdots,n\}} \|\mathbf{x}_i\|_\infty < \epsilon_1 \right\}, \quad \text{and} \quad E_4 := \left\{ 0 < \max_{j \in \{1,2,\cdots,p\}} \mathbb{E}_{\mathbb{X}}[X_j] < \epsilon_2 \right\},
\end{aligned}
$$

where $\epsilon_1 := 3\log p + \log n$ and $\epsilon_2 := C_1 + \sqrt{\frac{2\log p}{n}}$ are defined in Lemma 7.8. By

Lemma 7.4, Lemma 7.5, Lemma 7.6 and Lemma 7.7, it follows that

$$P(E_1 \mid E_3, E_4) \leqslant 4\exp(-2\log p) \quad \text{and} \quad P(E_2 \mid E_3, E_4) \leqslant 4\exp(-2\log p). \tag{7.22}$$

Therefore,

$$\begin{aligned}
P(E_1 \cup E_2) =& P(E_1 \cup E_2 \mid E_3, E_4)P(E_3, E_4) + P(E_1 \cup E_2 \mid E_3^c, E_4)P(E_3^c, E_4) \\
&+ P(E_1 \cup E_2 \mid E_3, E_4^c)P(E_3, E_4^c) + P(E_1 \cup E_2 \mid E_3^c, E_4^c)P(E_3^c, E_4^c) \\
\leqslant& P(E_1 \mid E_3, E_4) + P(E_2 \mid E_3, E_4) + P(E_3^c, E_4) + P(E_3, E_4^c) + P(E_3^C, E_4^c) \\
\leqslant& P(E_1 \mid E_3, E_4) + P(E_2 \mid E_3, E_4) + P(E_3^c) + P(E_4^c) \\
\leqslant& 8\exp(-2\log p) + \exp(C_1 + C_2/2 - \epsilon_1)np + \exp\left[-\frac{n(\epsilon_2 - C_1)^2}{2C_2}\right] \\
=& 8\exp(-2\log p) + \frac{\exp(C_1 + C_2/2)}{p^2} + p^{-\frac{1}{C_2}},
\end{aligned} \tag{7.23}$$

where the superscript c over an event represents the complement of that event, and the last inequality is due to (7.22) and Lemma 7.8. Also notice that by the definitions of E_1 and E_2,

$$2(C_3\epsilon_1 + \epsilon_1^2)\sqrt{\frac{\log p}{n}} + 2C_4\epsilon_2 > (C_3 + \epsilon_1)\sqrt{\frac{\log p}{n}} + C_4/n.$$

Therefore, with probability of $1 - P(E_1 \cup E_2) \geqslant 1 - \left((\exp(C_1 + C_2/2) + 8)\, p^{-2} + p^{-1/C_2}\right)$, neither E_1 nor E_2 occurs, and hence

$$\begin{aligned}
\|\boldsymbol{\nabla} F(\boldsymbol{\Theta}^*)\|_\infty \leqslant& 2(C_3\epsilon_1 + \epsilon_1^2)\sqrt{\frac{\log p}{n}} + 2C_4\epsilon_2 \\
=& 2\left[C_3(3\log p + \log n) + (3\log p + \log n)^2\right]\sqrt{\frac{\log p}{n}} \\
&+ 2C_4\left(C_1 + \sqrt{\frac{2\log p}{n}}\right).
\end{aligned}$$

Proof of (7.19)

Then, we study (7.19). We consider a map defined as

$$G(\boldsymbol{\Delta}_S) := -\mathbf{H}_{SS}^{-1}\left[\boldsymbol{\nabla}_S F(\boldsymbol{\Theta}^* + \boldsymbol{\Delta}_S) + \lambda \hat{\mathbf{Z}}_S\right] + \boldsymbol{\Delta}_S.$$

If $\|\boldsymbol{\Delta}\|_\infty \leqslant r$, by Taylor expansion of $\boldsymbol{\nabla}_S F(\boldsymbol{\Theta}^* + \boldsymbol{\Delta})$ centered at $\boldsymbol{\nabla}_S F(\boldsymbol{\Theta}^*)$,

$$\begin{aligned}
\|G(\boldsymbol{\Delta}_S)\|_\infty &= \left\|-\mathbf{H}_{SS}^{-1}\left[\boldsymbol{\nabla}_S F(\boldsymbol{\Theta}^*) + \mathbf{H}_{SS}\boldsymbol{\Delta}_S + \mathbf{R}_S(\boldsymbol{\Delta}) + \lambda \hat{\mathbf{Z}}_S\right] + \boldsymbol{\Delta}_S\right\|_\infty \\
&= \left\|-\mathbf{H}_{SS}^{-1}\left(\boldsymbol{\nabla}_S F(\boldsymbol{\Theta}^*) + \mathbf{R}_S(\boldsymbol{\Delta}) + \lambda \hat{\mathbf{Z}}_S\right)\right\|_\infty \\
&\leqslant \left\|\mathbf{H}_{SS}^{-1}\right\|_\infty \left(\|\boldsymbol{\nabla}_S F(\boldsymbol{\Theta}^*)\|_\infty + \|R_S(\boldsymbol{\Delta})\|_\infty + \lambda\|\hat{\mathbf{Z}}_S\|_\infty\right) \\
&\leqslant (C_5(\lambda + C_6 r^2 + \lambda) = C_5 C_6 r^2 + 2C_5\lambda,
\end{aligned}$$

where the inequality is due to $\|\boldsymbol{\nabla}_S F(\boldsymbol{\Theta}^*)\|_\infty \leqslant \lambda$ conditioning on $E_1^c \cap E_2^c$ and according to (7.18). Then, based on the definition of r, we can derive the upper bound of $\|G(\boldsymbol{\Delta}_S)\|_\infty$ as $\|G(\boldsymbol{\Delta}_S)\|_\infty \leqslant r/2 + r/2 = r$.

Therefore, according to the fixed point theorem (Ortega and Rheinboldt, 2000; Yang and Ravikumar, 2011), there exists $\boldsymbol{\Delta}_S$ satisfying $G(\boldsymbol{\Delta}_S) = \boldsymbol{\Delta}_S$, which indicates $\boldsymbol{\nabla}_S F(\boldsymbol{\Theta}^* + \boldsymbol{\Delta}) + \lambda \hat{\mathbf{Z}}_S = \mathbf{0}$. Considering that the optimal solution to (7.25) is unique, $\tilde{\boldsymbol{\Delta}}_S = \boldsymbol{\Delta}_S$, whose infinite norm is bounded by $\|\tilde{\boldsymbol{\Delta}}_S\|_\infty \leqslant r$, with probability larger than $1 - \left((\exp(C_1 + C_2/2) + 8)\, p^{-2} + p^{-1/C_2}\right)$. □

Lemma 7.10. *Let* $\hat{\boldsymbol{\Theta}}$ *be an optimal solution to* (7.12)*, and* $\hat{\mathbf{Z}}$ *be the corresponding dual solution. If* $\hat{\mathbf{Z}}$ *satisfies* $\|\hat{\mathbf{Z}}_I\|_\infty < 1$*, then any given optimal solution to* (7.12) $\tilde{\boldsymbol{\Theta}}$ *satisfies* $\tilde{\boldsymbol{\Theta}}_I = \mathbf{0}$*. Moreover, if* $\mathbf{H}_{SS}$ *is positive definite, then the solution to* (7.12) *is unique.*

Proof. Specifically, following the same rationale as Lemma 1 in Wainwright 2009b, Lemma 1 in Ravikumar et al. 2010, and Lemma 2 in Yang and Ravikumar 2011, we can derive Lemma 7.10 characterizing the optimal solution of (7.12). □

7.8.4 Proof of Theorem 7.1

The proof follows the primal-dual witness (PDW) technique, which is widely used in this line of research (Wainwright, 2009b; Ravikumar et al., 2010; Yang and Ravikumar, 2011; Yang et al., 2015a). Specifically, by Lemma 7.10, we can prove the sparsistency by building an optimal solution to (7.12) satisfying $\|\hat{\mathbf{Z}}_{\mathrm{I}}\|_\infty < 1$, which is summarized as *strict dual feasibility* (SDF). To this end, we apply PDW to build a qualified optimal solution with the assumption that $\mathbf{H}_{SS}$ is positively definite.

Solve a Restricted Problem

First of all, we derive the KKT condition of (7.12):

$$\nabla \mathrm{F}(\hat{\boldsymbol{\Theta}}) + \lambda \hat{\mathbf{Z}} = \mathbf{0}. \tag{7.24}$$

To construct an optimal optimal primal-dual pair solution, we define $\tilde{\boldsymbol{\Theta}}$ as an optimal solution to the restricted problem:

$$\tilde{\boldsymbol{\Theta}} := \min_{\boldsymbol{\Theta}} \mathrm{F}(\boldsymbol{\Theta}) + \lambda \|\boldsymbol{\Theta}\|_1, \tag{7.25}$$

with $\boldsymbol{\Theta}_{\mathrm{I}} = \mathbf{0}$, where $\tilde{\boldsymbol{\Theta}}$ is unique according to Lemma 7.10 with the assumption that $\mathbf{H}_{SS} \succ \mathbf{0}$. Denote the subgradient corresponding to $\tilde{\boldsymbol{\Theta}}$ as $\tilde{\mathbf{Z}}$. Then $(\tilde{\boldsymbol{\Theta}}, \tilde{\mathbf{Z}})$ is optimal for the restricted problem (7.25). Therefore, $\tilde{\mathbf{Z}}_S$ can be determined according to the values of $\tilde{\boldsymbol{\Theta}}_S$ via the KKT conditions of (7.25). As a result,

$$\nabla_S \mathrm{F}(\tilde{\boldsymbol{\Theta}}) + \lambda \tilde{\mathbf{Z}}_S = \mathbf{0}, \tag{7.26}$$

where ∇_S represents the gradient components with respect to S. Furthermore, by letting $\hat{\boldsymbol{\Theta}} = \tilde{\boldsymbol{\Theta}}$, we determine $\tilde{\mathbf{Z}}_{\mathrm{I}}$ according to (7.24). It remains to show that $\tilde{\mathbf{Z}}_{\mathrm{I}}$ satisfies SDF.

Check SDF

Now, we demonstrate that $\tilde{\mathbf{\Theta}}$ and $\tilde{\mathbf{Z}}$ satisfy SDF. By (7.26), and by the Taylor expansion of $\nabla_S F(\tilde{\mathbf{\Theta}})$, we have that

$$\begin{aligned} &\mathbf{H}_{SS}\tilde{\mathbf{\Delta}}_S + \nabla_S F(\mathbf{\Theta}^*) + \mathbf{R}_S(\tilde{\mathbf{\Delta}}) + \lambda\tilde{\mathbf{Z}}_S = \mathbf{0} \\ \Rightarrow &\tilde{\mathbf{\Delta}}_S = \mathbf{H}_{SS}^{-1}\left[-\nabla_S F(\mathbf{\Theta}^*) - \mathbf{R}_S(\tilde{\mathbf{\Delta}}) - \lambda\tilde{\mathbf{Z}}_S\right], \end{aligned} \tag{7.27}$$

where $\tilde{\mathbf{\Delta}} := \tilde{\mathbf{\Theta}} - \mathbf{\Theta}^*$, $\mathbf{R}_S(\tilde{\mathbf{\Delta}})$ represents the components of $\mathbf{R}(\mathbf{\Delta})$ corresponding to S, and we have used the fact that $\mathbf{H}_{SS}$ is positive definite and hence invertible. By the definition of $\tilde{\mathbf{\Theta}}$ and $\tilde{\mathbf{Z}}$,

$$\begin{aligned} \nabla F(\tilde{\mathbf{\Theta}}) + \lambda\tilde{\mathbf{Z}} = \mathbf{0} \Rightarrow &\nabla F(\mathbf{\Theta}^*) + \mathbf{H}\tilde{\mathbf{\Delta}} + \mathbf{R}(\tilde{\mathbf{\Delta}}) + \lambda\tilde{\mathbf{Z}} = \mathbf{0} \\ \Rightarrow &\nabla_I F(\tilde{\mathbf{\Theta}}) + \mathbf{H}_{IS}\tilde{\mathbf{\Delta}}_S + \mathbf{R}_I(\tilde{\mathbf{\Delta}}) + \lambda\tilde{\mathbf{Z}}_I = \mathbf{0}, \end{aligned} \tag{7.28}$$

where $\mathbf{R}_I(\tilde{\mathbf{\Delta}})$ represents the components of $\mathbf{R}(\mathbf{\Delta})$ corresponding to I, and we have used the fact that $\tilde{\mathbf{\Delta}}_I = \mathbf{0}$ because $\tilde{\mathbf{\Theta}}_I = \mathbf{\Theta}^* = \mathbf{0}$. As a result,

$$\begin{aligned} \lambda\|\tilde{\mathbf{Z}}_I\|_\infty =& \|-\mathbf{H}_{IS}\tilde{\mathbf{\Delta}}_S - \nabla_I F(\mathbf{\Theta}^*) - \mathbf{R}_I(\tilde{\mathbf{\Delta}})\|_\infty \\ \leqslant& \left\|\mathbf{H}_{IS}\mathbf{H}_{SS}^{-1}\left[-\nabla_S F(\mathbf{\Theta}^*) - \mathbf{R}_S(\tilde{\mathbf{\Delta}}) - \lambda\tilde{\mathbf{Z}}_S\right]\right\|_\infty + \|\nabla_I F(\mathbf{\Theta}^*) + \mathbf{R}_I(\tilde{\mathbf{\Delta}})\|_\infty \\ \leqslant& \left\|\mathbf{H}_{IS}\mathbf{H}_{SS}^{-1}\right\|_\infty \left\|\nabla_S F(\mathbf{\Theta}^*) + \mathbf{R}_S(\tilde{\mathbf{\Delta}})\right\|_\infty + \left\|\mathbf{H}_{IS}\mathbf{H}_{SS}^{-1}\right\|_\infty \left\|\lambda\tilde{\mathbf{Z}}_S\right\|_\infty \\ &+\|\nabla_I F(\mathbf{\Theta}^*) + \mathbf{R}_I(\tilde{\mathbf{\Delta}})\|_\infty \\ \leqslant& (1-\alpha)\left(\|\nabla_S F(\mathbf{\Theta}^*)\|_\infty + \|\mathbf{R}_S(\tilde{\mathbf{\Delta}})\|_\infty\right) + (1-\alpha)\lambda \\ &+ \left(\|\nabla_I F(\mathbf{\Theta}^*)\|_\infty + \|\mathbf{R}_I(\tilde{\mathbf{\Delta}})\|_\infty\right) \\ \leqslant& (2-\alpha)\left(\|\nabla F(\mathbf{\Theta}^*)\|_\infty + \|\mathbf{R}(\tilde{\mathbf{\Delta}})\|_\infty\right) + (1-\alpha)\lambda, \end{aligned} \tag{7.29}$$

where we have used (7.27) in the first inequality, and the third inequality is due to Assumption 3.

With (7.29), it remains to control $\|\nabla F(\mathbf{\Theta}^*)\|_\infty$ and $\|\mathbf{R}(\tilde{\mathbf{\Delta}})\|_\infty$. On one hand, according to Lemma 7.9 and the assumption on λ in Theorem 7.1, $\|\nabla F(\mathbf{\Theta}^*)\|_\infty \leqslant 2\left[3C_3\log p + C_3\log n + (3\log p + \log n)^2\right]\sqrt{\frac{\log p}{n}} + 2C_4\left(C_1 + \sqrt{\frac{2\log p}{n}}\right) \leqslant \frac{\alpha\lambda}{4}$, with

probability larger than $1-\left((\exp\left(C_1+C_2/2\right)+8)\,p^{-2}+p^{-1/C_2}\right)$.

On the other hand, according to Assumption 4 and Lemma 7.9,

$$\begin{aligned}\|\mathbf{R}(\tilde{\boldsymbol{\Delta}})\|_\infty \leqslant & C_6\|\boldsymbol{\Delta}\|_\infty^2 \leqslant C_6 r^2 \leqslant C_6(4C_5\lambda)^2 \\ = & \lambda\frac{64C_5^2C_6}{\alpha}\frac{\alpha\lambda}{4} \leqslant \left(C_7\sqrt{\frac{\log^5 p}{n}}\right)\frac{64C_5^2C_6}{\alpha}\frac{\alpha\lambda}{4},\end{aligned} \tag{7.30}$$

where in the last inequality we have used the assumption $\lambda \propto \sqrt{\frac{\log^5 p}{n}}$ in Theorem 7.1, and hence there exists C_7 satisfying $\lambda \leqslant C_7\sqrt{\frac{\log^5 p}{n}}$. Therefore, when we choose $n \geqslant \left(64C_7C_5^2C_6/\alpha\right)^2\log^5 p$ as assumed in Theorem 1, then from (7.30), we can conclude that $\|\mathbf{R}(\tilde{\boldsymbol{\Delta}})\|_\infty \leqslant \frac{\alpha\lambda}{4}$. As a result, $\lambda\|\hat{\mathbf{Z}}_I\|_\infty$ can be bounded by $\lambda\|\tilde{\mathbf{Z}}_I\|_\infty < \alpha\lambda/2 + \alpha\lambda/2 + (1-\alpha)\lambda = \lambda$. Combined with Lemma 7.10, we demonstrate that any optimal solution of (7.12) satisfies $\tilde{\boldsymbol{\Theta}}_I = \mathbf{0}$. Furthermore, (7.19) controls the difference between the optimal solution of (7.12) and the real parameter by $\|\tilde{\boldsymbol{\Delta}}_S\|_\infty \leqslant r$, by the fact that $r \leqslant \|\boldsymbol{\Theta}_S^*\|_\infty$ in Theorem 7.1, $\hat{\boldsymbol{\Theta}}_S$ shares the same sign with $\boldsymbol{\Theta}_S^*$.

8 STOCHASTIC LEARNING FOR SPARSE DISCRETE MARKOV RANDOM FIELDS WITH CONTROLLED GRADIENT APPROXIMATION ERROR

8.1 Introduction

Markov random fields (MRFs, a.k.a. Markov networks, undirected graphical models) are a compact representation of the joint distribution among multiple variables, with each variable being a node and an edge between two nodes indicating conditional dependence between the two corresponding variables. Sparse discrete MRF learning is proposed in the seminal work of Lee et al. (2006). By considering an L_1-regularized MLE problem, many components of the parameterization are driven to zero, yielding a sparse solution to structure learning. However, in general, solving an L_1-regularized MLE problem exactly for a discrete MRF is infamously difficult due to the NP-hard inference problem posed by exact gradient evaluation (Koller and Friedman, 2009). We hence inevitably have to compromise accuracy for the gain of efficiency and scalability via *inexact* learning techniques.

In this chapter, we consider stochastic proximal gradient (SPG; Honorio 2012a; Atchade et al. 2014; Miasojedow and Rejchel 2016), a stochastic learning framework for L_1-regularized discrete MRFs. SPG hinges on a stochastic oracle for gradient approximation of the log-likelihood function (inexact inference). However, both the theoretical guarantees and the practical performances of existing algorithms are unsatisfactory.

The stochastic oracle behind SPG is Gibbs sampling (Levin et al., 2009), which is an effective approach to draw samples from an intractable probability distribution. With enough samples, the intractable distribution can be approximated effectively by the empirical distribution, and hence many quantities (e.g., the gradient of the log-likelihood function) related to the intractable distribution can be estimated efficiently. Since SPG uses Gibbs sampling for gradient approximation, it can be viewed as an inexact proximal gradient method (Schmidt et al., 2011), whose

success depends on whether the gradient approximation error can be effectively controlled. While previous works (Honorio, 2012a; Atchade et al., 2014; Miasojedow and Rejchel, 2016) have shown that the quality of the gradient approximation can be improved *in the long run* with increasingly demanding computational resources, such long term guarantees might not translate to satisfactory performance in practice (see Section 8.8). Therefore, it is desirable to estimate and control the gradient approximation error of SPG meticulously in each iteration so that a more refined approximation to the exact gradient will be rewarded with a higher gain of efficiency and accuracy in practice.

Careful analysis and control of the quality of the gradient approximation of SPG call for the cross-fertilization of theoretical and empirical insights from stochastic approximate inference (Bengio and Delalleau, 2009; Fischer and Igel, 2011), inexact proximal methods (Schmidt et al., 2011), and statistical sampling (Mitliagkas and Mackey, 2017). Our contributions are hence both theoretical and empirical. Theoretically, we provide novel *verifiable* bounds (Section 8.4) to inspect and control the gradient approximation error induced by Gibbs sampling. Also, we provide a proof sketch for the main results in Section 8.5. Empirically, we propose the *tighten asymptotically* (TAY) learning strategy (Section 8.6) based on the verifiable bounds to boost the performance of SPG.

8.2 Background

We first introduce L_1-regularized discrete MRFs in Section 8.2.1. We then briefly review SPG as a combination of proximal gradient for sparse statistical learning and Gibbs sampling for addressing the intractable exact gradient evaluation problem.

8.2.1 L_1-Regularized Discrete MRF

For the derivation, we focus on the binary pairwise case and we illustrate that our framework can be generalized to other models in Section 8.6. Let

$$\mathbf{X} = \begin{bmatrix} X_1, X_2, \cdots, X_p \end{bmatrix}^\top \in \{0,1\}^p$$

be a $p \times 1$ binary random vector. We use an uppercase letter such as X to denote a random variable and the corresponding lowercase letter to denote a particular *assignment* of the random variable, i.e., $X = x$. We also use boldface letters to represent vectors and matrices and regular letters to represent scalars. We define the function $\boldsymbol{\psi} : \{0,1\}^p \to \{0,1\}^m,\ \mathbf{x} \to \boldsymbol{\psi}(\mathbf{x})$ to represent the *sufficient statistics* (a.k.a. *features*) whose values depend on the assignment $\mathbf{x}$ and compose an $m \times 1$ vector $\boldsymbol{\psi}(\mathbf{x})$, with its j^{th} component denoted as $\psi_j(\mathbf{x})$. We use $\mathbb{X}$ to represent a dataset with n independent and identically distributed (i.i.d.) samples.

With the notation introduced above, the L_1-regularized discrete MRF problem can be formulated as the following convex optimization problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} -\frac{1}{n} \sum_{\mathbf{x} \in \mathbb{X}} \boldsymbol{\theta}^\top \boldsymbol{\psi}(\mathbf{x}) + A(\boldsymbol{\theta}) + \lambda \|\boldsymbol{\theta}\|_1, \tag{8.1}$$

with

$$A(\boldsymbol{\theta}) = \log \sum_{\mathbf{x} \in \{0,1\}^p} \exp(\boldsymbol{\theta}^\top \boldsymbol{\psi}(\mathbf{x})),$$

where $\boldsymbol{\Theta} \subseteq \mathbb{R}^m$ is the parameter space of $\boldsymbol{\theta}$'s, $\lambda \geqslant 0$, and $A(\boldsymbol{\theta})$ is the *log partition function*. We denote the differentiable part of (8.1) as

$$f(\boldsymbol{\theta}) = -\frac{1}{n} \sum_{\mathbf{x} \in \mathbb{X}} \boldsymbol{\theta}^\top \boldsymbol{\psi}(\mathbf{x}) + A(\boldsymbol{\theta}). \tag{8.2}$$

Solving (8.1) requires evaluating the gradient of $f(\boldsymbol{\theta})$, which is given by:

$$\nabla f(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}} \boldsymbol{\psi}(\mathbf{x}) - \mathbb{E}_{\mathbb{X}} \boldsymbol{\psi}(\mathbf{x}), \tag{8.3}$$

Algorithm 3 Gibbs Sampling (Gibbs-1)

Require: initial samples $\mathbb{S}_0$ and θ.
Ensure: $\mathbb{S}$.

function GIBBS-1($\mathbb{S}_0$, θ)
  $\mathbb{S} \leftarrow \mathbb{S}_0$, and decide p from $\mathbb{S}_0$.
  for $i \in \{1, \cdots, p\}$ **do**
    for $x \in \mathbb{S}$ **do**
      Compute $P_\theta(X_i \mid x_{-i})$ according to (8.5).
      Update x_i by $P_\theta(X_i \mid x_{-i})$.
    end for
  end for
  return $\mathbb{S}$.
end function

Algorithm 4 Gradient Approximation (GRAD)

Require: θ, $\mathbb{E}_{\mathbb{X}}\psi(x)$, and q.
Ensure: $\Delta f(\theta)$.

function GRAD(θ, $\mathbb{E}_{\mathbb{X}}\psi(x)$, q)
  Initialize $\mathbb{S}$ with q samples.
  while true **do**
    $\mathbb{S} \leftarrow$ GIBBS-1($\mathbb{S}$, θ).
    if stopping criteria met **then**
      Compute $\mathbb{E}_{\mathbb{S}}\psi(x)$ according to (8.6).
      $\Delta f(\theta) \leftarrow \mathbb{E}_{\mathbb{S}}\psi(x) - \mathbb{E}_{\mathbb{X}}\psi(x)$.
      break.
    end if
  end while
  return $\Delta f(\theta)$.
end function

with

$$\mathbb{E}_\theta\psi(x) = \sum_{x \in \{0,1\}^p} P_\theta(x)\psi(x), \quad \mathbb{E}_{\mathbb{X}}\psi(x) = \frac{1}{n}\sum_{x \in \mathbb{X}} \psi(x). \tag{8.4}$$

$\mathbb{E}_\theta\psi(x)$ represents the expectation of the sufficient statistics under $P_\theta(x) = \frac{\exp(\theta^\top \psi(x))}{\exp(A(\theta))}$, which is a discrete MRF probability distribution parameterized by θ. $\mathbb{E}_{\mathbb{X}}\psi(x)$ rep-

Algorithm 5 Stochastic Proximal Gradient (SPG)

Require: $\mathbb{X}$, λ, and q.
Ensure: $\tilde{\theta}$.

```
function SPG(𝕏, λ, q)
    Compute 𝔼_𝕏 ψ(x) according to (8.4).
    Initialize θ^(0) randomly and k ← 0.
    Choose step length α.
    while true do
        Δf(θ^(k)) ← Grad(θ^(k), 𝔼_𝕏 ψ(x), q).
        θ^(k+1) ← S_αλ(θ^(k) − αΔf(θ^(k))).
        if Stopping criteria met then
            θ̃ = θ^(k+1), return θ̃.
        end if
        k ← k + 1
    end while
end function
```

resents the expectation of the sufficient statistics under the empirical distribution. Computing $\mathbb{E}_{\mathbb{X}}\boldsymbol{\psi}(\mathbf{x})$ is straightforward, but computing $\mathbb{E}_{\theta}\boldsymbol{\psi}(\mathbf{x})$ exactly is intractable due to the entanglement of $A(\theta)$. As a result, various approximations have been made (Wainwright et al., 2007; Höfling and Tibshirani, 2009; Viallon et al., 2014).

8.2.2 Stochastic Proximal Gradient

To efficiently solve (8.1), many efforts have been made in combining Gibbs sampling (Levin et al., 2009) and proximal gradient descent (Parikh et al., 2014) into SPG, a method that adopts the proximal gradient framework to update iterates, but uses Gibbs sampling as a stochastic oracle to approximate the gradient when the gradient information is needed (Honorio, 2012a; Atchade et al., 2014; Miasojedow and Rejchel, 2016).

Specifically, Gibbs sampling with q chains running τ steps (Gibbs-τ) can generate q samples for $P_{\theta}(\mathbf{x})$. Gibbs-τ is achieved by iteratively applying Gibbs-1 for τ times.

Gibbs-1 is summarized in Algorithm 3, where

$$P_\theta(X_i \mid \mathbf{x}_{-i}) = P_\theta(\mathbf{x}_i \mid x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_p) \tag{8.5}$$

represents the conditional distribution of X_i given the assignment of the remaining variables $\mathbf{x}_{-i}$ under the parameterization θ. Denoting the set of these q (potentially repetitive) samples as $\mathbb{S}$, we can approximate $\mathbb{E}_\theta \boldsymbol{\psi}(\mathbf{x})$ by the easily computable

$$\mathbb{E}_{\mathbb{S}} \boldsymbol{\psi}(\mathbf{x}) = \frac{1}{q} \sum_{\mathbf{x} \in \mathbb{S}} \boldsymbol{\psi}(\mathbf{x}) \tag{8.6}$$

and thus reach the approximated gradient $\boldsymbol{\Delta} f(\theta) = \mathbb{E}_{\mathbb{S}} \boldsymbol{\psi}(\mathbf{x}) - \mathbb{E}_{\mathbb{X}} \boldsymbol{\psi}(\mathbf{x})$ with the gradient approximation error:

$$\delta(\theta) = \boldsymbol{\Delta} f(\theta) - \boldsymbol{\nabla} f(\theta).$$

By replacing $\boldsymbol{\nabla} f(\theta)$ with $\boldsymbol{\Delta} f(\theta)$ in proximal gradient, the update rule for SPG can be derived as $\theta^{(k+1)} = \mathcal{S}_{\alpha\lambda}\left(\theta^{(k)} - \alpha \boldsymbol{\Delta} f(\theta^{(k)})\right)$, where $\alpha > 0$ is the step length and $\mathcal{S}_\lambda(\mathbf{a})$ is the soft-thresholding operator whose value is also an $m \times 1$ vector, with its i^{th} component defined as $\mathcal{S}_\lambda(\mathbf{a})_i = \operatorname{sgn}(a_i) \max(0, |a_i| - \lambda)$ and $\operatorname{sgn}(a_i)$ is the sign function.

By defining

$$\mathbf{G}_\alpha(\theta^{(k)}) := \frac{1}{\alpha}\left(\theta^{(k)} - \theta^{(k+1)}\right) = \frac{1}{\alpha}\left(\theta^{(k)} - S_{\alpha\lambda}\left(\theta^{(k)} - \alpha \boldsymbol{\Delta} f(\theta^{(k)})\right)\right), \tag{8.7}$$

we can rewrite the previous update rule in a form analogous to the update rule of a standard gradient descent, resulting in the update rule of a *generalized gradient descent* algorithm:

$$\theta^{(k+1)} = \theta^{(k)} - \alpha \mathbf{G}_\alpha(\theta^{(k)}). \tag{8.8}$$

SPG is summarized in Algorithm 5. Its gradient evaluation procedure based on Algorithm 3 is given in Algorithm 4.

8.3 Motivation

Both practical performance and theoretical guarantees of SPG are still far from satisfactory. Empirically, there are no convincing schemes for selecting τ and q, which hinders the efficiency and accuracy of SPG. Theoretically, to the best of our knowledge, existing non-asymptotic convergence rate guarantees can only be achieved for SPG with an averaging scheme (Schmidt et al., 2011; Honorio, 2012a; Atchade et al., 2014) (see also Section 8.3.3), instead of ordinary SPG. In contrast, in the exact proximal gradient descent method, the objective function value is non-decreasing and convergent to the optimal value under some mild assumptions (Parikh et al., 2014). In Section 8.3.2, we identify that the absence of non-asymptotic convergence rate guarantee for SPG primarily comes from the existence of gradient approximation error $\delta(\theta)$. In Section 8.3.3, we further validate that the objective function value achieved by SPG is also highly dependent on $\delta(\theta)$. These issues bring about the demand of inspecting and controlling $\delta(\theta)$ in each iteration.

8.3.1 Setup and Assumptions

For the ease of presentation, we rewrite the objective function in (8.1) as $g(\theta) = f(\theta) + h(\theta)$, where $h(\theta) = \lambda \|\theta\|_1$, and $f(\theta)$ is given in (8.2). Since $\nabla f(\theta)$ is Lipschitz continuous (Honorio, 2012b), we denote its Lipschitz constant as L. We also make the same assumption that $\alpha \leqslant 1/L$ as Schmidt et al. (2011).

8.3.2 Decreasing Objective

It is well-known that exact proximal gradient enjoys a $O\left(\frac{1}{k}\right)$ convergence rate (Parikh et al., 2014). One premise for this convergence result is that the objective function value decreases in each iteration. However, satisfying the decreasing condition is much more intricate in the context of SPG. Theorem 8.1 clearly points out that $\delta(\theta)$ is one main factor determining whether the objective function decreases in SPG.

Theorem 8.1. Let $\theta^{(k)}$ be the iterate of SPG after the k^{th} iteration. Let $\theta^{(k+1)}$ be defined as in (8.8). With $\alpha \leqslant 1/L$, we have

$$g(\theta^{(k+1)}) - g(\theta^{(k)}) \leqslant \alpha\delta(\theta^{(k)})^\top \mathbf{G}_\alpha(\theta^{(k)}) - \frac{\alpha}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2^2.$$

Furthermore, a sufficient condition for $g(\theta^{(k+1)}) < g(\theta^{(k)})$ is

$$\|\delta(\theta^{(k)})\|_2 < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2.$$

According to Theorem 8.1, if the magnitude of the noise, quantified by $\|\delta(\theta^{(k)})\|_2$, is reasonably small, the objective function value decreases in each iteration. Under this condition, we can further construct a theoretical support for the convergence rate of the objective function value in the Section 8.3.3.

8.3.3 Convergence Rate

Assuming that $\delta(\theta)$ is small enough in each iteration to generate a decreasing objective value sequence, we can derive Theorem 8.2 following Proposition 1 in Schmidt et al. (2011):

Theorem 8.2. Let $\mathcal{K} = (\theta^{(0)}, \theta^{(1)}, \theta^{(2)}, \cdots, \theta^{(\kappa)})$ be the iterates generated by Algorithm 5. Then if $g(\theta^{(k+1)}) \leqslant g(\theta^{(k)})$ with $k \in \{1, 2, \cdots, \kappa - 1\}$, we have

$$g(\theta^{(\kappa)}) - g(\hat{\theta}) \leqslant \frac{L}{2\kappa}\left(\|\theta^{(0)} - \hat{\theta}\|_2 + \frac{2}{L}\sum_{k=1}^{\kappa}\|\delta(\theta^{(k)})\|_2\right)^2. \tag{8.9}$$

Recall that $\hat{\theta}$ is an optimal solution to the sparse MLE problem defined in (8.1). From (8.9), it is obvious that if the gradient approximation error is reasonably small, then during the early iterations of SPG, $\|\theta^{(0)} - \hat{\theta}\|_2$ dominates $\frac{2}{L}\sum_{k=1}^{\kappa}\|\delta(\theta^{(k)})\|_2$. Therefore, in the beginning, the convergence rate is $O(1/\kappa)$. However, as the iteration proceeds, $\frac{2}{L}\sum_{k=1}^{\kappa}\|\delta(\theta^{(k)})\|_2$ accumulates and hence in practice SPG can only

maintain a convergence rate of $O(1/\kappa)$ up to some noise level that is closely related to $\delta(\theta^{(k)})$. Therefore, $\delta(\theta^{(k)})$ plays an importance role in the performance of SPG.

Notice that Theorem 8.2 offers convergence analysis of the objective function value in the last iteration $g(\theta^{(\kappa)})$. This result is different from the existing non-asymptotic analysis on $g(\sum_{k=1}^{\kappa} \theta^{(k)}/\kappa)$, the objective function evaluated on the average of all the visited solutions (Schmidt et al., 2011; Honorio, 2012a; Atchade et al., 2014). Theorem 8.2 is more practical than previous analysis, since $\sum_{k=1}^{\kappa} \theta^{(k)}/\kappa$ is a dense parameterization not applicable to structure learning.

According to the analysis above, we need to control $\delta(\theta^{(k)})$ in each iteration to achieve a decreasing and $O\left(\frac{1}{k}\right)$-converging objective function value sequence. Therefore, we focus on checkable bounds for gradient approximation error in Section 8.4.

8.4 Main Results: Bounding the Gradient Approximation Error

In this section, we derive an asymptotic and a non-asymptotic bound to control the gradient approximation error $\delta(\theta^{(k)})$ in each iteration. For this purpose, we consider an arbitrary θ, and perform gradient approximation via Gibbs-τ using Algorithm 4, given an initial value for the Gibbs sampling algorithm, $\tilde{x}_0$. By bounding $\delta(\theta)$, we can apply the same technique to address $\delta(\theta^{(k)})$.

We first provide a bound for the magnitude of the conditional expectation of $\delta(\theta)$, $\|\mathbb{E}_{\tilde{x}_\tau}[\delta(\theta) \mid \tilde{x}_0]\|_2$, in Section 8.4.1. Based on this result, we further draw a non-asymptotic bound for the magnitude of the gradient approximation error, $\|\delta(\theta)\|_2$, in Section 8.4.2. Both results are *verifiable* in each iteration.

For the derivation of the conclusions, we will focus on binary pairwise Markov networks (BPMNs). Let $x \in \{0,1\}^p$ and θ be given, a binary pairwise Markov

network (Höfling and Tibshirani, 2009; Geng et al., 2017) is defined as:

$$P_\theta(\mathbf{x}) = \frac{1}{Z(\theta)} \exp\left(\sum_{i=1}^{p} \sum_{j \geqslant i}^{p} \theta_{ij} x_i x_j\right), \tag{8.10}$$

where $Z(\theta) = \exp(A(\theta))$ is the partition function. θ_{ij} is a component of θ that represents the strength of conditional dependence between X_i and X_j.

8.4.1 An Asymptotic Bound

We first consider the magnitude of the conditional expectation of $\delta(\theta)$ with respect to $\tilde{\mathbf{x}}_\tau$, $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$. To this end, we define $\mathbf{U}$ a $p \times p$ *computable* matrix that is related to θ and the type of MRF in question. U_{ij}, the component in the i^{th} row and the j^{th} column of $\mathbf{U}$, is defined as follows:

$$U_{ij} = \frac{|\exp(-\xi_{ij}) - 1| b^*}{(1 + b^* \exp(-\xi_{ij}))(1 + b^*)}, \tag{8.11}$$

where

$$b^* = \max\left\{r, \min\left\{s, \exp\left(\frac{\xi_{ij}}{2}\right)\right\}\right\},$$

$$s = \exp\left(-\xi_{ii} - \sum_{k \neq i, k \neq j} \xi_{ik} \max\{-\mathrm{sgn}(\xi_{ik}), 0\}\right),$$

$$r = \exp\left(-\theta_{ii} - \sum_{k \neq i, k \neq j} \xi_{i,k} \max\{\mathrm{sgn}(\xi_{i,k}), 0\}\right),$$

and $\mathrm{sgn}(\xi_{ik})$ is the sign function evaluated on $\xi_{ij} = \theta_{\min\{i,j\},\max\{i,j\}}$.

We then define $\mathbf{B}_i$ as a $p \times p$ identity matrix except that its i^{th} row is replaced by the i^{th} row of $\mathbf{U}$, with $i \in \{1, 2, \cdots, p\}$. We further define

$$\mathbf{B} = \mathbf{B}_p \mathbf{B}_{p-1} \mathbf{B}_{p-2} \cdots \mathbf{B}_i \cdots \mathbf{B}_1$$

and the grand sum $\mathcal{G}(\mathbf{B}) = \sum_{i=1}^{p} \sum_{j=1}^{p} B_{ij}$, where B_{ij} is the entry in the i^{th} row

and the j^{th} column of $\mathbf{B}$. With the definitions above, $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$ can be upper bounded by Theorem 8.3.

Theorem 8.3. Let $\tilde{\mathbf{x}}_\tau$ be the sample generated after running Gibbs sampling for τ steps (Gibbs-τ) under the parameterization θ initialized by $\tilde{\mathbf{x}}_0 \in \{0,1\}^p$; then with m denoting the size of sufficient statistics, the following inequality holds:

$$\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2 \leqslant 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau), \tag{8.12}$$

where $\mathbf{B}^\tau$ represents the τ^{th} power of $\mathbf{B}$.

In Theorem 8.3, the bound provided is not only observable in each iteration, but also efficient to compute, offering a convenient method to inspect the quality of the gradient approximation. When the spectral norm of $\mathbf{U}$ is less than 1, the left hand side of (8.12) will converge to 0 (Mitliagkas and Mackey, 2017). Thus, by increasing τ, we can decrease $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$ to an arbitrarily small value.

Theorem 8.3 is derived by bounding the influence of a variable on another variable in $\mathbf{X}$ (i.e., the Dobrushin influence defined in 8.7) with $\mathbf{U}$. Furthermore, $\mathbf{U}$ defined in (8.11) is a sharp bound of the Dobrushin influence whenever $b^* \neq \exp\left(\frac{\xi_{ij}}{2}\right)$, explaining why (8.12) using the definition of $\mathbf{U}$ is tight enough for practical applications.

8.4.2 A Non-Asymptotic Bound

In order to provide a non-asymptotic guarantee for the quality of the gradient approximation, we need to concentrate $\|\delta(\theta)\|_2$ around $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$. Let q defined in Section 8.2.2 be given. Then, q trials of Gibbs sampling are run, resulting in q samples, $\{\tilde{\mathbf{x}}_\tau^{(1)}, \tilde{\mathbf{x}}_\tau^{(2)}, \cdots, \tilde{\mathbf{x}}_\tau^{(q)}\}$. That is to say, for each sufficient statistic, $\psi_j(\theta)$, with $j \in \{1, 2, \cdots, m\}$, we have q samples, $\left\{\psi_j^{(1)}(\theta), \psi_j^{(2)}(\theta), \cdots, \psi_j^{(q)}(\theta)\right\}$. Defining the sample variance of the corresponding sufficient statistics as V_{ψ_j}, we have Theorem 8.4 to provide a non-asymptotic bound for $\|\delta(\theta)\|_2$:

Theorem 8.4. Let θ, q, and an arbitrary $\tilde{\mathbf{x}}_0 \in \{0,1\}^p$ be given. Let m represent the dimension of θ and $\|\delta(\theta)\|_2$ represent the magnitude of the gradient approximation error by running q trials of Gibbs-τ initialized by $\tilde{\mathbf{x}}_0$. Compute $\mathbf{B}$ according to Section 8.4.1 and choose $\epsilon_j > 0$. Then, with probability at least $1 - 2\sum_{j=1}^{m} \beta_j$, where $\beta_j > 0$, $j \in \{1, 2, \cdots, m\}$,

$$\|\delta(\theta)\|_2 \leqslant 2\sqrt{m}\left(\mathcal{G}(\mathbf{B}^\tau) + \sqrt{\frac{\sum_{j=1}^{m} \epsilon_j^2}{4m}}\right), \tag{8.13}$$

with β_j satisfying

$$\epsilon_j = 2\left(\sqrt{\frac{V_{\psi_j} \ln 2/\beta_j}{2q}} + \frac{7 \ln 2/\beta_j}{3(q-1)}\right). \tag{8.14}$$

Notice that the bound in Theorem 8.4 is easily *checkable*, i.e., given τ, q, V_{ψ_j}'s, and θ, we can determine a bound for $\|\delta(\theta)\|_2$ that holds with high probability. Furthermore, Theorem 8.4 provides the sample complexity needed for gradient estimation. Specifically, given small enough β_j's, if we let

$$\mathcal{G}(\mathbf{B}^\tau) = \sqrt{\sum_{j=1}^{m} \epsilon_j^2/4m},$$

we can show that

$$2\sqrt{m}\left(\mathcal{G}(\mathbf{B}^\tau) + \sqrt{\sum_{j=1}^{m} \epsilon_j^2/4m}\right) = O\left(\frac{1}{q}\right).$$

That is to say, by assuming that $\mathcal{G}(\mathbf{B}^\tau)$ and $\sqrt{\sum_{j=1}^{m} \epsilon_j^2/4m}$ share the same scale, the upper bound of the gradient approximation error converges to 0 as q increases. Moreover, we include sample variance, V_{ψ_j}'s, in (8.13). This is because the information provided by sample variance leads to an improved data dependent bound.

8.5 Proof Sketch of Main Results

As mentioned in Section sec:non-asy-bound, the non-asymptotic result in Theorem 8.4 is derived from the asymptotic bound in Theorem 8.3 by concentration inequalities, we therefore only highlight the proof of Theorem 8.3 in this section, and defer other technical results to Section 8.10. Specifically, the proof of Theorem 8.3 is divided into two parts: bounding $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$ by the total variation distance (Section 8.5.1) and bounding the total variation distance (Section 8.5.2).

8.5.1 Bounding $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$ by the Total Variation Distance

To quantify $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$, we first introduce the concept of total variation distance (Levin et al., 2009) that measures the distance between two distributions over $\{0,1\}^p$.

Definition 8.5. Let $u(\mathbf{x})$, and $v(\mathbf{x})$ be two probability distributions of $\mathbf{x} \in \{0,1\}^p$. Then the total variation distance between $u(\mathbf{x})$ and $v(\mathbf{x})$ is given as:

$$\|u(\mathbf{x}) - v(\mathbf{x})\|_{TV} = \frac{1}{2} \sum_{\mathbf{x} \in \{0,1\}^p} |u(\mathbf{x}) - v(\mathbf{x})|.$$

With the definition above, $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$ can be upper bounded by the total variation distance between two distributions ($P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0)$ and $P_\theta(\mathbf{x})$) using the following lemma:

Lemma 8.6. Let $\tilde{\mathbf{x}}_\tau$ be the sample generated after running Gibbs sampling for τ steps (Gibbs-τ) under the parameterization θ initialized by $\tilde{\mathbf{x}}_0 \in \{0,1\}^p$, then the following is true:

$$\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2 \leqslant 2\sqrt{m}\, \|P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0) - P_\theta(\mathbf{x})\|_{TV}\,.$$

With Lemma 8.6, bounding $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$ can be achieved by bounding the total variation distance $\|P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0) - P_\theta(\mathbf{x})\|_{TV}$. Recent advances in the quality control of Gibbs samplers offer us emphverifiable upper bounds for $\|P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0) - P_\theta(\mathbf{x})\|_{TV}$

on the learning of a variety of MRFs (Mitliagkas and Mackey, 2017). However, they can not be applied to BPMNs because of the positive constraint on parameters. We describe these next.

8.5.2 Bounding $\|P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0) - P_\theta(\mathbf{x})\|_{\mathbf{TV}}$

Now we generalize the analysis in Mitliagkas and Mackey (2017) to BPMNs without constraints on the sign of parameters by introducing the definition of the Dobrushin influence matrix and a technical lemma.

Definition 8.7 (Dobrushin influence matrix)**.** The Dobrushin influence matrix of $P_\theta(\mathbf{x})$ is a $p \times p$ matrix $\mathbf{C}$ with its component in the i^{th} row and the j^{th} column, C_{ij}, representing the influence of X_j on X_i given as:

$$C_{ij} = \max_{(\mathbf{X},\mathbf{Y}) \in N_j} \|P_\theta(X_i \mid \mathbf{X}_{-i}) - P_\theta(Y_i \mid \mathbf{Y}_{-i})\|_{TV},$$

where $(\mathbf{X}, \mathbf{Y}) \in N_j$ represents $X_l = Y_l$ for all $l \neq j$.

Lemma 8.8. Let $P_\theta(\mathbf{x})$ represent a binary pairwise Markov network defined in (8.10) that is parameterized by θ. An upper bound of the total influence matrix is given by $\mathbf{U}$ defined in Section 8.4.1.

It should be noticed that, similar to the Theorem 12 in Mitliagkas and Mackey (2017), Lemma 8.8 provides an exact calculation except when $b^* = \exp\left(\frac{\xi_{i,j}}{2}\right)$.

Therefore, we can consider the $\mathbf{U}$ defined in Section 8.4.1 as an upper bound for Dobrushin influence matrix in BPMN and thus apply $\mathbf{U}$ to Theorem 9 in Mitliagkas and Mackey (2017). Then, we have

$$\|P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0) - P_\theta(\mathbf{x})\|_{TV} \leqslant \mathcal{G}(\mathbf{B}^\tau),$$

where $\mathbf{B}^\tau$ represents the τ^{th} power of $\mathbf{B}$. Theorem 8.3 follows this combined with Lemma 8.6

8.6 Application to Structure Learning

With the two bounds introduced in Section 8.4, we can easily examine and control the quality of gradient approximation in each iteration by choosing τ. In detail, we introduce a criterion for the selection of τ in each iteration. Satisfying the proposed criterion, the objective function is guaranteed to decrease asymptotically. That is to say, the difference between $g(\theta^{(k+1)})$ and $g(\hat{\theta})$ is asymptotically *tightened*, compared with the difference between $g\left(\theta^{(k)}\right)$ and $g(\hat{\theta})$. Therefore, we refer to the proposed criterion as TAY-Criterion. Furthermore, using TAY-Criterion we provide an improved SPG method denoted by TAY for short.

Specifically, staring from $\tau = 1$, TAY stops increasing τ when the following is satisfied:

$$2\sqrt{m}\mathcal{G}(\mathbf{B}^{\tau}) < \frac{1}{2}\|\mathbf{G}_{\alpha}(\theta^{(k)})\|_2. \qquad \text{(TAY-Criterion)}$$

We can also derive a non-asymptotic counterpart of TAY-Criterion by combining the results of Theorem 8.1 and Theorem 8.4:

$$\begin{aligned} 0 < 2\sqrt{m}\left(\mathcal{G}(\mathbf{B}^{\tau}) + \sqrt{\frac{\sum_{j=1}^{m}\epsilon_j^2}{4m}}\right) &\leqslant \frac{1}{2}\|\mathbf{G}_{\alpha}(\theta^{(k)})\|_2, \\ \epsilon_j = 2\left(\sqrt{\frac{2V_{\psi_j}\ln 2/\beta_j}{4q}} + \frac{7\ln 2/\beta_j}{3(q-1)}\right), \end{aligned} \tag{8.15}$$

where the V_{ψ_j}'s and β_j's are defined in Theorem 8.4. (8.15) provides the required sample complexity, q, for TAY in each iteration. However, the selection of q according to (8.15) is conservative, because it includes the worst-case scenario where the gradient approximation errors in any two iterations cannot offset each other.

In Section 8.6.1 and 8.6.2, we theoretically analyze the performance guarantees of TAY-Criterion and the convergence of TAY, respectively.

8.6.1 Guarantees of TAY-Criterion

The theorem below provides the performance guarantee for TAY-Criterion in each iteration.

Theorem 8.9. Let $\theta^{(k)}$ and $\tilde{x}_0$ be given. Let q and $\mathbf{B}$ defined in Theorem 8.4 be given. For $\theta^{(k+1)}$ generated in Algorithm 5 using TAY-Criterion, the following is true:

$$\lim_{q\to\infty} P\left(g(\theta^{(k+1)}) < g(\theta^{(k)}) \mid \sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2\right) = 1.$$

Theorem 8.9 makes a statement that the objective function value decreases with large q. Specifically, TAY-Criterion assumes that the upper bound of the conditional expectation of $\|\delta(\theta)\|_2$ is small enough to satisfy the sufficient condition proven in Theorem 8.1. When the number of samples q is large enough, $\|\delta(\theta)\|_2$ itself is very likely to meet the condition and hence the objective function is also likely to decrease with TAY-Criterion satisfied.

8.6.2 Convergence of TAY

Finally, based on Theorem 8.2 and Theorem 8.9, we derive the following theorem on the convergence of TAY.

Theorem 8.10. *Let* $\mathcal{K} = (\theta^{(0)}, \theta^{(1)}, \theta^{(2)}, \cdots, \theta^{(\kappa)})$ *be the iterates generated by TAY. Then, with* $k \in \{1, 2, \cdots, \kappa - 1\}$*, the following is true:*

$$\lim_{q\to\infty} P\left[g(\theta^{(\kappa)}) - g(\hat{\theta}) \leqslant \frac{L}{2\kappa}\left(\|\theta^{(0)} - \hat{\theta}\|_2 + \frac{2}{L}\sum_{k=1}^{\kappa}\|\delta(\theta^{(k)})\|_2\right)^2\right] = 1,$$

where $\hat{\theta}$ *is defined in (8.1).*

8.7 Generalizations

As we demonstrate in Section 8.4 and Section 8.5, the derivation of our main results relies on bounding the Dobrushin influence with $\mathbf{U}$ and we show a procedure to construct $\mathbf{U}$ in the context of BPMNs. Moreover, Mitliagkas and Mackey (2017) and Liu and Domke (2014) provide upper bounds $\mathbf{U}$'s for other types of discrete pairwise MRFs. Therefore, combined with their results, our framework can also be applied to other discrete pairwise Markov networks. Dealing with pairwise MRFs is without any loss of generality, since any discrete MRF can be transformed into a pairwise one (Wainwright et al., 2008; Ravikumar et al., 2010).

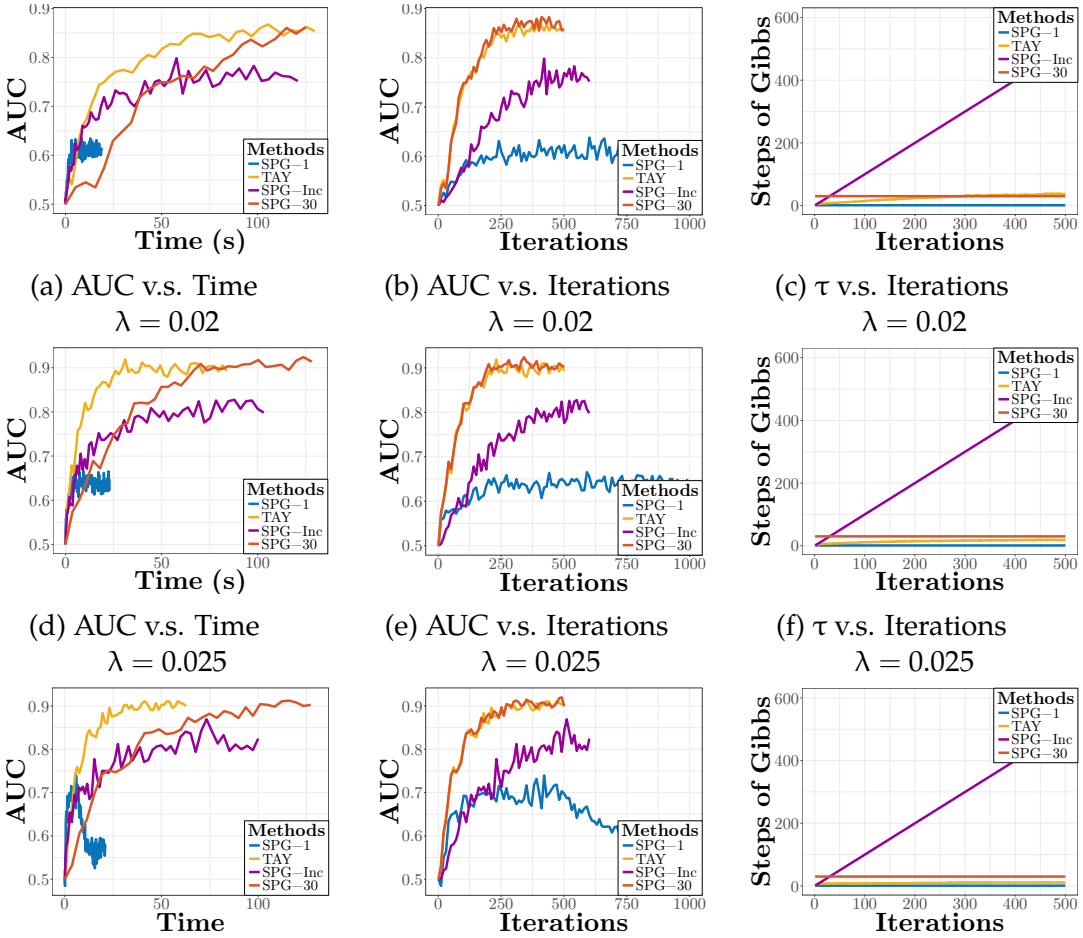
8.8 Experiments

We demonstrate that the structure learning of discrete MRFs benefits substantially from the application of TAY with synthetic data and that the bound provided on the gradient estimation error by Theorem 8.3 is tighter than existing bounds. To illustrate that TAY is readily available for practical problems, we also run TAY using a real world dataset.

8.8.1 Structure Learning

In order to demonstrate the utility of TAY for effectively learning the structures of BPMNs, we simulate two BPMNs (one with 10 nodes and the other one with 20 nodes):

- We set the number of features to $p = 10$ ($p = 20$). Components of θ in the ground truth model are randomly chosen to be nonzero with an edge generation probability of 0.3. The non-zero components of the real parameter have a uniform distribution on $[-2, -1] \bigcup [1, 2]$
- 1000 (2000 for 20 nodes) samples are generated by Gibbs sampling with 1000 burn-in steps.



(a) AUC v.s. Time $\lambda = 0.02$ (b) AUC v.s. Iterations $\lambda = 0.02$ (c) τ v.s. Iterations $\lambda = 0.02$

(d) AUC v.s. Time $\lambda = 0.025$ (e) AUC v.s. Iterations $\lambda = 0.025$ (f) τ v.s. Iterations $\lambda = 0.025$

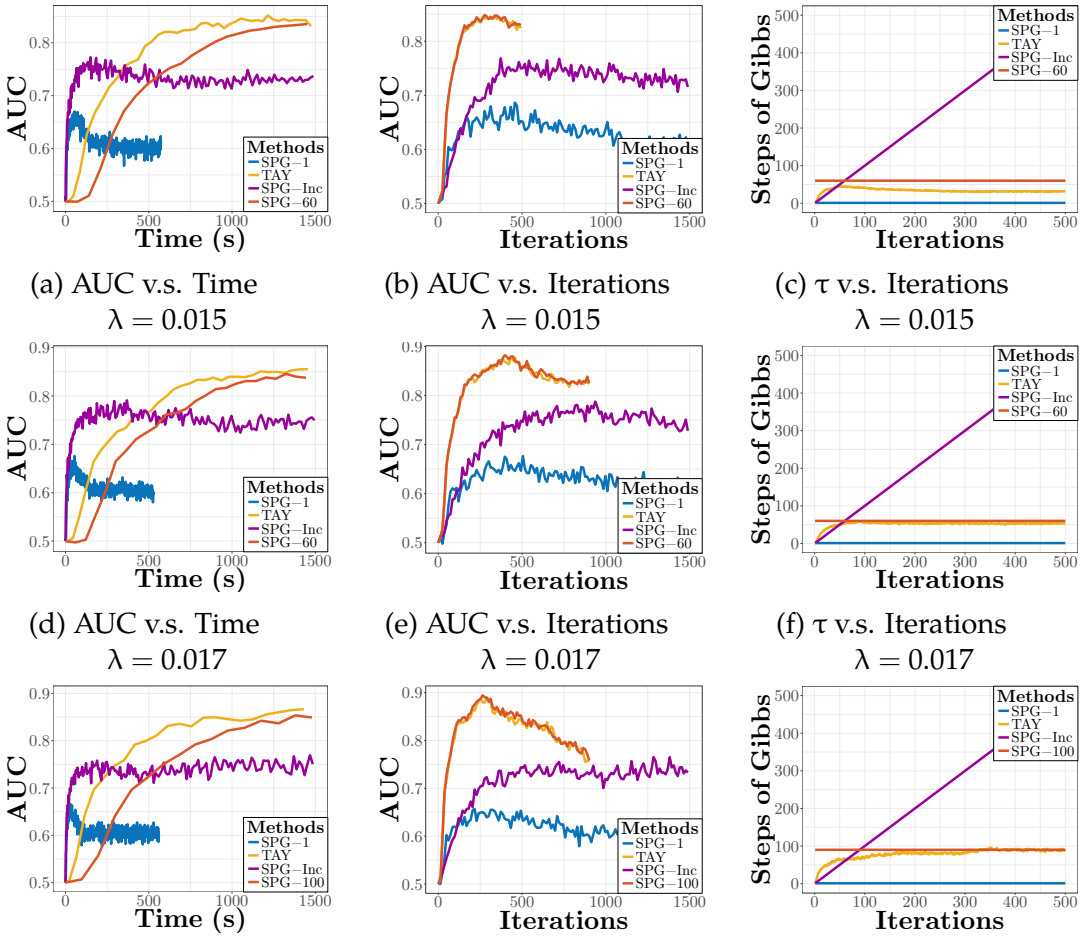
(g) AUC v.s. Time $\lambda = 0.03$ (h) AUC v.s. Iterations $\lambda = 0.03$ (i) τ v.s. Iterations $\lambda = 0.03$

Figure 8.1: Area under curve (AUC) and the steps of Gibbs sampling (τ) for the structure learning of a 10-node network with different λ's.

- The results are averaged over 10 trials.

To be consistent with the literature, the sizes of the BPMNs generated in this chapter are comparable to those in (Honorio, 2012a; Atchade et al., 2014; Miasojedow and Rejchel, 2016).

Then, using the generated samples, we consider SPG and TAY. According to the analysis in Section 8.4, the quality of the gradient approximation is closely



(a) AUC v.s. Time $\lambda = 0.015$ (b) AUC v.s. Iterations $\lambda = 0.015$ (c) τ v.s. Iterations $\lambda = 0.015$

(d) AUC v.s. Time $\lambda = 0.017$ (e) AUC v.s. Iterations $\lambda = 0.017$ (f) τ v.s. Iterations $\lambda = 0.017$

(g) AUC v.s. Time $\lambda = 0.02$ (h) AUC v.s. Iterations $\lambda = 0.02$ (i) τ v.s. Iterations $\lambda = 0.02$

Figure 8.2: Area under curve (AUC) and the steps of Gibbs sampling (τ) for the structure learning of a 20-node network with different λ's.

related to the number of Gibbs sampling steps τ. However, for SPG, there are no convincing schemes for selecting τ. Therefore, we consider a large enough $\tau = 30$ ($\tau = 60$ for 20 nodes) to make sure that the gradient approximation error is small enough. Furthermore, we also evaluate the performance of the algorithm using an increasing τ ($\tau = k$ in the k^{th} iteration), suggested by Atchade et al. (2014) (SPG-Inc).

To strike a fair comparison, we use the same step length $\alpha = 0.4$ and regularization parameter $\lambda \in \{0.02, 0.025, 0.03\}$ ($\lambda \in \{0.015, 0.017, 0.02\}$ for 20 nodes) for different methods. We do not tune the step length individually for each method, since Atchade et al. (2014) has shown that various learning rate selection schemes have minimal impact on the performance in the context of SPG. The number of chains used in Gibbs sampling, q, is not typically a tunable parameter either, since it indicates the allocation of the computational resources. For each method, it can be easily noticed that the larger the number of samples is, the slower but more accurate the method will be. Furthermore, if the q's are different for different methods, it would be difficult to distinguish the effect of τ from that of q. Therefore, we set it to 2000 for 10-node networks and 5000 for 20-node networks. Performances of different methods are compared using the area under curve (AUC) of receiver operating characteristic (ROC) for structure learning in Figure 8.1 and Figure 8.2. The Gibbs sampling steps in each method are also compared.

In Figure 8.1 and Figure 8.2, we plot AUCs against both time and iterations. The two kinds of plots provide different information about the performances of different methods: the former ones focus on overall complexity and the latter illustrate iteration complexity. We run each method until it converges. Using much less time, TAY achieves a similar AUC to SPG with $\tau = 30$ and $\tau = 60$. Moreover, SPG with $\tau = 1$ reaches the lowest AUC, since the quality of the gradient approximation cannot be guaranteed with such a small τ. Therefore, the experimental results indicate that TAY adaptively chooses a τ achieving reasonable accuracy as well as efficiency for structure learning in each iteration.

8.8.2 Tightness of the Proposed Bound

According to the empirical results above, TAY needs a τ only on the order of ten, suggesting that the bound in Theorem 8.3 is tight enough for practical applications. To illustrate this more clearly, we compare (8.12) with another bound on the expectation of the gradient approximation error derived by Fischer (2015). Specifically, we calculate the gradient approximation error, the bound (8.12), and Fischer (2015)'s

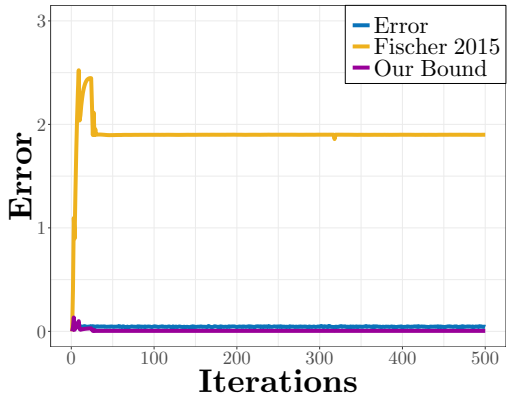


Figure 8.3: The gradient approximation error, the existing bound and the bound (8.12) in the structure learning of a 10-node network.

bound, in each iteration of learning a 10-node network. The results are reported in Figure 8.3. Notice that the bound in Fischer (2015) gets extraordinarily loose with more iterations. Considering this, we may need run Gibbs chains for thousands of steps if we use this bound. In contrast, bound (8.12) is close to and even slightly less than the real error. This is reflective of the fact that the proposed bound is on the expectation instead of the error itself. As a result, (8.12) is much tighter and thus more applicable.

8.8.3 Real World Data

In our final experiment, we run TAY using the Senate voting data from the second session of the 109^{th} Congress (USS). The dataset has 279 samples and 100 variables. Each sample represents the vote cast by each of the 100 senators for a particular bill, where 0 represents nay, and 1 represents yea. Missing data are imputed as 0's. The task of interest is to learn a BPMN model that identifies some clusters that represent the dependency between the voting inclination of each senator and the party with which the senator is affiliated.

We use TAY with $\alpha = 0.4$. 5000 Markov chains are used for Gibbs sampling. Since our task is exploratory analysis, $\lambda = 0.1$ is selected in order to deliver an interpretable result. The proposed algorithm is run for 100 iterations. The resultant

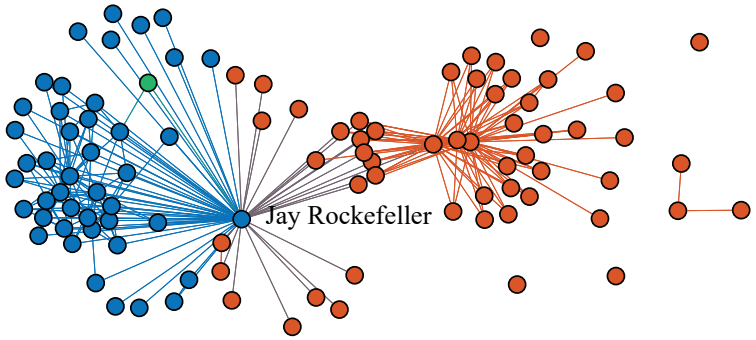


Figure 8.4: The result of TAY on the senator voting data: Red vertices denote Republicans, blue Democracts, and green Independent. Red arcs are among the Republicans and blue Democrats. Purple arcs represent strong ties from certain Republicans to the Democratic cluster. The figure is rendered by `Gephi` (Bastian et al., 2009).

BPMN with only edges corresponding to the positive parameters is shown in Figure 8.4, where each node represents the voting record of a senator and the edges represent some positive dependency between the pair of senators connected. The nodes in red represent Republicans and the nodes in blue represents Democrats. The clustering effects of voting consistency within a party are captured, coinciding with conventional wisdom. More interestingly, Jay Rockefeller, as a Democrat, has many connections with Republicans. This is consistent with the fact that his family has been a "traditionally Republican dynasty" (Wikipedia, 2017).

8.9 Conclusion

We have considered stochastic proximal gradient for L_1-regularized discrete Markov random field estimation. Furthermore, we have conducted a careful analysis of the gradient approximation error of SPG and have provided upper bounds to quantify its magnitude. With the aforementioned analysis, we introduce a learning strategy called *tighten asymptotically* and show that TAY can improve the accuracy and efficiency of SPG in practice.

8.10 Proofs

8.10.1 Proof of Theorem 8.1

We first introduce the following technical lemma.

Lemma 8.11. Let $g(\theta)$, $f(\theta)$, and $h(\theta)$ be defined as in Section 8.2.1; hence $f(\theta)$ is convex and differentiable, and $\boldsymbol{\nabla} f(\theta)$ is Lipschitz continuous with Lipschitz constant L. Let $\alpha \leqslant 1/L$. Let $\mathbf{G}_\alpha(\theta)$ and $\boldsymbol{\Delta} f(\theta)$ be defined as in Section (8.2.2). Then for all θ_1 and θ_2, the following inequality holds:

$$g(\theta_1^\dagger) \leqslant g(\theta_2) + \mathbf{G}_\alpha^\top(\theta_1)(\theta_1 - \theta_2) + (\boldsymbol{\nabla} f(\theta_1) - \boldsymbol{\Delta} f(\theta_1))^\top(\theta_1^\dagger - \theta_2) - \frac{\alpha}{2}\|\mathbf{G}_\alpha(\theta_1)\|_2^2, \tag{8.16}$$

where $\theta_1^\dagger = \theta_1 - \alpha \mathbf{G}_\alpha(\theta_1)$.

Proof. The proof is based on the convergence analysis of the standard proximal gradient method (Vandenberghe, 2016). $f(\theta)$ is a convex differentiable function whose gradient is Lipschitz continuous with Lipschitz constant L. By the quadratic bound of the Lipschitz property:

$$f(\theta_1^\dagger) \leqslant f(\theta_1) - \alpha \boldsymbol{\nabla}^\top f(\theta_1)\mathbf{G}_\alpha(\theta_1) + \frac{\alpha^2 L}{2}\|\mathbf{G}_\alpha(\theta_1)\|_2^2.$$

With $\alpha \leqslant 1/L$, and adding $h(\theta_1^\dagger)$ on both sides of the quadratic bound, we have an upper bound for $g(\theta_1^\dagger)$:

$$g(\theta_1^\dagger) \leqslant f(\theta_1) - \alpha \boldsymbol{\nabla}^\top f(\theta_1)\mathbf{G}_\alpha(\theta_1) + \frac{\alpha}{2}\|\mathbf{G}_\alpha(\theta_1)\|_2^2 + h(\theta_1^\dagger).$$

By convexity of $f(\theta)$ and $h(\theta)$, we have:

$$\begin{aligned} f(\theta_1) &\leqslant f(\theta_2) + \boldsymbol{\nabla}^\top f(\theta_1)(\theta_1 - \theta_2), \\ h(\theta_1^\dagger) &\leqslant h(\theta_2) + (\mathbf{G}_\alpha(\theta_1) - \Delta f(\theta_1))^\top(\theta_1^+ - \theta_2), \end{aligned}$$

which can be used to further upper bound $g(\theta_1^\dagger)$, and results in (8.16). Note that we have used the fact that $G_\alpha(\theta_1) - \Delta f(\theta_1)$ is a subgradient of $h(\theta_1^\dagger)$ in the last inequality. □

With Lemma 8.11, we are now able to prove Theorem 8.1. In Lemma 8.11, let $\theta_1 = \theta_2 = \theta^{(k)}$. Then by (8.8), $\theta_1^\dagger = \theta^{(k+1)}$. The inequality in (8.16) can then be simplified as:

$$g(\theta^{(k+1)}) - g(\theta^{(k)}) \leqslant \alpha\delta(\theta^{(k)})^\top \mathbf{G}_\alpha(\theta^{(k)}) - \frac{\alpha}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2^2.$$

By the Cauchy-Schwarz inequality and the sufficient condition that $\|\delta(\theta^{(k)})\|_2 < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2$, we can further simplify the inequality and conclude $g(\theta^{(k+1)}) < g(\theta^{(k)})$.

8.10.2 Proof of Theorem 8.2

To prove Theorem 8.2, we first review Proposition 1 in Schmidt et al. (2011):

Theorem 8.12 (Convergence on Average, Schmidt et al. (2011))**.** Let

$$\mathcal{K} = (\theta^{(0)}, \theta^{(1)}, \theta^{(2)}, \cdots, \theta^{(\kappa)})$$

be the iterates generated by Algorithm 5, then

$$g\left(\frac{1}{\kappa}\sum_{k=1}^{\kappa}\theta^{(k)}\right) - g(\hat{\theta}) \leqslant \frac{L}{2\kappa}\left(\|\theta^{(0)} - \hat{\theta}\|_2 + \frac{2}{L}\sum_{k=1}^{\kappa}\|\delta(\theta^{(k)})\|_2\right)^2.$$

Furthermore, according to the assumption that $g(\theta^{(k+1)}) \leqslant g(\theta^{(k)})$ with $k \in \{1, 2, \cdots, \kappa\}$, we have: $g\left(\frac{1}{\kappa}\sum_{k=1}^{\kappa}\theta^{(k)}\right) \geqslant g(\theta^{(\kappa)})$. Therefore,

$$g(\theta^{(\kappa)}) - g(\hat{\theta}) \leqslant \frac{L}{2\kappa}\left(\|\theta^{(0)} - \hat{\theta}\|_2 + \frac{2}{L}\sum_{k=1}^{\kappa}\|\delta(\theta^{(k)})\|_2\right)^2.$$

8.10.3 Proof of Theorem 8.3

Proof of Lemma 8.6

The rationale behind our proof follow that of Bengio and Delalleau (2009) and Fischer and Igel (2011).

Let $\tilde{\mathbf{x}}_0 \in \{0,1\}^p$ be an initialization of the Gibbs sampling algorithm. Let θ be the parameterization from which the Gibbs sampling algorithm generates new samples. A Gibbs-τ algorithm hence uses the τ^{th} sample , $\tilde{\mathbf{x}}_\tau$, generated from the chain to approximate the gradient. Since there is only one Markov chain in total, we have $\mathbb{S} = \{\tilde{\mathbf{x}}_\tau\}$. The gradient approximation of Gibbs-τ is hence given by:

$$\Delta f(\theta) = \psi(\tilde{\mathbf{x}}_\tau) - \mathbb{E}_{\mathbb{X}}\psi(\mathbf{x}). \tag{8.17}$$

The actual gradient, $\nabla f(\theta)$, is given in (8.3). Therefore, the difference between the approximation and the actual gradient is

$$\delta(\theta) = \Delta f(\theta) - \nabla f(\theta) = \psi(\tilde{\mathbf{x}}_\tau) - \mathbb{E}_\theta \psi(\mathbf{x}) = \nabla \log P_\theta(\tilde{\mathbf{x}}_\tau).$$

We rewrite

$$P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0) = P(\tilde{\mathbf{X}}_\tau = \mathbf{x} \mid \tilde{\mathbf{x}}_0) = P_\theta(\mathbf{x}) + \epsilon_\tau(\mathbf{x}),$$

where $\epsilon_\tau(\mathbf{x})$ is the difference between $P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0)$ and $P_\theta(\mathbf{x})$. Consider the expectation of the j^{th} component of $\delta(\theta)$, $\delta_j(\theta)$, where $j \in \{1, 2, \cdots, m\}$, after running Gibbs-$\tau$ that is initialized by $\tilde{\mathbf{x}}_0$:

$$\begin{aligned}
\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\theta) \mid \tilde{\mathbf{x}}_0] &= \sum_{\mathbf{x} \in \{0,1\}^p} P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0)\delta_j(\theta) = \sum_{\mathbf{x} \in \{0,1\}^p} (P_\theta(\mathbf{x}) + \epsilon_\tau(\mathbf{x}))\delta_j(\theta) \\
&= \sum_{\mathbf{x} \in \{0,1\}^p} \epsilon_\tau(\mathbf{x})\delta_i(\theta) = \sum_{\mathbf{x} \in \{0,1\}^p} (P_\tau(\mathbf{x} \mid \mathbf{x}_0) - P_\theta(\mathbf{x}))\delta_j(\theta) \\
&= \sum_{\mathbf{x} \in \{0,1\}^p} (P_\tau(\mathbf{x} \mid \mathbf{x}_0) - P_\theta(\mathbf{x}))\nabla_j \log P_\theta(\tilde{\mathbf{x}}_\tau),
\end{aligned} \tag{8.18}$$

where we have used the fact that $\sum_{\mathbf{x} \in \{0,1\}^p} P_\theta(\mathbf{x})\nabla_j \log P_\theta(\mathbf{x}) = 0$, and $\nabla_j \log P_\theta(\mathbf{x})$

represents the j^{th} component of $\boldsymbol{\nabla} \log P_\theta(\tilde{\mathbf{x}}_\tau)$, with $j \in \{1, 2, \cdots, m\}$.

Therefore, from (8.18),

$$\begin{aligned} |\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\theta) \mid \tilde{\mathbf{x}}_0]| &\leqslant \sum_{\mathbf{x} \in \{0,1\}^p} |P_\tau(\mathbf{x} \mid \mathbf{x}_0) - P_\theta(\mathbf{x})| \cdot |\nabla_j \log P_\theta(\tilde{\mathbf{x}}_\tau)| \\ &\leqslant \sum_{\mathbf{x} \in \{0,1\}^p} |P_\tau(\mathbf{x} \mid \mathbf{x}_0) - P_\theta(\mathbf{x})| = 2 \left\| P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0) - P_\theta(\mathbf{x}) \right\|_{TV}, \end{aligned} \tag{8.19}$$

where we have used the fact that $|\nabla_j \log P_\theta(\tilde{\mathbf{x}}_\tau)| \leqslant 1$ when $\boldsymbol{\psi}(\mathbf{x}) \in \{0,1\}^m$, for all $\mathbf{x} \in \{0,1\}^p$.

Therefore, by (8.19),

$$\begin{aligned} \|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\boldsymbol{\delta}(\theta) \mid \tilde{\mathbf{x}}_0]\|_2 =& \sqrt{\sum_{j=1}^{m} |\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\theta) \mid \tilde{\mathbf{x}}_0]|^2} \leqslant \sqrt{m \times (2 \left\| P_\tau(\mathbf{x} \mid \mathbf{x}_0) - P_\theta(\mathbf{x}) \right\|_{TV})^2} \\ =& 2\sqrt{m} \left\| P_\tau(\mathbf{x} \mid \mathbf{x}_0) - P_\theta(\mathbf{x}) \right\|_{TV}. \end{aligned}$$

Proof of Lemma 8.8

Let $j \neq i$ be given. With $\xi_{ij} = \theta_{\min\{i,j\},\max\{i,j\}}$, consider

$$\begin{aligned} P_\theta(X_i = 1 \mid \mathbf{X}_{-i}) =& \frac{P_\theta(X_i = 1, \mathbf{X}_{-i})}{P_\theta(X_i = 0, \mathbf{X}_{-i}) + P_\theta(X_i = 1, \mathbf{X}_{-i})} \\ =& \frac{1}{1 + \exp\left(-\theta_{ii} - \sum_{k \neq i} \xi_{i,k} X_k\right)} \\ =& \frac{1}{1 + \exp\left(-\theta_{ii} - \sum_{k \neq i, k \neq j} \xi_{i,k} X_k\right) \exp\left(-\xi_{i,j} X_j\right)} \\ =& g\left(\exp\left(-\xi_{i,j} X_j\right), b_1\right), \end{aligned}$$

where

$$b = \exp\left(-\theta_{ii} - \sum_{k \neq i, k \neq j} \xi_{i,k} X_k\right) \in [r, s],$$

with

$$r = \exp\left(-\theta_{ii} - \sum_{k\neq i, k\neq j} \xi_{i,k} \max\{\operatorname{sgn}(\xi_{i,k}), 0\}\right),$$
$$s = \exp\left(-\theta_{ii} - \sum_{k\neq i, k\neq j} \xi_{i,k} \max\{-\operatorname{sgn}(\xi_{i,k}), 0\}\right).$$

Therefore,

$$\begin{aligned}
C_{ij} &= \max_{\mathbf{X},\mathbf{Y}\in N_j} \frac{1}{2}|P_\theta(X_i = 1 \mid \mathbf{X}_{-i}) - P_\theta(Y_i = 1 \mid \mathbf{Y}_{-i})| \\
&\quad + \frac{1}{2}|P_\theta(X_i = 0 \mid \mathbf{X}_{-i}) - P_\theta(Y_i = 0 \mid \mathbf{Y}_{-i})| \\
&= \max_{\mathbf{X},\mathbf{Y}\in N_j} |P_\theta(X_i = 1 \mid \mathbf{X}_{-i}) - P_\theta(Y_i = 1 \mid \mathbf{Y}_{-i})| \\
&= \max_{\mathbf{X},\mathbf{Y}\in N_j} |g\left(\exp\left(-\xi_{i,j}X_j\right), b\right) - g\left(\exp\left(-\xi_{i,j}Y_j\right), b\right)| \\
&= \max_{\mathbf{X},\mathbf{Y}\in N_j} \frac{|\exp\left(-\xi_{i,j}X_j\right) - \exp\left(-\xi_{i,j}Y_j\right)|b}{\left(1 + b\exp\left(-\xi_{i,j}X_j\right)\right)\left(1 + b_1\exp\left(-\xi_{i,j}Y_j\right)\right)} \\
&= \max_{\mathbf{X},\mathbf{Y}\in N_j} \frac{|\exp\left(-\xi_{i,j}\right) - 1|b}{\left(1 + b\exp\left(-\xi_{i,j}\right)\right)(1 + b)}.
\end{aligned}$$

Then following the Lemma 15 in Mitliagkas and Mackey (2017), we have

$$C_{ij} \leqslant \frac{|\exp\left(-\xi_{i,j}\right) - 1|b^*}{\left(1 + b_1^*\exp\left(-\xi_{i,j}\right)\right)(1 + b^*)}, \tag{8.20}$$

with $b^* = \max\left\{r, \min\left\{s, \exp\left(\frac{\xi_{i,j}}{2}\right)\right\}\right\}$.

8.10.4 Proof of Theorem 8.4

We are interested in concentrating $\|\delta(\theta)\|_2$ around $\|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2$. To this end, we first consider concentrating $\delta_j(\theta)$ around $\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\theta) \mid \tilde{\mathbf{x}}_0]$, where $j \in \{1, 2, \cdots, m\}$. Let q defined in Algorithm 4 be given. Then q trials of Gibbs sampling are run, resulting in $\left\{\delta_j^{(1)}(\theta), \delta_j^{(2)}(\theta), \cdots, \delta_j^{(q)}(\theta)\right\}$, and $\left\{\psi_j^{(1)}(\theta), \psi_j^{(2)}(\theta), \cdots, \psi_j^{(q)}(\theta)\right\}$ defined in

Section 8.4.2, one element for each of the q trials. Since all the trials are independent, $\delta_j^{(i)}(\theta)$'s can be considered as i.i.d. samples with mean $\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\theta) \mid \tilde{\mathbf{x}}_0]$. Furthermore, $\delta_j^{(i)}(\theta) = \nabla_j \log P_\theta(\tilde{\mathbf{x}}_\tau) \in [-1, 1]$ when $\psi(\mathbf{x}) \in \{0,1\}^m$, for all $\mathbf{x} \in \{0,1\}^p$. Let $\beta_j > 0$ be given; we define the adversarial event:

$$E_j^q(\epsilon_j) = \left| \frac{1}{q} \sum_{i=1}^{q} \delta_j^{(i)}(\theta) - \mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\theta) \mid \tilde{\mathbf{x}}_0] \right| > \epsilon_j, \tag{8.21}$$

with $j \in \{1, 2, \cdots, m\}$.

Define another random variable $Z_j = \frac{1+\delta_j(\theta)}{2}$ with samples $Z_j^{(i)} = \frac{1+\delta_j^{(i)}(\theta)}{2}$ and the sample variance $V_{Z_j} = \frac{V_{\delta_j}}{4} = \frac{V_{\psi_j}}{4}$.

Considering $Z \in [0, 1]$, we can apply Theorem 4 in Maurer and Pontil (2009) and achieve

$$P\left(\left| \frac{1}{q} \sum_{i=1}^{q} Z_j^{(i)} - \mathbb{E}_{\tilde{\mathbf{x}}_\tau}[Z_j \mid \tilde{\mathbf{x}}_0] \right| > \frac{\epsilon_j}{2} \right) \leqslant 2\beta_j,$$

where

$$\frac{\epsilon_j}{2} = \sqrt{\frac{2V_{Z_j} \ln 2/\beta_j}{q}} + \frac{7 \ln 2/\beta_j}{3(p-1)} = \sqrt{\frac{V_{\psi_j} \ln 2/\beta_j}{2q}} + \frac{7 \ln 2/\beta_j}{3(p-1)}.$$

That is to say

$$P\left(E_j^q(\epsilon_j)\right) \leqslant 2\beta_j.$$

Now, for all $j \in \{1, 2, \cdots, m\}$, we would like $\frac{1}{m} \sum_{i=1}^{m} \delta_j^{(i)}(\theta)$ to be close to $\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\theta) \mid \tilde{\mathbf{x}}_0]$. i.e.,

$$\left| \frac{1}{q} \sum_{i=1}^{q} \delta_j^{(i)}(\theta) - \mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\theta) \mid \tilde{\mathbf{x}}_0] \right| \leqslant \epsilon_j.$$

This concentrated event will occur with probability:

$$1 - P\left(E_j(\epsilon_j)\right) \geqslant 1 - P\left(E_j^q(\epsilon_j)\right) \geqslant 1 - 2\beta_j.$$

When all the concentrated events occur for each j,

$$\begin{aligned}
&\|\boldsymbol{\delta}(\boldsymbol{\theta})\|_2 - \|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\boldsymbol{\delta}(\boldsymbol{\theta}) \mid \tilde{\mathbf{x}}_0]\|_2 \\
\leqslant &\|\boldsymbol{\delta}(\boldsymbol{\theta}) - \mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\boldsymbol{\delta}(\boldsymbol{\theta}) \mid \tilde{\mathbf{x}}_0]\|_2 = \left\| \frac{1}{q} \sum_{i=1}^{q} \boldsymbol{\delta}^{(i)}(\boldsymbol{\theta}) - \mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\boldsymbol{\delta}(\boldsymbol{\theta}) \mid \tilde{\mathbf{x}}_0] \right\|_2 \\
= &\sqrt{\sum_{j=1}^{m} \left(\frac{1}{q} \sum_{i=1}^{q} \delta_j^{(i)}(\boldsymbol{\theta}) - \mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta_j(\boldsymbol{\theta}) \mid \tilde{\mathbf{x}}_0] \right)^2} \leqslant \sqrt{\sum_{j=1}^{m} \epsilon_j^2}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\|\boldsymbol{\delta}(\boldsymbol{\theta})\|_2 &\leqslant \|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\boldsymbol{\delta}(\boldsymbol{\theta}) \mid \tilde{\mathbf{x}}_0]\|_2 + \sqrt{\sum_{j=1}^{m} \epsilon_j^2} \leqslant 2\sqrt{m}\, \|P_\tau(\mathbf{x} \mid \tilde{\mathbf{x}}_0) - P_{\boldsymbol{\theta}}(\mathbf{x})\|_{TV} + \sqrt{\sum_{j=1}^{m} \epsilon_j^2} \\
&\leqslant 2\sqrt{m} \left(\mathcal{G}(\mathbf{B}^\tau) + \sqrt{\frac{\sum_{j=1}^{m} \epsilon_j^2}{4m}} \right).
\end{aligned}$$

That is to say, we can conclude that (8.13) holds provided that all the concentrated events occur. Thus, the probability that (8.13) holds follows the inequality below:

$$\begin{aligned}
P\left(\|\boldsymbol{\delta}(\boldsymbol{\theta})\|_2 \leqslant 2\sqrt{m} \left(\mathcal{G}(\mathbf{B}^\tau) + \sqrt{\frac{\sum_{j=1}^{m} \epsilon_j^2}{4m}} \right) \right) &\geqslant 1 - P\left(\bigcup_{j=1}^{m} E_j(\epsilon_j) \right) \\
&\geqslant 1 - \sum_{j=1}^{m} P\left(E_j^q(\epsilon_j) \right) \geqslant 1 - 2\sum_{j=1}^{m} \beta_j.
\end{aligned}$$

8.10.5 Proof of Theorem 8.9

We consider the probability that the achieved objective function value decreases in the k^{th} iteration provided that the criterion TAY-Criterion is satisfied:

$$P\left(g(\boldsymbol{\theta}^{(k+1)}) < g(\boldsymbol{\theta}^{(k)}) \,|\, 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\boldsymbol{\theta}^{(k)})\|_2 \right).$$

Since $\|\delta(\theta^{(k)})\|_2 \leqslant \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2$ provided in Theorem 8.1 is a sufficient condition for $g(\theta^{(k+1)}) \leqslant g(\theta^{(k)})$, we have:

$$
\begin{aligned}
&P\left(g(\theta^{(k+1)}) \leqslant g(\theta^{(k)}) \mid 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2\right) \\
\geqslant & P\left(\|\delta(\theta^{(k)})\|_2 \leqslant \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2 \mid 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2\right) \\
= & 1 - P\left(\|\delta(\theta^{(k)})\|_2 > \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2 \mid 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2\right) \\
\geqslant & 1 - P\Big(\|\delta(\theta^{(k)})\|_2 - \|\mathbb{E}_{\tilde{\mathbf{x}}_\tau}[\delta(\theta) \mid \tilde{\mathbf{x}}_0]\|_2 > \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2 - 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) \mid \\
& 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2\Big) \\
\geqslant & 1 - \sum_{j=1}^{m} P\left(\mathbb{E}_j^q(\frac{1}{2\sqrt{m}}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2 - 2\mathcal{G}(\mathbf{B}^\tau)) \mid 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2\right),
\end{aligned}
$$

where $\mathbb{E}_j^q(\frac{1}{2\sqrt{m}}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2 - 2\mathcal{G}(\mathbf{B}^\tau)$ is defined in (8.21) and in the penultimate inequality we apply (8.12). As q approaches infinity, by the weak law of large numbers, we have

$$
\lim_{q\to\infty} P\left(\mathbb{E}_j^q(\frac{1}{2\sqrt{m}}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2 - 2\mathcal{G}(\mathbf{B}^\tau))\right) = 0.
$$

Then,

$$
\begin{aligned}
&\lim_{q\to\infty} P\left(g(\theta^{(k+1)}) < g(\theta^{(k)}) \mid 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2\right) \\
\geqslant & 1 - \lim_{q\to\infty}\sum_{j=1}^{m} P\left(\mathbb{E}_j^q(\frac{1}{2\sqrt{m}}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2 - 2\mathcal{G}(\mathbf{B}^\tau)) \mid 2\sqrt{m}\mathcal{G}(\mathbf{B}^\tau) < \frac{1}{2}\|\mathbf{G}_\alpha(\theta^{(k)})\|_2\right) \\
= & 1.
\end{aligned}
$$

8.10.6 Proof of Theorem 8.10

According to Theorem 8.2, we only need to show

$$\lim_{q\to\infty} P\left(g(\theta^{(k+1)}) \leqslant g(\theta^{(k)})\right) = 1,$$

for $k = 1, 2, \cdots, \kappa - 1$.

By a union bound, the following inequality is true:

$$\lim_{q\to\infty} P\left(g(\theta^{(k+1)}) \leqslant g(\theta^{(k)})\right) \leqslant 1 - \sum_{k=1}^{\kappa-1} \lim_{q\to\infty} P\left(g(\theta^{(k+1)}) > g(\theta^{(k)})\right).$$

Notice that, following TAY, we always have:

$$P\left(2\sqrt{m}\mathcal{G}(\mathbf{B}^{\tau}) < \frac{1}{2}\|\mathbf{G}_{\alpha}(\theta^{(k)})\|_2\right) = 1,$$

suggesting

$$\begin{aligned}
&\lim_{q\to\infty} P\left(g(\theta^{(k+1)}) > g(\theta^{(k)})\right) \\
&= \lim_{q\to\infty} P\left(g(\theta^{(k+1)}) > g(\theta^{(k)}) \mid 2\sqrt{m}\mathcal{G}(\mathbf{B}^{\tau}) < \frac{1}{2}\|\mathbf{G}_{\alpha}(\theta^{(k)})\|_2\right) = 0,
\end{aligned}$$

where the equality is due to Theorem 8.9.

Finally, with Theorem 8.2, we can finish the proof.

9 A SCREENING RULE FOR ℓ_1-REGULARIZED ISING MODEL ESTIMATION

9.1 Introduction

While the field of statistical learning with sparsity (Hastie et al., 2015) has been steadily rising to prominence ever since the introduction of the lasso (least absolute shrinkage and selection operator) at the end of the last century (Tibshirani, 1996), it was not until the recent decade that various *screening rules* debuted to further equip the ever-evolving optimization arsenals for some of the most fundamental problems in sparse learning such as ℓ_1-regularized generalized linear models (GLMs, Friedman et al. 2010) and inverse covariance matrix estimation (Friedman et al., 2008). Screening rules, usually in the form of an analytic formula or an optimization procedure that is extremely fast to solve, can accelerate learning drastically by leveraging the inherent sparsity of many high-dimensional problems. Generally speaking, screening rules can identify a significant portion of the zero components of an optimal solution beforehand at the cost of minimal computational overhead, and hence substantially reduce the dimension of the parameterization, which makes possible efficient computation for large-scale sparse learning problems.

Pioneered by Ghaoui et al. 2010, various screening rules have emerged to speed up learning for generative models (e.g. Gaussian graphical models) as well as for discriminative models (e.g. GLMs), and for continuous variables (e.g. lasso) as well as for discrete variables (e.g. logistic regression, support vector machines). Table 9.1 summarizes some of the iconic work in the literature, where, to the best of our knowledge, screening rules for generative models with discrete variables are still notably absent.

Contrasted with this notable absence is the ever stronger craving in the big data era for scaling up the learning of generative models with discrete variables, especially in a blockwise structure identification setting. For example, in gene mutation analysis (Wan et al., 2015, 2016), among tens of thousands of sparse binary variables

Table 9.1: Screening rules in the literature at a glance.

	Discriminative Models	Generative Models
Continuous Variables	Ghaoui et al. 2010 Tibshirani et al. 2012 Liu et al. 2013b Wang et al. 2013 Fercoq et al. 2015 Xiang et al. 2016 Lee et al. 2017	Banerjee et al. 2008 Honorio and Samaras 2010 Witten et al. 2011 Mazumder and Hastie 2012 Danaher et al. 2014 Luo et al. 2014 Yang et al. 2015b
Discrete Variables	Ghaoui et al. 2010 Tibshirani et al. 2012 Wang et al. 2014 Ndiaye et al. 2015	?

representing mutations of genes, we are interested in identifying a handful of mutated genes that are connected into various blocks and exert synergistic effects on the cancer. While a sparse Ising model is a desirable choice, for such an application the scalability of the model could fail due to the innate $\mathcal{NP}$-hardness (Karger and Srebro, 2001) of inference, and hence maximum likelihood learning, owing to the partition function. To date, even with modern approximation techniques, a typical application with sparse discrete graphical models usually involves only hundreds of variables (Viallon et al., 2014; Barber et al., 2015; Vuffray et al., 2016).

Between the need for the scalability of high-dimensional Ising models and the absence of screening rules that are deemed crucial to accelerated and scalable learning, we have a technical gap to bridge: *can we identify screening rules that can speed up the learning of ℓ_1-regularized Ising models?* The major contribution of this chapter is to give an affirmative answer to this question. Specifically, we show the following.

- The screening rule is a simple closed-form formula that is a necessary and sufficient condition for exact blockwise structure recovery of the solution with a given regularization parameter. Upon the identification of blockwise structures, different blocks of variables can be considered as different Ising

models and can be solved separately. The various blocks can even be solved *in parallel* to attain further efficiency. Empirical results on both simulated and real-world datasets demonstrate the tremendous efficiency, scalability, and insights gained from the introduction of the screening rule. Efficient learning of ℓ_1-regularized Ising models from thousands of variables on a single machine is hence readily attainable.

- As an initial attempt to fill in the vacancy illustrated in Table 9.1, our work is instructive to further exploration of screening rules for other graphical models with discrete random variables, and to combining screening rules with various optimization methods to facilitate better learning. Furthermore, compared with its Gaussian counterpart, where screening rules are available (Table 9.1) and learning is scalable (Hsieh et al., 2013), the proposed screening rule is especially valuable and desperately needed to address the more challenging learning problem of sparse Ising models.

9.2 Notation and Background

9.2.1 Ising Models

Let $X = \left[X_1, X_2, \cdots, X_p\right]^\top$ be a $p \times 1$ binary random vector, with $X_i \in \{-1, 1\}$, and $i \in \{1, 2, \cdots, p\} \triangleq V$. Let there be a dataset $\mathbb{X}$ with n independent and identically distributed samples of X, denoted as $\mathbb{X} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$. Here, $x^{(k)}$ is a $p \times 1$ vector of assignments that realizes X, where $k \in \{1, 2, \cdots, n\}$. We further use $x_i^{(k)}$ to denote the i^{th} component of the k^{th} sample in the dataset. Let $\theta \in \Theta$ be a $p \times p$ *symmetric* matrix whose diagonal entries are zeros. An Ising model (Wan et al., 2016) with the parameterization θ is:

$$P_\theta(x) = \frac{1}{Z(\theta)} \exp\left(\sum_{i=1}^{p-1} \sum_{j>i}^{p} \theta_{ij} x_i x_j\right), \tag{9.1}$$

where θ_{ij} represents the component of θ at the i^{th} row and the j^{th} column, and x_i and x_j represent the i^{th} and the j^{th} components of x, respectively. $Z(\theta)$ is a normalization constant, partition function, that ensures the probabilities sum up to one. The partition function is given as $Z(\theta) = \sum_{x\in\{-1,1\}^p} \exp\left(\sum_{i=1}^{p-1}\sum_{j>i}^{p} \theta_{ij}x_ix_j\right)$. Note that for ease of presentation, we consider Ising models with only pairwise interaction/potential here. Generalization to Ising models with unary potentials is given in Section 9.5.

9.2.2 Graphical Interpretation

With the notion of the probability given by an Ising model in (9.1), estimating an ℓ_1-regularized Ising model is defined as finding $\hat{\theta}$, the penalized maximum likelihood estimator (MLE) under the lasso penalty:

$$\begin{aligned}\hat{\theta} &= \arg\max_{\theta} \frac{1}{n}\sum_{k=1}^{n} \log P_\theta\left(x^{(k)}\right) - \frac{\lambda}{2}\|\theta\|_1 \\ &= \arg\min_{\theta} -\frac{1}{n}\sum_{k=1}^{n}\sum_{i=1}^{p-1}\sum_{j>i}^{p} \theta_{ij}x_i^{(k)}x_j^{(k)} + A(\theta) + \frac{\lambda}{2}\|\theta\|_1.\end{aligned} \tag{9.2}$$

Here, $A(\theta) = \log Z(\theta)$ is the log-partition function; $\|\theta\|_1 = \sum_{i=1}^{p}\sum_{j=1}^{p}|\theta_{ij}|$ is the lasso penalty that encourages a sparse parameterization. $\lambda \geqslant 0$ is a given regularization parameter. Using $\frac{\lambda}{2}$ is suggestive of the symmetry of θ so that $\frac{\lambda}{2}\|\theta\|_1 = \lambda\sum_{i=1}^{p-1}\sum_{j>i}^{p}|\theta_{ij}|$, which echoes the summations in the negative log-likelihood function. Note that θ corresponds to the adjacency matrix constructed by the p components of X as nodes, and $\theta_{ij} \neq 0$ indicates that there is an edge between X_i and X_j. We further denote a *partition* of V into L blocks as $\{C_1, C_2, \cdots, C_L\}$, where $C_l, C_{l'} \subseteq V$, $C_l \cap C_{l'} = \emptyset$, $\bigcup_{l=1}^{L} C_l = V$, $l \neq l'$, and for all $l, l' \in \{1, 2, \cdots, L\}$. Without loss of generality, we assume that the nodes in different blocks are ordered such that if $i \in C_l$, $j \in C_{l'}$, and $l < l'$, then $i < j$.

9.2.3 Blockwise Solutions

We introduce the definition of a blockwise parameterization:

Definition 9.1. We call θ *blockwise* with respect to the partition $\{C_1, C_2, \cdots, C_L\}$ if $\forall l$ and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and $\forall i \in C_l$, $\forall j \in C_{l'}$, we have $\theta_{ij} = 0$.

When θ is blockwise, we can represent θ in a block diagonal fashion:

$$\theta = \operatorname{diag}(\theta_1, \theta_2, \cdots, \theta_L), \tag{9.3}$$

where θ_1, θ_2, $\cdots$, and θ_L are symmetric matrices that correspond to C_1, C_2, $\cdots$, and C_L, respectively. Note that if we can identify the blockwise structure of $\hat{\theta}$ in advance, we can solve each block independently (See 9.8.1). Since the size of each block could be much smaller than the size of the original problem, each block could be much easier to learn compared with the original problem. Therefore, efficient identification of blockwise structure could lead to substantial speedup in learning.

9.3 The Screening Rule

9.3.1 Main Results

The preparation in Section 9.2 leads to the discovery of the following strikingly simple screening rule presented in Theorem 9.2.

Theorem 9.2. Let a partition of V, $\{C_1, C_2, \cdots, C_L\}$, be given. Let the dataset $\mathbb{X} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$ be given. Define $\mathbb{E}_{\mathbb{X}} X_i X_j = \frac{1}{n} \sum_{k=1}^{n} x_i^{(k)} x_j^{(k)}$. A necessary and sufficient condition for $\hat{\theta}$ to be blockwise with respect to the given partition is that

$$|\mathbb{E}_{\mathbb{X}} X_i X_j| \leqslant \lambda, \tag{9.4}$$

for all l and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and for all $i \in C_l$, $j \in C_{l'}$.

In terms of exact blockwise structure identification, Theorem 9.2 provides a foolproof (necessary and sufficient) and yet easily checkable result by comparing the

Algorithm 6 Blockwise Minimization

1: **Input:** dataset $\mathbb{X}$, regularization parameter λ.
2: **Output:** $\hat{\theta}$.
3: $\forall i, j \in V$ such that $j > i$, compute the second empirical moments $\mathbb{E}_{\mathbb{X}} X_i X_j$'s .
4: Identify the partition $\{C_1, C_2, \cdots, C_L\}$ using the second empirical moments from the previous step and according to Witten et al. (2011); Mazumder and Hastie (2012).
5: $\forall l \in L$, perform blockwise optimization over C_l for $\hat{\theta}_l$.
6: Ensemble $\hat{\theta}_l$'s according to (9.3) for $\hat{\theta}$.
7: **Return** $\hat{\theta}$.

absolute second empirical moments $|\mathbb{E}_{\mathbb{X}} X_i X_j|$'s with the regularization parameter λ. We also notice the remarkable similarity between the proposed screening rule and the screening rule for Gaussian graphical model blockwise structure identification in Witten et al. 2011; Mazumder and Hastie 2012. In the Gaussian case, the screening rule can be attained by simply replacing the second empirical moment matrix in (9.4) with the sample covariance matrix. While the exact solution in the Gaussian case can be computed in polynomial time, estimating an Ising model via maximum likelihood in general is $\mathcal{NP}$-hard . However, as a consequence of applying the screening rule, the blockwise structure of an ℓ_1-regularized Ising model can be determined *as easily as* the blockwise structure of a Gaussian graphical model, despite the fact that within each block, exact learning of a sparse Ising model could still be challenging.

Furthermore, the screening rule also provides us a principal approach to leverage sparsity for the gain of efficiency: by increasing λ, the nodes of the Ising model will be shattered into smaller and smaller blocks, according to the screening rule. Solving many Ising models with small blocks of variables is amenable to both estimation algorithm and parallelism.

9.3.2 Regularization Parameters

The screening rule also leads to a significant implication to the range of regularization parameters in which $\hat{\theta} \neq 0$. Specifically, we have the following theorem.

Theorem 9.3. Let the dataset $\mathbb{X} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$ be given, and let $\lambda = \lambda_{\max}$ represent the smallest regularization parameter such that $\hat{\theta} = 0$ in (9.2). Then $\lambda_{\max} = \max_{i,j \in V, i \neq j} |\mathbb{E}_{\mathbb{X}} X_i X_j| \leqslant 1$.

With $\lambda_{\max}$, one can decide the range of regularization parameters, $[0, \lambda_{\max}]$, that generates graphs with nonempty edge sets, which is an important first step for pathwise optimization algorithms (a.k.a. homotopy algorithms) that learn the solutions to the problem under a range of λ's. Furthermore, the fact that $\lambda_{\max} \leqslant 1$ for any given dataset $\mathbb{X}$ suggests that comparison across different networks generated by different datasets is comprehensible. Finally, in Section 9.4, $\lambda_{\max}$ will also help to establish the connection between the screening rule for exact learning and some of the popular inexact (alternative) learning algorithms in the literature.

9.3.3 Fully Disconnected Nodes

Another consequence of the screening rule is the necessary and sufficient condition that determines the regularization parameter with which a node is fully disconnected from the remaining nodes:

Corollary 9.4. Let the dataset $\mathbb{X} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$ be given. X_i is fully disconnected from the remaining nodes in $\hat{\theta}$, where $i \in V$ (i.e., $\hat{\theta}_{ij} = \hat{\theta}_{ji} = 0, \forall j \in V \setminus \{i\}$), if and only if $\lambda \geqslant \max_{j \in V \setminus \{i\}} |\mathbb{E}_{\mathbb{X}} X_i X_j|$.

In high-dimensional exploratory data analysis, it is usually the case that *most* of the variables are fully disconnected (Danaher et al., 2014; Wan et al., 2016). In this scenario, Corollary 9.4 provides a regularization parameter threshold with which we can identify *exactly* the subset of fully disconnected nodes. Since we can choose a threshold large enough to make *any* nodes fully disconnected, we can discard a significant portion of the variables efficiently and flexibly at will with exact optimization guarantees due to Corollary 9.4. By discarding the large portion of fully disconnected variables, the learning algorithm can focus on only a moderate number of connected variables, which potentially results in a substantial efficiency gain.

9.3.4 Blockwise Minimization

We conclude this section by providing the blockwise minimization algorithm in Algorithm 6 due to the screening rule. Note that both the second empirical moments and the partition of V in the algorithm can be computed in $O(p^2)$ operations (Witten et al., 2011; Mazumder and Hastie, 2012). On the contrary, the complexity of the exact optimization of a block of variables grows exponentially with respect to the maximal clique size of that block. Therefore, by encouraging enough sparsity, the blockwise minimization due to the screening rule can provide remarkable speedup by not only shrinking the size of the blocks in general but also potentially reducing the size of cliques within each block via eliminating enough edges.

9.4 Applications to Inexact (Alternative) Methods

We now discuss the interplay between the screening rule and two popular inexact (alternative) estimation methods: node-wise (NW) logistic regression (Wainwright et al., 2006; Ravikumar et al., 2010) and the pseudolikelihood (PL) method (Höfling and Tibshirani, 2009). In what follows, we use $\hat{\theta}^{\mathrm{NW}}$ and $\hat{\theta}^{\mathrm{PL}}$ to denote the solutions given by the node-wise logistic regression method and the pseudolikelihood method, respectively. NW can be considered as an *asymmetric* pseudolikelihood method (i.e., $\exists i,j \in V$ such that $i \neq j$ and $\hat{\theta}^{\mathrm{NW}}_{ij} \neq \hat{\theta}^{\mathrm{NW}}_{ji}$), while PL is a pseudolikelihood method that is similar to NW but imposes additional *symmetric* constraints on the parameterization (i.e., $\forall i,j \in V$ where $i \neq j$, we have $\hat{\theta}^{\mathrm{PL}}_{ij} = \hat{\theta}^{\mathrm{PL}}_{ji}$).

Our incorporation of the screening rule to the inexact methods is straightforward: after using the screening rule to identify different blocks in the solution, we use inexact methods to solve each block for the solution. As shown in Section 9.3, when combined with exact optimization, the screening rule is foolproof for blockwise structure identification. However, in general, when combined with inexact methods, the proposed screening rule is not foolproof any more because the screening rule is derived from the exact problem in (9.2) instead of the approximate problems such as NW and PL. We provide a toy example in 9.8.6 to illustrate mistakes made by the

screening rule when combined with inexact methods. Nonetheless, as we will show in this section, NW and PL are deeply connected to the screening rule, and when given a large enough regularization parameter, the application of the screening rule to NW and PL can be lossless in practice (see Section 9.6). Therefore, when applied to NW and PL, the proposed screening rule can be considered as a strong rule (i.e., a rule that is not foolproof but barely makes mistakes) and an optimal solution can be safeguarded by adjusting the screened solution to optimality based on the KKT conditions of the inexact problem (Tibshirani et al., 2012).

9.4.1 Node-wise (NW) Logistic Regression and the Pseudolikelihood (PL) Method

In NW, for each $i \in V$, we consider the conditional probability of X_i upon $X_{\setminus i}$, where $X_{\setminus i} = \{X_t \mid t \in V \setminus \{i\}\}$. This is equivalent to solving p ℓ_1-regularized logistic regression problems separately, i.e., $\forall i \in V$:

$$\hat{\theta}^{NW}_{\setminus i} = \arg\min_{\theta_{\setminus i}} \frac{1}{n} \sum_{k=1}^{n} \left[-y_i^{(k)} \eta_{\setminus i}^{(k)} + \log\left(1 + \exp\left(\eta_{\setminus i}^{(k)}\right)\right)\right] + \lambda \left\|\theta_{\setminus i}\right\|_1, \qquad (9.5)$$

where $\eta_{\setminus i}^{(k)} = \theta_{\setminus i}^{\top}(2x_{\setminus i}^{(k)})$, $y_i^{(k)} = 1$ represents a successful event $x_i^{(k)} = 1$, $y_i^{(k)} = 0$ represents an unsuccessful event $x_i^{(k)} = -1$, and

$$\theta_{\setminus i} = \begin{bmatrix} \theta_{i1} & \theta_{i2} & \cdots & \theta_{i(i-1)} & \theta_{i(i+1)} & \cdots & \theta_{ip} \end{bmatrix}^{\top},$$
$$x_{\setminus i}^{(k)} = \begin{bmatrix} x_{i1}^{(k)} & x_{i2}^{(k)} & \cdots & x_{i(i-1)}^{(k)} & x_{i(i+1)}^{(k)} & \cdots & x_{ip}^{(k)} \end{bmatrix}^{\top}.$$

Note that $\hat{\theta}^{NW}$ constructed from $\hat{\theta}^{NW}_{\setminus i}$'s is asymmetric, and ad hoc post processing techniques are used to generate a symmetric estimation such as setting each pair of elements from $\hat{\theta}^{NW}$ in symmetric positions to the one with a larger (or smaller) absolute value.

On the other hand, PL can be considered as solving all p ℓ_1-regularized logistic regression problems in (9.5) jointly with symmetric constraints over the parameter-

ization (Geng et al., 2017):

$$\hat{\theta}^{\mathrm{PL}} = \arg\min_{\theta \in \Theta} \frac{1}{n} \sum_{k=1}^{n} \sum_{i=1}^{p} \left[-y_i^{(k)} \xi_i^{(k)} + \log\left(1 + \exp\left(\xi_i^{(k)}\right)\right)\right] + \frac{\lambda}{2} \|\theta\|_1 , \qquad (9.6)$$

where $\xi_i^{(k)} = \sum_{j \in V \setminus \{i\}} 2\theta_{\min\{i,j\},\max\{i,j\}} x_j^{(k)}$. That is to say, if $i < j$, then $\theta_{\min\{i,j\},\max\{i,j\}} = \theta_{ij}$; if $i > j$, then $\theta_{\min\{i,j\},\max\{i,j\}} = \theta_{ji}$. Recall that Θ in (9.6) defined in Section 9.2.1 represents a space of symmetric matrices whose diagonal entries are zeros.

9.4.2 Regularization Parameters in NW and PL

Since the blockwise structure of a solution is given by the screening rule under a *fixed* regularization parameter, the ranges of regularization parameters under which NW and PL can return nonzero solutions need to be linked to the range $[0, \lambda_{\max}]$ in the exact problem. Theorem 9.5 and Theorem 9.6 establish such relationships for NW and PL, respectively.

Theorem 9.5. Let the dataset $\mathbb{X} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$ be given, and let $\lambda = \lambda_{\max}^{\mathrm{NW}}$ represent the smallest regularization parameter such that $\hat{\theta}_{\setminus i}^{\mathrm{NW}} = 0$ in (9.5), $\forall i \in V$. Then $\lambda_{\max}^{\mathrm{NW}} = \lambda_{\max}$.

Theorem 9.6. Let the dataset $\mathbb{X} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$ be given, and let $\lambda = \lambda_{\max}^{\mathrm{PL}}$ represent the smallest regularization parameter such that $\hat{\theta}^{\mathrm{PL}} = 0$ in (9.6), then $\lambda_{\max}^{\mathrm{PL}} = 2\lambda_{\max}$.

Let λ be the regularization parameter used in the exact problem. A strategy is to set the corresponding $\lambda^{\mathrm{NW}} = \lambda$ when using NW and $\lambda^{\mathrm{PL}} = 2\lambda$ when using PL, based on the range of regularization parameters given in Theorem 9.5 and Theorem 9.6 for NW and PL. Since the magnitude of the regularization parameter is suggestive of the magnitude of the gradient of the unregulated objective, the proposed strategy leverages that the magnitudes of the gradients of the unregulated objectives for NW and PL are roughly the same as, and roughly twice as large as, that of the unregulated exact objective, respectively.

This observation has been made in the literature of binary pairwise Markov networks (Höfling and Tibshirani, 2009; Viallon et al., 2014). Here, by Theorem 9.5 and Theorem 9.6, we demonstrate that this relationship is exactly true if the optimal parameterization is zero. Höfling and Tibshirani 2009 even further exploits this observation in PL for exact optimization. Their procedure can be viewed as iteratively solving adjusted PL problems regularized by $\lambda^{\mathrm{PL}} = 2\lambda$ in order to obtain an exact solution regularized by λ. The close quantitative correspondence between the derivatives of the inexact objectives and that of the exact objective also provides insights into why combing the screening rule with inexact methods does not lose much in practice.

9.4.3 Preservation for Fully Disconnectedness

While the screening rule is not foolproof when combined with NW and PL, it turns out that in terms of identifying fully disconnected nodes, the necessary and sufficient condition in Corollary 9.4 can be preserved when applying NW with caution, as shown in the following.

Theorem 9.7. Let the dataset $\mathbb{X} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$ be given. Let $\hat{\theta}^{\mathrm{NW}}_{\min} \in \Theta$ denote a symmetric matrix derived from $\hat{\theta}^{\mathrm{NW}}$ by setting each pair of elements from $\hat{\theta}^{\mathrm{NW}}$ in symmetric positions to the one with a smaller absolute value. A sufficient condition for X_i to be fully disconnected from the remaining nodes in $\hat{\theta}^{\mathrm{NW}}_{\min}$, where $i \in V$, is that $\lambda^{\mathrm{NW}} \geqslant \max_{j \in V \setminus \{i\}} |\mathbb{E}_{\mathbb{X}} X_i X_j|$. Furthermore, when $\hat{\theta}^{\mathrm{NW}}_{\setminus i} = 0$, the sufficient condition is also necessary.

In practice, the utility of Theorem 9.7 is to provide us a lower bound for λ above which we can fully disconnect X_i (sufficiency). Moreover, if $\hat{\theta}^{\mathrm{NW}}_{\setminus i} = 0$ also happens to be true, which is easily verifiable, we can conclude that such a lower bound is tight (necessity).

9.5 Generalization

With unary potentials, the ℓ_1-regularized MLE for the Ising model is defined as:

$$\hat{\theta} = \arg\min_{\theta} -\frac{1}{n}\sum_{k=1}^{n}\left(\sum_{i=1}^{p}\theta_{ii}x_i^{(k)} + \sum_{i=1}^{p-1}\sum_{j>i}^{p}\theta_{ij}x_i^{(k)}x_j^{(k)}\right) + A(\theta) + \frac{\lambda}{2}\|\theta\|_{1,\text{off}}, \quad (9.7)$$

where $\|\theta\|_{1,\text{off}} = \sum_{i=1}^{p}\sum_{j\neq i}^{p}|\theta_{ij}|$. Note that the unary potentials are not penalized, which is a common practice (Wainwright et al., 2006; Höfling and Tibshirani, 2009; Ravikumar et al., 2010; Viallon et al., 2014) to ensure a hierarchical parameterization. Formally, the generalized screening rule for Ising models with unary potentials is given by Theorem 9.8.

Theorem 9.8. Let a partition of V, $\{C_1, C_2, \cdots, C_L\}$, be given. Let the dataset $\mathbb{X} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$ be given. Define $\mathbb{E}_{\mathbb{X}}X_iX_j = \frac{1}{n}\sum_{k=1}^{n}x_i^{(k)}x_j^{(k)}$, and $\mathbb{E}_{\mathbb{X}}X_i = \frac{1}{n}\sum_{k=1}^{n}x_i^{(k)}$. A necessary and sufficient condition for $\hat{\theta}$ to be blockwise with respect to the given partition is that

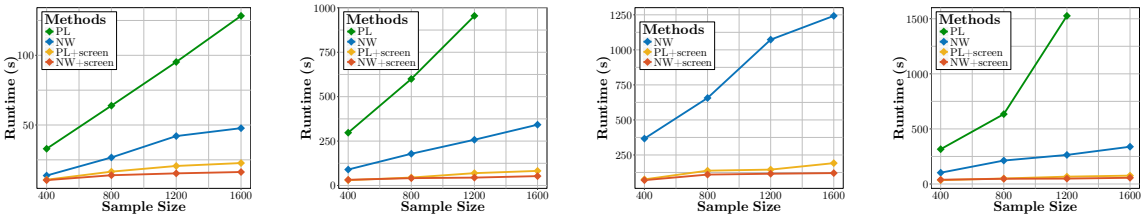
$$|\mathbb{E}_{\mathbb{X}}X_iX_j - \mathbb{E}_{\mathbb{X}}X_i\mathbb{E}_{\mathbb{X}}X_j| \leqslant \lambda, \quad (9.8)$$

for all l and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and for all $i \in C_l$, $j \in C_{l'}$.

The proof of Theorem 9.8 can be found in Section 9.8.10.

A most noteworthy consequence of Theorem 9.8 is that the blockwise structure of an Ising model with unary potentials can be identified *in the exact same way* as the blockwise structure of a Gaussian graphical model. This can be seen by comparing Theorem 9.8 with the results in Witten et al. (2011), and Mazumder and Hastie (2012). Such a correspondence between Ising models and Gaussian graphical models have striking implications.

Since Gaussian graphical models enjoy the precious property that the sparsity pattern of its precision matrix corresponds to the sparsity pattern of its structure, it might not be surprising that a screening rule for sample covariance matrix can offer an effective approach to identify the blockwise structure of a Gaussian graphical



(a) Network 1 (b) Network 2 (c) Network 3 (d) Network 4

Figure 9.1: Runtime of pathwise optimization on networks in Table 9.2. Runtime plotted is the median runtime over five trials. The experiments of the baseline method PL without screening can not be fully conducted on larger networks due to high memory cost. **NW**: Node-wise logistic regression without screening; **NW+screen**: Node-wise logistic regression with screening; **PL**: Pseudolikelihood without screening; **PL+screen**: Pseudolikelihood with screening.

model. On the contrary, in the regime of Ising models, in general there is no element-to-element exact sparsity pattern equivalence. Nonetheless, granted by Theorem 9.8, the block structure of an Ising model with unary potentials can still be identified by the same procedure as in the Gaussian case, which establishes an easily *verifiable* correspondence between the sample covariance matrix and the underlying structure for Ising models. This verifiable correspondence also distinguishes our work from Loh et al. (2012, 2013), where the correspondence between an *unverifiable* generalized precision matrix and the structure of a discrete graphical model is established. Our work is also different from Loh et al. (2012, 2013) in terms of the objective functions. While we consider the optimization perspective of the MLE problem in this work, the log-determinant problem is considered in Loh et al. (2012, 2013) with an emphasis on statistical consistency.

Furthermore, to the best of our knowledge, the screening rule in Witten et al. (2011) and Mazumder and Hastie (2012) is the strongest safe blockwise screening for Gaussian graphical models in the literature. Given the general intractability of discrete graphical model learning via maximum likelihood, the same safe screening achieved for Ising models provides an especially valuable and desperately needed guarantee that is as strong as the best known result for its polynomial-time Gaussian counterpart.

9.6 Experiments

Experiments are conducted on both synthetic data and real world data. We will focus on *efficiency* in Section 9.6.1 and discuss *support recovery* performance in Section 9.6.2. We consider three synthetic networks (Table 9.2) with 20, 35, and 50 blocks of 20-node, 35-node, and 50-node subnetworks, respectively. To demonstrate the estimation of networks with unbalanced-size subnetworks, we also consider a 46-block network with power law degree distributed subnetworks of sizes ranging from 5 to 50. Within each network, the subnetwork is generated according to a power law degree distribution, which mimics the structure of a biological network and is believed to be more challenging to recover compared with other less complicated structures (Chen and Sharp, 2004; Peng et al., 2009; Danaher et al., 2014). Each edge of each network is associated with a weight first sampled from a standard normal distribution, and then increased or decreased by 0.2 to further deviate from zero. For each network, 1600 samples are generated via Gibbs sampling *within each subnetwork*. Experiments on exact optimization are reported in 9.6.3.

9.6.1 Pathwise Optimization

Pathwise optimization aims to compute solutions over a range of different λ's. Formally, we denote the set of λ's used in (9.2) as $\Lambda = \{\lambda_1, \lambda_2, \cdots, \lambda_\tau\}$, and without loss of generality, we assume that $\lambda_1 < \lambda_2 < \cdots < \lambda_\tau$.

The introduction of the screening rule provides us insightful heuristics for the determination of Λ. We start by choosing a λ_1 that reflects the sparse blockwise structural assumption on the data. To achieve sparsity and avoid densely connected structures, we assume that the number of edges in the ground truth network is $O(p)$. This assumption coincides with networks generated according to a power law degree distribution and hence is a faithful representation of the prior knowledge stemming from many biological problems. As a heuristic, we relax and apply the screening rule in (9.4) on each of the $\binom{p}{2}$ second empirical moments and choose λ_1 such that the number of the absolute second empirical moments that are greater than λ_1 is about $p \log p$. Given a λ_1 chosen this way, one can check how many blocks

$\hat{\theta}(\lambda_1)$ has by the screening rule. To encourage blockwise structures, we magnify λ_1 via $\lambda_1 \leftarrow 1.05\lambda_1$ until the current $\hat{\theta}(\lambda_1)$ has more than one block. We then choose λ_τ such that the number of absolute second empirical moments that are greater than λ_τ is about p. In our experiments, we use an evenly spaced Λ with $\tau = 25$.

To estimate the networks in Table 9.2, we implement both NW and PL with and without screening using `glmnet` (Friedman et al., 2010) in `R` as a building block for logistic regression according to Ravikumar et al. 2010 and Geng et al. 2017. To generate a symmetric parameterization for NW, we set each pair of elements from θ^{NW} in symmetric positions to the element with a larger absolute value. Given Λ, we screen only at λ_1 to identify various blocks. Each block is then solved separately in a pathwise fashion under Λ without further screening. The rationale of performing only one screening is that starting from a λ_1 chosen in the aforementioned way has provided us a sparse blockwise structure that sets a significant portion of the parameterization to zeros; further screening over larger λ's hence does not necessarily offer more efficiency gain.

Figure 9.1 summarizes the runtime of pathwise optimization on the four synthetic networks in Table 9.2. The experiments are conducted on a PowerEdge R720 server with two Intel(R) Xeon(R) E5-2620 CPUs and 128GB RAM. As many as 24 threads can be run in parallel. For robustness, each runtime reported is the median runtime over five trials. When the sample size is less than 1600, each trial uses a subset of samples (subsamples) that are randomly drawn from the original datasets without replacement. As illustrated in Figure 9.1, the efficiency gain due to the screening rule is self-evident. Both NW and PL benefit substantially from the application of the screening rule. The speedup is more apparent with the increase of sample size as well as the increase of the dimension of the data. In our experiments, we observe that even with arguably the state-of-the-art implementation (Geng et al., 2017), PL without screening still has a significantly larger memory footprint compared with that of NW. Therefore, the experiments for PL without screening are not fully conducted in Figure 9.1b,9.1c, and 9.1d for networks with thousands of nodes. On the contrary, PL with the screening rule has a comparable memory footprint with that of NW. Furthermore, as shown in Figure 9.1, after applying the screening

rule, PL also has a similar runtime with NW. This phenomenon demonstrates the utility of the screening rule for effectively reducing the memory footprint of PL, making PL readily available for large-scale problems.

9.6.2 Model Selection

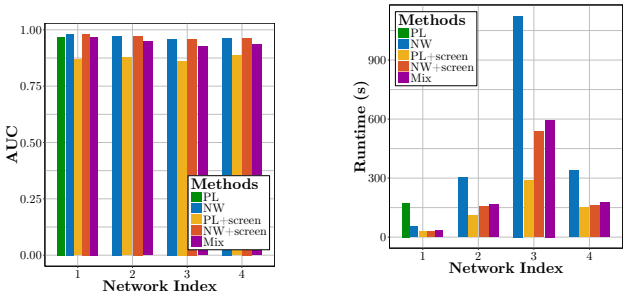
Our next experiment performs model selection by choosing an appropriate λ from the regularization parameter set Λ. We leverage the Stability Approach to Regularization Selection (StARS, Liu et al. 2010a) for this task. In a nutshell, StARS learns a set of various models, denoted as $\mathcal{M}$, over Λ using many subsamples that are drawn randomly from the original dataset without replacement. It then picks a $\lambda^* \in \Lambda$ that strikes the best balance between network sparsity and edge selection stability among the models in $\mathcal{M}$. After the determination of λ^*, it is used on the entire original dataset to learn a model with which we compare the ground truth model and calculate its support recovery Area Under Curve (AUC). Our model selection procedure is a variant of that in Liu et al. 2010a. To introduce enough variation, we neglect edges that do not show up in the solutions at least once under any $\lambda \in \Lambda$ when computing the total instability defined in Liu et al. 2010a. We choose $\beta = 0.1$ defined in the paper. We refer interested readers to the paper for the details of StARS.

In Figure 9.2, we summarize the experimental results of model selection, where 24 subsamples are used for pathwise optimization in parallel to construct $\mathcal{M}$. In Figure 9.2a, NW with and without screening achieve the same high AUC values over all four networks, while the application of the screening rule to NW provides roughly a 2x speedup, according to Figure 9.2b. The same AUC value shared by the two variants of NW is due to the same λ^* chosen by the model selection procedure. Even more importantly, it is also because that under the same λ^*, the screening rule is able to *perfectly* identify the blockwise structure of the parameterization.

Due to high memory cost, the model selection for PL without screening (green bars in Figure 9.2) is omitted in some networks. To control the memory footprint, the model selection for PL with screening (golden bars in Figure 9.2) also needs to

indx	#blk	#nd/blk	TL#nd
1	20	20	400
2	35	35	1225
3	50	50	2500
4	46	5-50	1265

Table 9.2: Summary of the four synthetic networks used in the experiments. `indx` represents the index of each network. `#blk` represents the number of blocks each network has. `#nd/blk` represents the number of nodes each block has. `TL#nd` represents the total number of nodes each network has.



(a) Edge recovery AUC

(b) Model selection runtime

Figure 9.2: Model selection performance. **Mix**: provide PL +screen with the regularization parameter chosen by the model selection of NW+screen. Other legend labels are the same as in Figure 9.1.

be carried out meticulously by avoiding small λ's in Λ that correspond to dense structures in $\mathcal{M}$ during estimation from subsamples. While avoiding dense structures makes PL with screening the fastest among all (Figure 9.2b), it comes at the cost of delivering the least accurate (though still reasonably effective) support recovery performance (Figure 9.2a). To improve the accuracy of this approach, we also leverage the connection between NW and PL by substituting $2\lambda^*_{\text{NW}}$ for the resultant regularization parameter from model selection of PL, where λ^*_{NW} is the regularization parameter selected for NW. This strategy results in better performance in support recovery (purple bars in Figure 9.2a).

9.6.3 Exact Optimization

To demonstrate the efficiency gain provided by the screening rule in exact optimization, we consider a dataset of 1600 samples generated from a network with 16 power law degree distributed subnetworks of 16 nodes. We select λ^*_{NW} using the model selection procedure in Section 9.6.2 and compute the exact solution under λ^*_{NW} using the proximal gradient method with constant step length (Pena and Tib-

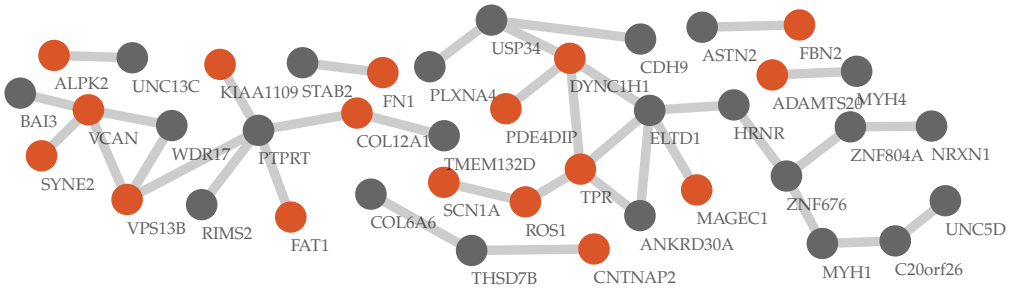
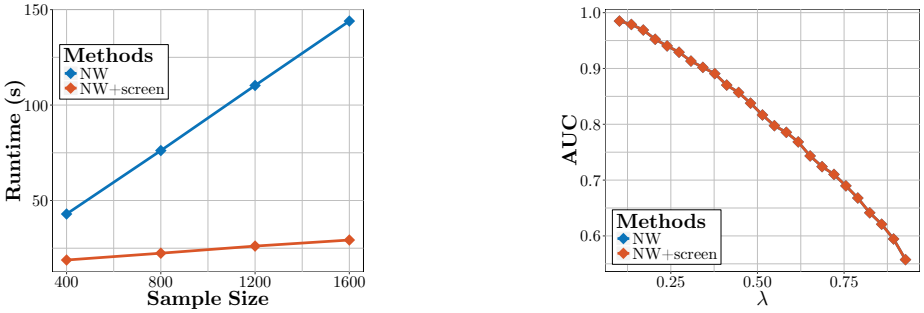


Figure 9.3: Connected components learned from lung squamous cell carcinoma mutation data. Genes in red are (lung) cancer and other disease related genes (Uhlén et al., 2015). Mutation data are extracted via the `TCGA2STAT` package (Wan et al., 2015) in `R` and the figure is rendered by `Cytoscape`.

shirani, 2016). Under λ^*_{NW}, the network can be successfully divided into 16 blocks according to the screening rule. Without further assumption on the structure of the subnetworks, we then compute the solution to each block separately in parallel using the NW solution as initialization. The problem can be solved in about 90 seconds. Since there are 256 nodes in the network, exact optimization in this fashion would be unimaginable had the screening rule not been applied to this problem.

9.6.4 Real World Data

Our real world data experiment applies NW with and without screening to a real world gene mutation dataset collected from 178 lung squamous cell carcinoma samples (Weinstein et al., 2013). Each sample contains 13,665 binary variables representing the mutation statuses of various genes. For ease of interpretation, we keep genes whose mutation rates are at least 10% across all samples, yielding a subset of 145 genes in total. We use the model selection procedure introduced in Section 9.6.2 to determine a λ^*_{NW} with which we learn the gene mutation network whose connected components are shown in Figure 9.3. For model selection, other than the configuration in 9.6.2, we choose $\tau = 25$. 384 trials are run in parallel using all 24 threads. We also choose λ_1 such that about $2p\log(p)$ absolute second empirical moments are greater than λ_1. We choose λ_τ such that about $0.25p$ absolute



(a) Runtime v.s. Sample size

(b) AUC v.s. λ using all samples

Figure 9.4: Runtime and support recovery performance for Ising models with unary potentials. Note that in in Figure 9.4b, the two curves overlap.

second empirical moments are greater than λ_τ.

In our experiment, NW with and without screening select the same λ^*_{NW}, and generate the same network. Since the dataset in question has a lower dimension and a smaller sample size compared with the synthetic data, NW without screening is adequately efficient. Nonetheless, with screening NW is still roughly 20% faster. This phenomenon once again indicates that in practice the screening rule can perfectly identify the blockwise sparsity pattern in the parameterization and deliver a significant efficiency gain. The genes in red in Figure 9.3 represent (lung) cancer and other disease related genes, which are scattered across the seven subnetworks discovered by the algorithm. In our experiment, we also notice that *all* the weights on the edges are positive. This is consistent with the biological belief that associated genes tend to *mutate together* to cause cancer.

9.6.5 Experiments in the Generalized Setting

To demonstrate the utility of the screening rule for Ising models with unary potentials, we generate a network that consists of 40 power law degree distributed subnetworks of 20 nodes. The weights on the edges are generated in the same way as in Section 9.6. The weights on all nodes are set to be 0.1 for simplicity. As many as 1600 samples are used for learning. Figure 9.4 reports the runtime as well

as the AUC of pathwise optimization using NW with and without screening for Ising models with unary potentials. The phenomenon we observed in this case is consistent with the phenomenon for Ising models with only pairwise potentials. The screening can accelerate learning tremendously and in this experiment even delivers lossless screening. This can be seen from Figure 9.4b, where the AUC v.s. λ curves of NW with and without screening completely overlap with each other.

9.7 Conclusion

We have proposed a screening rule for ℓ_1-regularized Ising model estimation. The simple closed-form screening rule is a necessary and sufficient condition for exact blockwise structural identification. Experimental results suggest that the proposed screening rule can provide drastic speedups for learning when combined with various optimization algorithms. Future directions include deriving screening rules for more general undirected graphical models (Liu et al., 2012, 2014c,b; Liu, 2014; Liu et al., 2016), and deriving screening rules for other inexact optimization algorithms (Liu and Page, 2013b). Further theoretical justifications regarding the conditions upon which the screening rule can be combined with inexact algorithms to recover block structures losslessly are also desirable.

9.8 Auxiliary Results

9.8.1 A Lemma and a Theorem

We first show that the following lemma is true with classic graphical model inference techniques (Koller and Friedman, 2009):

Lemma 9.9. Let $\theta \in \Theta$ be given, and let C_l and $C_{l'}$ be two elements of a partition of V, where $l \neq l'$. If the nodes in C_l are not connected with the nodes in $C_{l'}$,

i.e., $\forall i \in C_l$ and $\forall j \in C_{l'}$, $\theta_{ij} = 0$, then

$$\mathbb{E}_\theta X_i X_j = \sum_{x \in \mathcal{X}} x_i x_j P_\theta(x) = 0. \tag{9.9}$$

Proof. Without loss of generality, suppose V is partitioned as $\{C_l, C_{l'}\}$. Since C_l and $C_{l'}$ are disconnected, $P_\theta(x) = P_{C_l}(x) P_{C_{l'}}(x)$, where $P_{C_l}(x)$ and $P_{C_{l'}}(x)$ represent the marginal distributions among the variables indexed by C_l and $C_{l'}$, respectively. Therefore, $\forall i \in C_l$ and $\forall j \in C_{l'}$,

$$\mathbb{E}_\theta X_i X_j = \sum_{x \in \mathcal{X}} x_i x_j P_\theta(x) = \sum_{x \in \mathcal{X}} x_i x_j P_{C_l}(x) P_{C_{l'}}(x) = \sum_{\substack{x_i, x_j \in \\ \{-1,1\}}} x_i x_j P(x_i) P(x_j), \tag{9.10}$$

By a symmetric argument, one can show that $P(x_i) = \frac{1}{2}$, $\forall i \in V$. Therefore, in (9.10), $\mathbb{E}_\theta X_i X_j = 0$. □

In (9.9), $\mathbb{E}_\theta X_i X_j$ represents the element at the i^{th} row and the j^{th} column of the expectation of the second moment of the random vector X about the origin under $P_\theta(x)$, $\mathbb{E}_\theta X X^\top$. The theorem establishes the sparsity pattern correspondence between θ and $\mathbb{E}_\theta X X^\top$ for *any* given $\theta \in \Theta$. In Section 9.3, we will see its significant role played in the derivation of the screening rule.

If we can identify the blockwise structure of $\hat{\theta}$ in advance, we can solve each block independently due to the following theorem.

Theorem 9.10. If $\hat{\theta}$ is blockwise as shown in (9.3), we can identify $\hat{\theta}$ by solving, $\forall l \in \{1, 2, \cdots, L\}$, separately for:

$$\hat{\theta}_l = \arg\min_{\theta_l} -\frac{1}{n} \sum_{k=1}^{n} \sum_{i=1}^{|C_l|-1} \sum_{j>i}^{|C_l|} \theta_{lij} x_i^{(k)} x_j^{(k)} + A(\theta_l) + \frac{\lambda}{2} \|\theta_l\|_1,$$

where $|C_l|$ represents the cardinality of C_l.

Proof. Theorem 9.10 can be proved by inspection. □

9.8.2 Optimality Conditions

Another essential element for the derivation of the screening rule is the Karush-Kuhn-Tucker (KKT) conditions for the ℓ_1-regularized Ising model. Let $i \in V$, and $j > i$ be given, the KKT condition with respect to $\hat{\theta}_{ij}$ is given by:

$$\mathbb{E}_{\hat{\theta}} X_i X_j - \mathbb{E}_{\mathbb{X}} X_i X_j + \lambda t_{ij} = 0, \tag{9.11}$$

where $\mathbb{E}_{\mathbb{X}} X_i X_j = \frac{1}{n} \sum_{k=1}^{n} x_i^{(k)} x_j^{(k)}$'s are second empirical moments from the second empirical moment matrix $\mathbb{E}_{\mathbb{X}} XX^\top$, and t_{ij} is the component of a subgradient that corresponds to $\hat{\theta}_{ij}$, with $t_{ij} = 1$ when $\hat{\theta}_{ij} > 0$, $t_{ij} = -1$ when $\hat{\theta}_{ij} < 0$, and $t_{ij} \in [-1, 1]$ when $\hat{\theta}_{ij} = 0$. Since the minimization problem for the ℓ_1-regularized Ising model in (9.2) is a convex problem, the KKT conditions can be satisfied if and only if (9.2) reaches its optimal solution $\hat{\theta}$.

9.8.3 Proof of Theorem 9.2

Proof. The rationale behind our proof is similar to that in Witten et al. 2011:

- We first prove necessity. Since $\hat{\theta}$ is blockwise, by Lemma 9.9, $\mathbb{E}_{\hat{\theta}} X_i X_j = 0$, for all l and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and for all $i \in C_l$, $j \in C_{l'}$. By the KKT condition in (9.11), $\lambda t_{ij} = \mathbb{E}_{\mathbb{X}} X_i X_j - \mathbb{E}_{\hat{\theta}} X_i X_j = \mathbb{E}_{\mathbb{X}} X_i X_j \Rightarrow |\mathbb{E}_{\mathbb{X}} X_i X_j| \leqslant \lambda$, for all l and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and for all $i \in C_l$, $j \in C_{l'}$. Note that we have used the fact that $\hat{\theta}_{ij} = 0 \Rightarrow |t_{ij}| \leqslant 1$.

- We then prove sufficiency via construction techniques. Specifically, we construct a blockwise $\tilde{\theta}$ and show that $\tilde{\theta}$ satisfies KKT conditions so that $\tilde{\theta}$ is, in fact, optimal, i.e., $\tilde{\theta} = \hat{\theta}$. For this purpose, we first set all the off-block-diagonal elements in $\tilde{\theta}$ that satisfy (9.4) to zeros. In this way, $\tilde{\theta}$ is blockwise with respect to the partition $\{C_1, C_2, \cdots, C_L\}$ and hence Lemma 9.9 can be applied. The consequence is that $\mathbb{E}_{\tilde{\theta}} X_i X_j = 0$, for all l and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and for all $i \in C_l$, $j \in C_{l'}$. Therefore, the KKT conditions for these off-block-diagonal zero elements of $\tilde{\theta}$ can be satisfied. Furthermore,

now that $\tilde{\theta}$ is blockwise, the block diagonal elements can also be computed via exact optimization separately. In this way, the KKT conditions for the block diagonal elements of $\tilde{\theta}$ can also be satisfied. We have shown that all the elements in $\tilde{\theta}$ satisfy KKT conditions. Therefore, $\tilde{\theta}$ constructed in this way is indeed optimal and hence $\tilde{\theta} = \hat{\theta}$.

□

9.8.4 Proof of Theorem 9.3

Proof. When $\hat{\theta} = 0$, all the nodes are disconnected from each other, which is equivalent to considering the fully disconnected partition $\{\{1\}, \{2\}, \cdots, \{p\}\}$. Using this partition, by Theorem 9.2, it is necessary and sufficient for $\lambda_{max} = \max_{i,j \in V, i \neq j} |\mathbb{E}_{\mathbb{X}} X_i X_j|$ to guarantee that $\hat{\theta} = 0$. Furthermore, since X_i, $X_j \in \{-1, 1\}$, $\forall i, j \in V$, we have $\max_{i,j \in V, i \neq j} |\mathbb{E}_{\mathbb{X}} X_i X_j| \leqslant 1 \Rightarrow \lambda_{max} \leqslant 1$. □

9.8.5 Proof of Corollary 9.4

Proof. Applying Theorem 9.2 to any partition with an element $\{i\}$ yields the result.

□

9.8.6 A Toy Example

We consider a dataset with three variables and five samples. i.e. $p = 3$, and $n = 5$. Specifically,

$$\mathbb{X} = \begin{bmatrix} -1 & 1 & -1 \\ -1 & -1 & -1 \\ -1 & -1 & -1 \\ -1 & -1 & 1 \\ 1 & -1 & 1 \end{bmatrix}, \quad \mathbb{E}_{\mathbb{X}} X X^{\top} = \begin{bmatrix} 1 & 0.2 & 0.6 \\ 0.2 & 1 & -0.2 \\ 0.6 & -0.2 & 1 \end{bmatrix}.$$

Therefore, according to the screening rule (Theroem 9.2 or Corollary 9.4), if we set $\lambda = 0.2$, X_2 should be disconnected from X_1 and X_3 in $\hat{\theta}$. Solving the exact problem

with $\lambda = 0.2$ confirms this proposition:

$$\hat{\theta} = \begin{bmatrix} 0 & 0 & 0.4237578 \\ 0 & 0 & 0 \\ 0.4237578 & 0 & 0 \end{bmatrix}.$$

Furthermore, with $\lambda = 0.2$,

$$\hat{\theta}^{\mathrm{NW}} = \begin{bmatrix} 0 & 0.1013663 & 0.4479399 \\ 0 & 0 & 0 \\ 0.4479399 & -0.1013663 & 0 \end{bmatrix},$$

$$\hat{\theta}^{\mathrm{NW}}_{\mathrm{min}} = \begin{bmatrix} 0 & 0 & 0.4479399 \\ 0 & 0 & 0 \\ 0.4479399 & 0 & 0 \end{bmatrix},$$

$$\hat{\theta}^{\mathrm{PL}} = \begin{bmatrix} 0 & 0.06702585 & 0.43879982 \\ 0.06702585 & 0 & -0.06702585 \\ 0.43879982 & -0.06702585 & 0 \end{bmatrix}.$$

This suggests that X_1, X_2, and X_3 are connected in $\hat{\theta}^{\mathrm{NW}}$ and $\hat{\theta}^{\mathrm{PL}}$, and the screening rule makes mistakes in this example. However, in $\hat{\theta}^{\mathrm{NW}}_{\mathrm{min}}$, X_2 is fully disconnected from X_1 and X_3, which is guaranteed by Theorem 9.7.

9.8.7 Proof of Theorem 9.5

Proof. Let $j \in \{1, 2, \cdots, p-1\}$ be given, by the KKT conditions of (9.5), for the $\hat{\theta}^{\mathrm{NW}}_{\setminus i,j}$ component,

$$\frac{1}{n}\sum_{k=1}^{n} 2x^{(k)}_{\setminus i,j}\left(y^{(k)}_i - \frac{1}{1+\exp\left(-\hat{\eta}^{(k)}_{\setminus i}\right)}\right) = \lambda t_j, \tag{9.12}$$

where t_j is the j^{th} component of the subgradient. Since $\lambda = \lambda_{max}^{NW} \Leftrightarrow \hat{\theta}^{NW} = 0 \Rightarrow \hat{\eta}_{\backslash i}^{(k)} = 0, \forall i \in V, \forall k$, we have that

$$y_i^{(k)} - \frac{1}{1 + \exp\left(-\hat{\eta}_{\backslash i}^{(k)}\right)} = y_i^{(k)} - \frac{1}{2} = \frac{1}{2} x_i^{(k)}. \tag{9.13}$$

Substitute (9.13) into (9.12) yields $|\mathbb{E}_{\mathbb{X}} X_i X_j| \leqslant \lambda_{max}^{NW} = \lambda_{max}$, where we have used the fact that $|t_j| \leqslant 1$ and Theorem 9.3. □

9.8.8 Proof of Theorem 9.6

Proof. We follow an argument that is similar to the proof of Theorem 9.5. Specifically, without loss of generality, we consider the case where $i < j$. When $\lambda = \lambda_{max}^{PL}$, by the KKT conditions of (9.6) with respect to $\hat{\theta}_{ij}^{PL}$:

$$\left| \frac{1}{n} \sum_{k=1}^{n} \left[2x_j^{(k)} \left(y_i^{(k)} - \frac{1}{2} \right) + 2x_i^{(k)} \left(y_j^{(k)} - \frac{1}{2} \right) \right] \right| = \left| \frac{2}{n} \sum_{k=1}^{n} x_i^{(k)} x_j^{(k)} \right| \leqslant \lambda_{max}^{PL}$$
$$\Rightarrow |\mathbb{E}_{\mathbb{X}} X_i X_j| \leqslant \frac{\lambda_{max}^{PL}}{2}.$$

Using Theorem 9.3 we have that $\lambda_{max}^{PL} = 2\lambda_{max}$. □

9.8.9 Proof of Theorem 9.7

Proof. We first prove necessity. $\hat{\theta}_{\backslash i}^{NW} = 0 \Rightarrow \hat{\eta}_{\backslash i}^{(k)} = 0, \forall k \Rightarrow$ (9.13) can be satisfied $\Rightarrow$ (9.12) can be satisfied using (9.13) $\Rightarrow \lambda^{NW} \geqslant \max_{j \in V \setminus \{i\}} |\mathbb{E}_{\mathbb{X}} X_i X_j|$. Note that $\hat{\theta}_{\backslash i}^{NW} = 0$ implies that X_i is fully disconnected in $\hat{\theta}_{min}^{NW}$. We then prove sufficiency. To this end, $\forall j \in V \setminus \{i\}$, we set $\tilde{\theta}_{ij}^{NW} = 0$. That is to say, $\tilde{\theta}_{\backslash i}^{NW} = 0$. Following the same rationale behind the proof of necessity, and using the assumption that $\lambda^{NW} \geqslant \max_{j \in V \setminus \{i\}} |\mathbb{E}_{\mathbb{X}} X_i X_j|$, the KKT conditions for $\tilde{\theta}_{\backslash i}^{NW} = 0$ can be satisfied. The KKT conditions for $\tilde{\theta}_{\backslash j}^{NW}$'s, where $j \in V \backslash \{i\}$ can be trivially satisfied by solving the corresponding penalized logistic regression problems. Therefore, $\tilde{\theta}^{NW}$ is indeed optimal. i.e. $\tilde{\theta}^{NW} = \hat{\theta}^{NW}$. Furthermore, by the definition of $\hat{\theta}_{min}^{NW}$, $\left(\hat{\theta}_{min}^{NW}\right)_{ij} = \left(\hat{\theta}_{min}^{NW}\right)_{ji} = 0$

because $\tilde{\theta}^{NW}_{\setminus i} = 0$. Therefore, X_i is fully disconnected from the remaining nodes in $\hat{\theta}^{NW}_{min}$. □

9.8.10 Proof of Theorem 9.8

To show that Theorem 9.8 is true, we first show that the following lemma is true:

Lemma 9.11. Let θ be given, and let C_l and $C_{l'}$ be two elements of a partition of V, where $l \neq l'$. If the nodes in C_l are not connected with the nodes in $C_{l'}$, i.e., $\forall i \in C_l$ and $\forall j \in C_{l'}$, $\theta_{ij} = 0$, then

$$\mathbb{E}_\theta X_i X_j = \mathbb{E}_\theta X_i \mathbb{E}_\theta X_j. \tag{9.14}$$

Proof. Without loss of generality, suppose V is partitioned as $\{C_l, C_{l'}\}$. Since C_l and $C_{l'}$ are disconnected, $P_\theta(x) = P_{C_l}(x)P_{C_{l'}}(x)$, where $P_{C_l}(x)$ and $P_{C_{l'}}(x)$ represent the marginal distributions among the variables indexed by C_l and $C_{l'}$, respectively. Therefore, $\forall i \in C_l$ and $\forall j \in C_{l'}$,

$$\begin{aligned}
\mathbb{E}_\theta X_i X_j =& \sum_{x \in \mathcal{X}} x_i x_j P_\theta(x) = \sum_{x \in \mathcal{X}} x_i x_j P_{C_l}(x) P_{C_{l'}}(x) \\
=& \sum_{\substack{x_i, x_j \in \\ \{-1,1\}}} x_i x_j P(x_i) P(x_j) = \left(\sum_{x_i \in \{-1,1\}} x_i P(x_i) \right) \left(\sum_{x_j \in \{-1,1\}} x_j P(x_j) \right) \\
=& \mathbb{E}_\theta X_i \mathbb{E}_\theta X_j.
\end{aligned}$$

□

Consider the KKT conditions for (9.7). The KKT condition for $\hat{\theta}_{ii}$ is:

$$\mathbb{E}_{\mathbb{X}} X_i = \mathbb{E}_{\hat{\theta}} X_i. \tag{9.15}$$

The KKT condition for $\hat{\theta}_{ij}$, where $i \neq j$, is:

$$\mathbb{E}_{\hat{\theta}} X_i X_j - \mathbb{E}_{\mathbb{X}} X_i X_j + \lambda t_{ij} = 0. \tag{9.16}$$

We are now ready to prove Theorem 9.8 as follows.

Proof. We first prove necessity. Since $\hat{\theta}$ is blockwise, by Lemma 9.11, $\mathbb{E}_{\hat{\theta}}X_iX_j = \mathbb{E}_{\hat{\theta}}X_i\mathbb{E}_{\hat{\theta}}X_j$, for all l and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and for all $i \in C_l$, $j \in C_{l'}$. By the KKT condition in (9.15) and (9.16), $\lambda t_{ij} = \mathbb{E}_{\mathbb{X}}X_iX_j - \mathbb{E}_{\hat{\theta}}X_iX_j = \mathbb{E}_{\mathbb{X}}X_iX_j - \mathbb{E}_{\hat{\theta}}X_i\mathbb{E}_{\hat{\theta}}X_j = \mathbb{E}_{\mathbb{X}}X_iX_j - \mathbb{E}_{\mathbb{X}}X_i\mathbb{E}_{\mathbb{X}}X_j \Rightarrow |\mathbb{E}_{\mathbb{X}}X_iX_j - \mathbb{E}_{\mathbb{X}}X_i\mathbb{E}_{\mathbb{X}}X_j| \leqslant \lambda$, for all l and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and for all $i \in C_l$, $j \in C_{l'}$. Note that we have used the fact that $\hat{\theta}_{ij} = 0 \Rightarrow |t_{ij}| \leqslant 1$.

We then prove sufficiency via construction techniques. Specifically, we construct a blockwise $\tilde{\theta}$ and show that $\tilde{\theta}$ satisfies KKT conditions so that $\tilde{\theta}$ is, in fact, optimal, i.e., $\tilde{\theta} = \hat{\theta}$. For this purpose, we first set all the off-block-diagonal elements in $\tilde{\theta}$ that satisfy (9.4) to zeros. In this way, $\tilde{\theta}$ is blockwise with respect to the partition $\{C_1, C_2, \cdots, C_L\}$ and hence Lemma 9.11 can be applied. The consequence is that $\mathbb{E}_{\tilde{\theta}}X_iX_j = \mathbb{E}_{\tilde{\theta}}X_i\mathbb{E}_{\tilde{\theta}}X_j$, for all l and $l' \in \{1, 2, \cdots, L\}$, where $l \neq l'$, and for all $i \in C_l$, $j \in C_{l'}$. Therefore, the KKT conditions for these off-block-diagonal zero elements of $\tilde{\theta}$ can be satisfied. Furthermore, now that $\tilde{\theta}$ is blockwise, the block diagonal elements can also be computed via exact optimization separately. In this way, the KKT conditions for the block diagonal elements of $\tilde{\theta}$ can also be satisfied. We have shown that all the elements in $\tilde{\theta}$ satisfy KKT conditions. Therefore, $\tilde{\theta}$ constructed in this way is indeed optimal and hence $\tilde{\theta} = \hat{\theta}$. □

Part V

Epilogue

10 CONCLUSION

In this dissertation, we have presented machine learning models and methods to identify potential causal relationships among various event types from longitudinal event data in the hope of gaining actionable insights for better decision-making. As a concrete example, we consider the use of electronic health records for two pivotal health applications: computational drug repositioning and adverse drug reaction discovery.

We focus on developing machine learning models and algorithms with high causal fidelity: by confronting various theoretical, methodological, and empirical issues stemming from the intricacies of LED, our models and algorithms strive to identify signals in LED that are reflective of potential causal relationships encoded in the data. Towards high causal fidelity, we identify and address three fundamental challenges constitutional to the intrinsic nature of LED - *inhomogeneity*, *irregularity*, and *interplay* - summarized as the 3-I challenge. Our studies demonstrate that a careful treatment of the 3-I challenge can lead to machine learning models and algorithms with high causal fidelity, as shown by the improved performance of CDR and ADR discovery exhibited in this dissertation.

REFERENCES

U.S. Senate. http://www.senate.gov/index.htm.

AACE. Management of common comorbidities of diabetes. http://outpatient.aace.com/type-2-diabetes/management-of-common-comorbidities-of-diabetes.

Abe, Masanori, Seiya Nakamura, Tatsuya Higa, Junichi Okubo, and Manabu Kakinohana. 2015. Frequent hypoglycemia after prescription of pregabalin in a patient with painful diabetic neuropathy. *Journal of Japan Society of Pain Clinicians* advpub.

ADA. Skin complications. http://www.diabetes.org/living-with-diabetes/complications/skin-complications.html.

Allen, Genevera I, and Zhandong Liu. 2013. A local poisson graphical model for inferring networks from sequencing data. *IEEE Transactions on Nanobioscience* 12(3):189–198.

Amsterdam, Jay D, Justine Shults, Nancy Rutherford, and Stanley Schwartz. 2006. Safety and efficacy of s-citalopram in patients with co-morbid major depression and diabetes mellitus. *Neuropsychobiology*.

Arnold, Taylor, Veeranjaneyulu Sadhanala, and Ryan J Tibshirani. 2014. glmgen: fast generalized lasso solver.

Ashburn, Ted T, and Karl B Thor. 2004. Drug repositioning: identifying and developing new uses for existing drugs. *Nature Reviews Drug Discovery*.

Atchade, YVES F, GERSENDE Fort, and ERIC Moulines. 2014. On stochastic proximal gradient algorithms.

Balakumar, Pitchai, Rajavel Varatharajan, Ying Nyo, Raja Renushia, Devarajan Raaginey, Ann Oh, Shaikh Akhtar, Mani Rupeshkumar, Karupiah Sundram, and Sokkalingam A Dhanaraj. 2014. Fenofibrate and dipyridamole treatments in low-doses either alone or in combination blunted the development of nephropathy in diabetic rats. *Pharmacological Research*.

Banerjee, Onureena, Laurent El Ghaoui, and Alexandre dâŁ™Aspremont. 2008. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine Learning Research* 9(Mar):485–516.

Bao, Yujia, Zhaobin Kuang, Peggy Peissig, David Page, and Rebecca Willett. 2017a. Hawkes process modeling of adverse drug reactions with longitudinal observational data. In *Machine learning for healthcare conference*, 177–190.

———. 2017b. Hawkes process modeling of adverse drug reactions with longitudinal observational data. In Bao et al. (2017a), 177–190.

Barber, Rina Foygel, Mathias Drton, and Others. 2015. High-dimensional Ising model selection with Bayesian information criteria. *Electronic Journal of Statistics* 9(1):567–607.

Bastian, Mathieu, Sebastien Heymann, Mathieu Jacomy, and Others. 2009. Gephi: an open source software for exploring and manipulating networks. *ICWSM* 8:361–362.

Bate, Andrew, Robert F Reynolds, and Patrick Caubel. 2018. The hope, hype and reality of Big Data for pharmacovigilance.

Beck, Amir, and Marc Teboulle. 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences* 2(1):183–202.

Bengio, Yoshua, and Olivier Delalleau. 2009. Justifying and generalizing contrastive divergence. *Neural Computation* 21(6):1601–1621.

Bühlmann, Peter, and Sara Van De Geer. 2011. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media.

Carter, Aleesa A, Tara Gomes, Ximena Camacho, David N Juurlink, Baiju R Shah, and Muhammad M Mamdani. 2013. Risk of incident diabetes among patients treated with statins: population based study. *BMJ*.

CDC. Smoking and diabetes. `http://www.cdc.gov/tobacco/campaign/tips/diseases/diabetes.html`.

Chavez-Demoulin, Valérie, and J A McGill. 2012. High-frequency financial data modeling using Hawkes processes. *Journal of Banking & Finance* 36(12):3415–3426.

Chen, Hao, and Burt M Sharp. 2004. Content-rich biological network constructed by mining PubMed abstracts. *BMC Bioinformatics* 5(1):147.

Chow, Bacon F, and Howard H Stone. 1957. The relationship of vitamin B12 to carbohydrate metabolism and diabetes mellitus. *The American Journal of Clinical Nutrition*.

Condat, Laurent. 2013. A direct algorithm for 1D total variation denoising. *IEEE Signal Processing Letters* 20(11):1054–1057.

Daley, Daryl J, and David Vere-Jones. 2003. *An introduction to the theory of point processes, vol. I: probability and its applications*. 2nd ed. New York: Springer-Verlag.

Damci, Taner, Serkan Tatliagac, Zeynep Osar, and Hasan Ilkova. 2003. Fenofibrate treatment is associated with better glycemic control and lower serum leptin and insulin levels in type 2 diabetic patients with hypertriglyceridemia. *European Journal of Internal Medicine*.

Danaher, Patrick, Pei Wang, and Daniela M Witten. 2014. The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 76(2):373–397.

Daniel, R M, S N Cousens, De B L Stavola, M G Kenward, and J A C Sterne. 2013. Methods for dealing with time dependent confounding. *Statistics in Medicine*.

Davies, P Laurie, and Arne Kovac. 2001. Local extremes, runs, strings and multiresolution. *Annals of Statistics* 1–48.

DiabetesInControl. 2015. Drugs that can affect blood glucose levels. `http://www.diabetesincontrol.com/wp-content/uploads/2010/07/www.diabetesincontrol.com_images_tools_druglistaffectingbloodglucose.pdf`.

Du, Nan, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. 2016. Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 1555–1564. ACM.

Ertekin, \cSeyda, Cynthia Rudin, Tyler H McCormick, and Others. 2015. Reactive point processes: a new approach to predicting power failures in underground electrical systems. *The Annals of Applied Statistics* 9(1):122–144.

Farrington, C P. 1995. Relative incidence estimation from case series for vaccine safety evaluation. *Biometrics* 228–235.

Farrington, Paddy, Heather Whitaker, and Yonas Ghebremichael Weldeselassie. 2018. Self-controlled case series studies: A modelling guide with r.

FDA. a. CIPRO medication guide. `https://www.fda.gov/downloads/Drugs/DrugSafety/UCM088572.pdf`.

———. b. Lipitor (atorvastatin calcium) tablets. `https://www.accessdata.fda.gov/drugsatfda_docs/label/2009/020702s057lbl.pdf`.

———. c. Premarin (conjugated estrogens tablets, USP). `http://www.accessdata.fda.gov/drugsatfda_docs/label/2006/004782s147lbl.pdf`.

———. d. ZESTRIL (lisinopril) label. `https://www.accessdata.fda.gov/drugsatfda_docs/label/2009/019777s054lbl.pdf`.

———. 2014. Consumer updates: FDA expands advice on statin risks. `http://www.fda.gov/ForConsumers/ConsumerUpdates/ucm293330.htm`.

Fercoq, Olivier, Alexandre Gramfort, and Joseph Salmon. 2015. Mind the duality gap: safer rules for the Lasso. In *Proceedings of the 32nd international conference on machine learning*, 333–342.

Findlay, Steven. 2015. Health policy briefs: The FDA's sentinel initiative. *Health Affiaris*.

Fischer, Asja. 2015. Training restricted Boltzmann machines. *KI-Künstliche Intelligenz* 29(4):441–444.

Fischer, Asja, and Christian Igel. 2011. Bounding the bias of contrastive divergence learning. *Neural Computation* 23(3):664–673.

Freeman, Dilys J, John Norrie, Naveed Sattar, R Dermot G Neely, Stuart M Cobbe, Ian Ford, Christopher Isles, A Ross Lorimer, Peter W Macfarlane, James H McKillop, and Others. 2001. Pravastatin and the development of diabetes mellitus evidence for a protective treatment effect in the West of Scotland Coronary Prevention Study. *Circulation*.

Frees, Edward W. 2004. *Longitudinal and panel data: analysis and applications in the social sciences*. Cambridge University Press.

Friedman, Jerome, Trevor Hastie, and Rob Tibshirani. 2009. glmnet: Lasso and elastic-net regularized generalized linear models. *R package version* 1(4).

———. 2010. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* 33(1):1.

Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. 2001. *The elements of statistical learning*, vol. 1. Springer series in statistics Springer, Berlin.

———. 2008. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* 9(3): 432–441.

Geng*, Sinong, Zhaobin Kuang*, Jie Liu, Stephen Wright, and David Page. 2018a. Stochastic learning for sparse discrete Markov random fields with controlled gradient approximation error. In *Proceedings of the thirty-fourth conference on uncertainty in artificial intelligence (2018)*.

———. 2018b. Stochastic learning for sparse discrete Markov random fields with controlled gradient approximation error. In Geng* et al. (2018a).

Geng, Sinong, Zhaobin Kuang, and David Page. 2017. An efficient pseudo-likelihood method for sparse binary pairwise Markov network estimation. *arXiv Preprint arXiv:1702.08320*.

Geng*, Sinong, Zhaobin Kuang*, Peggy Peissig, and David Page. 2018c. Temporal poisson square root graphical models. In *International conference on machine learning*.

———. 2018d. Temporal poisson square root graphical models. In Geng* et al. (2018c).

Ghaoui, Laurent El, Vivian Viallon, and Tarek Rabbani. 2010. Safe feature elimination for the lasso and sparse supervised learning problems. *arXiv Preprint*.

Girardin, E, and D Raccah. 1998. Interaction between converting enzyme inhibitors and hypoglycemic sulfonamides or insulin. *Presse Medicale (Paris, France: 1983)* 27(37):1914–1923.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. MIT Press.

Granger, Clive WJ. 1969. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: Journal of the Econometric Society* 424–438.

Gunawardana, Asela, Christopher Meek, and Puyang Xu. 2011. A model for temporal dependencies in event streams. In *Advances in neural information processing systems*, 1962–1970.

Hall, Eric C, Garvesh Raskutti, and Rebecca Willett. 2016. Inference of high-dimensional autoregressive generalized linear models. *arXiv Preprint arXiv:1605.02693*.

Hall, Eric C, and Rebecca M Willett. 2013. Dynamical models and tracking regret in online convex programming. In *Proceedings of the 30th international conference on international conference on machine learning*.

Han, Jiawei, Jian Pei, and Micheline Kamber. 2011. *Data mining: concepts and techniques*. Elsevier.

Harpaz, Rave, William DuMochel, and Nigam H Shah. 2015. Big data and adverse drug reaction detection. *Clinical Pharmacology & Therapeutics*.

Harpaz, Rave, William DuMouchel, Nigam H Shah, David Madigan, Patrick Ryan, and Carol Friedman. 2012. Novel data-mining methodologies for adverse drug event discovery and analysis. *Clinical Pharmacology & Therapeutics*.

Hastie, Trevor, Robert Tibshirani, and Martin Wainwright. 2015. *Statistical learning with sparsity: the lasso and generalizations*. CRC Press.

Hawkes, Alan G. 1971a. Point spectra of some mutually-exciting point processes. *Journal of the Royal Statistical Society. Series B (Methodological)* 33(3):438–443.

———. 1971b. Point spectra of some self-exciting and mutually-exciting point processes. *Journal of the Royal Statistical Society. Series B (Methodological)* 58:83–90.

Heckerman, David. 2018. Accounting for hidden common causes when infering cause and effect from observational data. *arXiv preprint arXiv:1801.00727*.

Heckerman, David, Deepti Gurdasani, Carl Kadie, Cristina Pomilla, Tommy Carstensen, Hilary Martin, Kenneth Ekoru, Rebecca N Nsubuga, Gerald Ssenyomo, Anatoli Kamali, et al. 2016. Linear mixed model for heritability estimation that explicitly addresses environmental variation. *Proceedings of the National Academy of Sciences* 113(27):7377–7382.

Hernan, Miguel, and James Robins. 2018. *Causal inference*. Boca Raton: Chapman & Hall/CRC.

Höfling, Holger, and Robert Tibshirani. 2009. Estimation of sparse binary pairwise Markov networks using pseudo-likelihoods. *Journal of Machine Learning Research* 10(Apr):883–906.

Honorio, Jean. 2012a. Convergence rates of biased stochastic optimization for learning sparse Ising models. In *Proceedings of the 29th international conference on machine learning (icml-12)*, ed. John Langford and Joelle Pineau, 257–264. ICML '12, New York, NY, USA: Omnipress.

———. 2012b. Lipschitz parametrization of probabilistic graphical models. *arXiv Preprint arXiv:1202.3733*.

Honorio, Jean, and Dimitris Samaras. 2010. Multi-task learning of Gaussian graphical models. In *Proceedings of the 27th international conference on machine learning (icml-10)*, 447–454.

Hripcsak, George, Jon D Duke, Nigam H Shah, Christian G Reich, Vojtech Huser, Martijn J Schuemie, Marc A Suchard, Rae Woong Park, Ian Chi Kei Wong, Peter R Rijnbeek, and Others. 2015. Observational health data sciences and informatics (OHDSI): Opportunities for observational researchers. *Studies in Health Technology and Informatics*.

Hsieh, Cho-Jui, Mátyás A Sustik, Inderjit S Dhillon, Pradeep K Ravikumar, and Russell Poldrack. 2013. BIG & QUIC: Sparse inverse covariance estimation for a million variables. In *Advances in neural information processing systems*, 3165–3173.

Hurle, M R, L Yang, Q Xie, D K Rajpal, P Sanseau, and P Agarwal. 2013. Computational drug repositioning: from data to therapeutics. *Clinical Pharmacology & Therapeutics* 93(4).

Imbens, Guido W, and Donald B Rubin. 2015. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

IMEDS. 2016. Innovation in medical evidence development and surveillance (IMEDS). `http://imeds.reaganudall.org/`.

Inouye, David, Pradeep Ravikumar, and Inderjit Dhillon. 2016. Square root graphical models: multivariate generalizations of univariate exponential families that permit positive dependencies. In *International conference on machine learning*, 2445–2453.

Inouye, David I, Pradeep K Ravikumar, and Inderjit S Dhillon. 2015. Fixed-length Poisson {MRF}: adding dependencies to the multinomial. In *Advances in neural information processing systems*, 3213–3221.

Inukai, T, Y Inukai, R Matsutomo, K Okumura, K Takanashi, K Takebayashi, K Tayama, Y Aso, and Y Takemura. 2004. Clinical usefulness of doxazosin in patients with type 2 diabetes complicated by hypertension: effects on glucose and lipid metabolism. *The Journal of International Medical Research*.

Jenkins, David J A, Cyril W C Kendall, Maryam Hamidi, Edward Vidgen, Dorothea Faulkner, Tina Parker, Nalini Irani, Thomas M S Wolever, Ignatius Fong, Peter Kopplin, and Others. 2005. Effect of antibiotics as cholesterol-lowering agents. *Metabolism*.

Johnson, Nicholas A. 2013. A dynamic programming algorithm for the fused lasso and l0-segmentation. *Journal of Computational and Graphical Statistics* 22(2):246–260.

Kadie, Carl M, and David Heckerman. 2017. Ludicrous speed linear mixed models for genome-wide association studies. *bioRxiv* 154682.

Kalisch, Markus, and Peter Bühlmann. 2007. Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *Journal of Machine Learning Research* 8(Mar):613–636.

Karger, David, and Nathan Srebro. 2001. Learning Markov networks: Maximum bounded tree-width graphs. In *Proceedings of the twelfth annual acm-siam symposium on discrete algorithms*, 392–401. Society for Industrial and Applied Mathematics.

Kesäniemi, Y A, and Scott M Grundy. 1984. Turnover of low density lipoproteins during inhibition of cholesterol absorption by neomycin. *Arteriosclerosis, Thrombosis, and Vascular Biology*.

Kesim, Murat, Ahmet Tiryaki, Mine Kadioglu, Efnan Muci, Nuri Ihsan Kalyoncu, and Ersin Yaris. 2011. The effects of sertraline on blood lipids, glucose, insulin and HBA1C levels: A prospective clinical trial on depressive patients. *Journal of Research in Medical Sciences: the Official Journal of Isfahan University of Medical Sciences*.

Knox, Craig, Vivian Law, Timothy Jewison, Philip Liu, Son Ly, Alex Frolkis, Allison Pon, Kelly Banco, Christine Mak, Vanessa Neveu, and Others. 2011. DrugBank 3.0: a comprehensive resource for âĽ˜omics' research on drugs. *Nucleic Acids Research* 39(suppl 1):D1035—-D1041.

Kodama, Junichi, Shigehiro Katayama, Kiyoshi Tanaka, Akira Itabashi, Shyoji Kawazu, and Jun Ishii. 1990. Effect of captopril on glucose concentration: possible role of augmented postprandial forearm blood flow. *Diabetes Care*.

Kolar, Mladen, Le Song, Amr Ahmed, and Eric P Xing. 2010. Estimating time-varying networks. *The Annals of Applied Statistics* 94–123.

Koller, Daphne, and Nir Friedman. 2009. *Probabilistic graphical models: principles and techniques*. MIT Press.

Kuang, Zhaobin, Sinong Geng, and David Page. 2017a. A screening rule for l1-regularized ising model estimation. In *Advances in neural information processing systems*, 720–731.

———. 2017b. A screening rule for l1-regularized ising model estimation. In Kuang et al. (2017a), 720–731.

Kuang, Zhaobin, Peggy Peissig, Vitor Santos Costa, Richard Maclin, and David Page. 2017c. Pharmacovigilance via baseline regularization with large-scale longitudinal observational data. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, 1537–1546. ACM.

———. 2017d. Pharmacovigilance via baseline regularization with large-scale longitudinal observational data. In Kuang et al. (2017c), 1537–1546.

Kuang, Zhaobin, James Thomson, Michael Caldwell, Peggy Peissig, Ron Stewart, and David Page. 2016a. Baseline regularization for computational drug repositioning with longitudinal observational data. In *Ijcai: Proceedings of the conference*, vol. 2016, 2521. NIH Public Access.

———. 2016b. Baseline regularization for computational drug repositioning with longitudinal observational data. In Kuang et al. (2016a), 2521.

———. 2016c. Computational drug repositioning using continuous self-controlled case series. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 491–500. ACM.

———. 2016d. Computational drug repositioning using continuous self-controlled case series. In Kuang et al. (2016c), 491–500.

Kuang, Zhaobin, Bao Yujia, James Thomson, Michael Caldwell, Peggy Pessig, Ron Stewart, Rebecca Willett, and David Page. 2018a. A machine-learning based drug repurposing approach using baseline regularization. In *In silico methods for drug repurposing: Methods and protocols*, ed. Quentin Vanhaelen. Springer.

———. 2018b. A machine-learning based drug repurposing approach using baseline regularization. In Kuang et al. (2018a).

Kuhn, Michael, Monica Campillos, Ivica Letunic, Lars Juhl Jensen, and Peer Bork. 2010. A side effect resource to capture phenotypic effects of drugs. *Molecular Systems Biology* 6(1):343.

Lamb, Justin. 2007. The connectivity map: a new tool for biomedical research. *Nature Reviews Cancer*.

Lee, Seunghak, Nico Gornitz, Eric P Xing, David Heckerman, and Christoph Lippert. 2017. Ensembles of lasso screening rules. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Lee, Su-In, Varun Ganapathi, and Daphne Koller. 2006. Efficient structure learning of Markov networks using l1-regularization. In *Proceedings of the 19th international conference on neural information processing systems*, 817–824. MIT Press.

Levin, David Asher, Yuval Peres, and Elizabeth Lee Wilmer. 2009. *Markov chains and mixing times*. American Mathematical Society.

Li, Jiao, Si Zheng, Bin Chen, Atul J Butte, S Joshua Swamidass, and Zhiyong Lu. 2015. A survey of current trends in computational drug repositioning. *Briefings in Bioinformatics*.

———. 2016. A survey of current trends in computational drug repositioning. *Briefings in Bioinformatics* 17(1):2–12.

Linderman, Scott, and Ryan Adams. 2014. Discovering latent network structure in point process data. In *International conference on machine learning*, 1413–1421.

Lippert, Christoph, Jennifer Listgarten, Ying Liu, Carl M Kadie, Robert I Davidson, and David Heckerman. 2011. Fast linear mixed models for genome-wide association studies. *Nature methods* 8(10):833.

Liu, Han, Kathryn Roeder, and Larry Wasserman. 2010a. Stability approach to regularization selection (stars) for high dimensional graphical models. In *Advances in neural information processing systems*, 1432–1440.

Liu, Jie. 2014. Statistical methods for genome-wide association studies and personalized medicine. Ph.D. thesis, The University of Wisconsin-Madison.

Liu, Jie, and David Page. 2013a. Bayesian estimation of latently-grouped parameters in undirected graphical models. In *Advances in neural information processing systems*, 1232–1240.

———. 2013b. Structure learning of undirected graphical models with contrastive divergence. *ICML 2013 Workshop on Structured Learning: Inferring Graphs from Structured and Unstructured Inputs*.

Liu, Jie, David Page, Houssam Nassif, Jude Shavlik, Peggy Peissig, Catherine McCarty, Adedayo A Onitilo, and Elizabeth Burnside. 2013a. Genetic variants improve breast cancer risk prediction on mammograms. In *Amia annual symposium proceedings*, vol. 2013, 876. American Medical Informatics Association.

Liu, Jie, David Page, Peggy Peissig, Catherine McCarty, Adedayo A Onitilo, Amy Trentham-Dietz, and Elizabeth Burnside. 2014a. New genetic variants improve personalized breast cancer diagnosis. *AMIA Summits on Translational Science Proceedings* 2014:83.

Liu, Jie, Peggy Peissig, Chunming Zhang, Elizabeth Burnside, Catherine McCarty, and David Page. 2012. Graphical-model based multiple testing under dependence, with applications to genome-wide association studies. In *Uncertainty in artificial intelligence*, vol. 2012, 511. NIH Public Access.

Liu, Jie, Yirong Wu, Irene Ong, David Page, Peggy Peissig, Catherine McCarty, Adedayo A Onitilo, and Elizabeth Burnside. 2015. Leveraging interaction between genetic variants and mammographic findings for personalized breast cancer diagnosis. *AMIA Summits on Translational Science Proceedings* 2015:107.

Liu, Jie, Chunming Zhang, Elizabeth Burnside, and David Page. 2014b. Learning heterogeneous hidden {M}arkov random fields. In *Artificial intelligence and statistics*, 576–584.

———. 2014c. Multiple testing under dependence via semiparametric graphical models. In *Proceedings of the 31st international conference on machine learning (icml-14)*, 955–963.

Liu, Jie, Chunming Zhang, David Page, and Others. 2016. Multiple testing under dependence via graphical models. *The Annals of Applied Statistics* 10(3):1699–1724.

Liu, Jun, Zheng Zhao, Jie Wang, and Jieping Ye. 2013b. Safe screening with variational inequalities and its application to lasso. *arXiv Preprint arXiv:1307.7577*.

Liu, Xianghang, and Justin Domke. 2014. Projecting Markov random field parameters for fast mixing. In *Advances in neural information processing systems*, 1377–1385.

Liu, Yanbin, Bin Hu, Chengxin Fu, and Xin Chen. 2010b. DCDB: drug combination database. *Bioinformatics*.

Loh, Po-Ling, Martin J Wainwright, and Others. 2012. Structure estimation for discrete graphical models: generalized covariance matrices and their inverses. In *Advances in neural information processing systems*, 2096–2104.

———. 2013. Structure estimation for discrete graphical models: generalized covariance matrices and their inverses. *The Annals of Statistics* 41(6):3022–3049.

Luo, Shikai, Rui Song, and Daniela Witten. 2014. Sure screening for Gaussian graphical models. *arXiv Preprint arXiv:1407.7819*.

vinh quoc Luong, Khanh, and Lan Thi Hoang Nguyen. 2012. The impact of thiamine treatment in the diabetes mellitus. *Journal of Clinical Medicine Research* 4(3):153.

Lustman, Patrick J, Monique M Williams, Gregory S Sayuk, Billy D Nix, and Ray E Clouse. 2007. Factors influencing glycemic control in type 2 diabetes during acute-and maintenance-phase treatment of major depressive disorder with bupropion. *Diabetes Care* 30(3):459–466.

Madigan, David, Nandini Raghavan, William Dumouchel, Martha Nason, Christian Posse, and Greg Ridgeway. 2002. Likelihood-based data squashing: A modeling approach to instance construction. *Data Mining and Knowledge Discovery*.

Madigan, David, Martijn J Schuemie, and Patrick B Ryan. 2013. Empirical performance of the case–control method: Lessons for developing a risk identification and analysis system. *Drug Safety*.

Mallat, Stephane. 2008. *A wavelet tour of signal processing: the sparse way*. Academic {Press}.

Maurer, Andreas, and Massimiliano Pontil. 2009. Empirical Bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*.

Mazumder, Rahul, and Trevor Hastie. 2012. Exact covariance thresholding into connected components for large-scale graphical lasso. *Journal of Machine Learning Research* 13(Mar):781–794.

Miasojedow, Błażej, and Wojciech Rejchel. 2016. Sparse estimation in Ising model via penalized Monte Carlo methods. *arXiv Preprint arXiv:1612.07497*.

Mitchell, Tom M. 1997. *Machine learning*. 1st ed. MGH.

Mitliagkas, Ioannis, and Lester Mackey. 2017. Improving Gibbs Sampler Scan Quality with DoGS.

Montastruc, Jean-Louis, Agnès Sommet, Haleh Bagheri, and Maryse Lapeyre-Mestre. 2011. Benefits and strengths of the disproportionality analysis for identification of adverse drug reactions in a pharmacovigilance database. *British Journal of Clinical Pharmacology*.

Muggeo, Vito M R. 2003. Estimating regression models with unknown break points. *Statistics in Medicine*.

Murphy, Kevin P. 2012. *Machine learning: a probabilistic perspective*. MIT Press.

Nadkarni, Prakash M. 2010. Drug safety surveillance using de-identified EMR and claims data: issues and challenges. *Journal of the American Medical Informatics Association: JAMIA* 17(6):671.

Ndiaye, Eugene, Olivier Fercoq, Alexandre Gramfort, and Joseph Salmon. 2015. GAP safe screening rules for sparse multi-task and multi-class models. In *Advances in neural information processing systems*, 811–819.

Neerati, Prasad, and Jyothsna Gade. 2011. Influence of atorvastatin on the pharmacokinetics and pharmacodynamics of glyburide in normal and diabetic rats. *European Journal of Pharmaceutical Sciences* 42(3):285–289.

Negahban, Sahand, Bin Yu, Martin J Wainwright, and Pradeep K Ravikumar. 2009. A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. In *Advances in neural information processing systems*, 1348–1356.

Nesterov, Yu. 2012. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*.

Norén, G Niklas, Tomas Bergvall, Patrick B Ryan, Kristina Juhlin, Martijn J Schuemie, and David Madigan. 2013. Empirical performance of the calibrated self-controlled cohort analysis within temporal pattern discovery: Lessons for developing a risk identification and analysis system. *Drug Safety*.

Ogarrio, Juan Miguel, Peter Spirtes, and Joe Ramsey. 2016. A hybrid causal search algorithm for latent variable models. In *Conference on probabilistic graphical models*, 368–379.

OHDIS. 2016. Frequently asked questions - OHDSI. `http://www.ohdsi.org/who-we-are/frequently-asked-questions/`.

OMOP. 2015. Ground truth for monitoring health outcomes of interest. `http://omop.org/sites/default/files/ground%20truth.pdf`.

———. 2016a. Archived research- OMOP. `http://omop.org/ResearchArchive`.

———. 2016b. Ground truth for monitoring health outcomes of interest. `http://omop.org/sites/default/files/ground%20truth.pdf`.

———. 2016c. Observational medical outcomes partnership. `http://omop.org/`.

Ortega, James M, and Werner C Rheinboldt. 2000. *Iterative solution of nonlinear equations in several variables*. SIAM.

Page, G L J, David Laight, and M H Cummings. 2011. Thiamine deficiency in diabetes mellitus and the impact of thiamine replacement on glucose metabolism and vascular disease. *International Journal of Clinical Practice* 65(6):684–690.

Parikh, Neal, Stephen Boyd, and Others. 2014. Proximal algorithms. *Foundations and Trends in Optimization* 1(3):127–239.

Pearl, Judea. 2009. *Causality*. Cambridge university press.

Pena, Javier, and Ryan Tibshirani. 2016. Lecture notes in machine learning 10-725/statistics 36-725-convex optimization (fall 2016).

Peng, Jie, Pei Wang, Nengfeng Zhou, and Ji Zhu. 2009. Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association* 104(486):735–746.

Pillow, Jonathan W, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, E J Chichilnisky, and Eero P Simoncelli. 2008. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature* 454:995–999.

Poudel, Resham Raj, and Nisha Kusum Kafle. 2017. Verapamil in diabetes. *Indian Journal of Endocrinology and Metabolism* 21(5):788.

Powell, Valerie, Franklin M Din, Amit Acharya, and Miguel Humberto Torres-Urquidy. 2012. *Integration of medical and dental care and patient data*. Springer Science & Business Media.

Raman, P G. 2016. Hypoglycemia induced by pregabalin. *Journal of The Association of Physicians of India* 64.

Ramdas, Aaditya, and Ryan J Tibshirani. 2015. Fast and flexible ADMM algorithms for trend filtering. *Journal of Computational and Graphical Statistics*.

Ravikumar, Pradeep, Han Liu, John Lafferty, and Larry Wasserman. 2007. Spam: Sparse additive models. In *Proceedings of the 20th international conference on neural information processing systems*, 1201–1208. Curran Associates Inc.

Ravikumar, Pradeep, Martin J Wainwright, John D Lafferty, and Others. 2010. High-dimensional Ising model selection using l1-regularized logistic regression. *The Annals of Statistics* 38(3):1287–1319.

Reisinger, Stephanie J, Patrick B Ryan, Donald J O'Hara, Gregory E Powell, Jeffery L Painter, Edward N Pattishall, and Jonathan A Morris. 2010. Development and evaluation of a common data model enabling active drug safety surveillance using disparate healthcare databases. *Journal of the American Medical Informatics Association*.

Robb, Melissa A, Judith A Racoosin, Rachel E Sherman, Thomas P Gross, Robert Ball, Marsha E Reichman, Karen Midthun, and Janet Woodcock. 2012. The US Food and Drug Administration's Sentinel Initiative: expanding the horizons of medical product safety. *Pharmacoepidemiology and Drug Safety*.

Ryan, P. 2015. Establishing a drug era persistence window for active surveillance. Foundation for the National Institutes of Health, 2010.

Ryan, P B. 2010. Establishing a drug era persistence window for active surveillance. *White Papers*.

Ryan, Patrick B, David Madigan, Paul E Stang, J Marc Overhage, Judith A Racoosin, and Abraham G Hartzema. 2012. Empirical assessment of methods for risk identification in healthcare data: Results from the experiments of the observational medical outcomes partnership. *Statistics in Medicine*.

Ryan, Patrick B, Martijn J Schuemie, Susan Gruber, Ivan Zorych, and David Madigan. 2013a. Empirical performance of a new user cohort method: Lessons for developing a risk identification and analysis system. *Drug Safety*.

Ryan, Patrick B, Martijn J Schuemie, and David Madigan. 2013b. Empirical performance of a self-controlled cohort method: lessons for developing a risk identification and analysis system. *Drug Safety* 36(1):95–106.

Samuel, Paul. 1979. Treatment of hypercholesterolemia with neomycin-a time for reappraisal. *New England Journal of Medicine*.

Satoh, Tetsuo, Shuichi Hara, Midori Takashima, and Haruo Kitagawa. 1980. Hyperglycemic effect of hydralazine in rats. *Journal of Pharmacobio-Dynamics*.

Schmidt, Mark, Nicolas L Roux, and Francis R Bach. 2011. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in neural information processing systems*, 1458–1466.

Scholl, Joep H G, Rike van Eekeren, and Eugène P van Puijenbroek. 2015. Six cases of (severe) hypoglycaemia associated with gabapentin use in both diabetic and non-diabetic patients. *British Journal of Clinical Pharmacology*.

Schuemie, Martijn J, David Madigan, and Patrick B Ryan. 2013. Empirical performance of LGPS and LEOPARD: Lessons for developing a risk identification and analysis system. *Drug Safety*.

Schuemie, Martijn J, Gianluca Trifirò, Preciosa M Coloma, Patrick B Ryan, and David Madigan. 2016. Detecting adverse drug reactions following long-term exposure in longitudinal observational data: The exposure-adjusted self-controlled case series. *Statistical Methods in Medical Research* 25(6): 2577–2592.

Simpson, Shawn E. 2011. Self-controlled methods for postmarketing drug safety surveillance in large-scale longitudinal data. Dissertation, Columbia University.

Simpson, Shawn E, David Madigan, Ivan Zorych, Martijn J Schuemie, Patrick B Ryan, and Marc A Suchard. 2013. Multiple self-controlled case series for large-scale longitudinal observational databases. *Biometrics* 69(4):893–902.

Spirtes, Peter, Clark N Glymour, Richard Scheines, David Heckerman, Christopher Meek, Gregory Cooper, and Thomas Richardson. 2000. *Causation, prediction, and search*. MIT press.

Sra, Suvrit, Sebastian Nowozin, and Stephen J Wright. 2012. *Optimization for machine learning*. MIT Press.

Suchard, Marc A, Shawn E Simpson, Ivan Zorych, Patrick Ryan, and David Madigan. 2013a. Massive parallelization of serial inference algorithms for a complex generalized linear model. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*.

Suchard, Marc A, Ivan Zorych, Shawn E Simpson, Martijn J Schuemie, Patrick B Ryan, and David Madigan. 2013b. Empirical performance of the self-controlled case series design: Lessons for developing a risk identification and analysis system. *Drug Safety*.

Sultana, Janet, Paola Cutroneo, Gianluca Trifiro, and Others. 2013. Clinical and economic burden of adverse drug reactions. *Journal of Pharmacology and Pharmacotherapeutics* 4(5):73.

Tatonetti, Nicholas P, P Ye Patrick, Roxana Daneshjou, and Russ B Altman. 2012. Data-driven prediction of drug effects and interactions. *Science Translational Medicine* 4(125):125ra31—-125ra31.

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

Tibshirani, Robert, Jacob Bien, Jerome Friedman, Trevor Hastie, Noah Simon, Jonathan Taylor, and Ryan J Tibshirani. 2012. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 74(2):245–266.

Tibshirani, Robert, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. 2005. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*.

Tibshirani, Ryan J, and Jonathan Taylor. 2011. The solution path of the generalized lasso. *The Annals of Statistics* 1335–1371.

Ting, Rose Zhao-Wei, Cheuk Chun Szeto, Michael Ho-Ming Chan, Kwok Kuen Ma, and Kai Ming Chow. 2006. Risk factors of vitamin B12 deficiency in patients receiving metformin. *Archives of Internal Medicine*.

Tseng, Paul. 2001. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications* 109(3):475–494.

Tukey, John W. 1977. *Exploratory data analysis*, vol. 2. Reading, Mass.

Uhlén, Mathias, Linn Fagerberg, Björn M Hallström, Cecilia Lindskog, Per Oksvold, Adil Mardinoglu, Åsa Sivertsson, Caroline Kampf, Evelina Sjöstedt, Anna Asplund, and Others. 2015. Tissue-based map of the human proteome. *Science* 347(6220):1260419.

Vandenberghe, Lieven. 2016. Lecture notes in EE236C-optimization methods for large-scale systems (Spring 2016).

Vazquez, Jose A, and Jack D Sobel. 1995. Fungal infections in diabetes. *Infectious Disease Clinics of North America*.

Vermes, Emmanuelle, Anique Ducharme, Martial G Bourassa, Myriam Lessard, Michel White, and Jean-Claude Tardif. 2003. Enalapril reduces the incidence of diabetes in patients with chronic heart failure insight from the studies of left ventricular dysfunction (SOLVD). *Circulation*.

Viallon, Vivian, Onureena Banerjee, Eric Jougla, Grégoire Rey, and Joel Coste. 2014. Empirical comparison study of approximate methods for structure selection in binary graphical models. *Biometrical Journal* 56(2):307–331.

Vuffray, Marc, Sidhant Misra, Andrey Lokhov, and Michael Chertkov. 2016. Interaction screening: efficient and sample-optimal learning of Ising models. In *Advances in neural information processing systems*, 2595–2603.

Wainwright, Martin J. 2009a. Sharp thresholds for high-dimensional and noisy recovery of sparsity using l1-constrained quadratic programming. *IEEE Transactions on Information Theory*.

———. 2009b. Sharp thresholds for high-dimensional and noisy sparsity recovery using l1-constrained quadratic programming (lasso). *IEEE Transactions on Information Theory* 55(5):2183–2202.

Wainwright, Martin J, Michael I Jordan, and Others. 2008. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning* 1(1–2):1–305.

Wainwright, Martin J, John D Lafferty, and Pradeep K Ravikumar. 2006. High-dimensional Graphical model selection using l1-regularized logistic regression. In *Advances in neural information processing systems*, 1465–1472.

Wainwright, Martin J, Pradeep Ravikumar, and John D Lafferty. 2007. High-dimensional graphical model selection using l1-regularized logistic regression. *Advances in Neural Information Processing Systems* 19:1465.

Wan, Ying-Wooi, Genevera I Allen, Yulia Baker, Eunho Yang, Pradeep Ravikumar, Matthew Anderson, and Zhandong Liu. 2016. XMRF: an R package to fit Markov networks to high-throughput genetics data. *BMC Systems Biology* 10(3):69.

Wan, Ying-Wooi, Genevera I Allen, and Zhandong Liu. 2015. TCGA2STAT: simple TCGA data access for integrated statistical analysis in R. *Bioinformatics* btv677.

Wang, Jie, Wei Fan, and Jieping Ye. 2015. Fused lasso screening rules via the monotonicity of subdifferentials. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37(9):1806–1820.

Wang, Jie, Jiayu Zhou, Jun Liu, Peter Wonka, and Jieping Ye. 2014. A safe screening rule for sparse logistic regression. In *Advances in neural information processing systems*, 1053–1061.

Wang, Jie, Jiayu Zhou, Peter Wonka, and Jieping Ye. 2013. Lasso screening rules via dual polytope projection. In *Advances in neural information processing systems*, 1070–1078.

Weinstein, John N, Eric A Collisson, Gordon B Mills, Kenna R Mills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M Stuart, Cancer Genome Atlas Research Network, and Others. 2013. The cancer genome atlas pan-cancer analysis project. *Nature Genetics* 45(10): 1113–1120.

Weiss, Jeremy, Sriraam Natarajan, and David Page. 2012. Multiplicative forests for continuous-time processes. In *Advances in neural information processing systems*, 458–466.

Weiss, Jeremy C, and David Page. 2013. Forest-based point process for event prediction from electronic health records. In *Joint european conference on machine learning and knowledge discovery in databases*, 547–562. Springer.

Weissbrod, Omer, Christoph Lippert, Dan Geiger, and David Heckerman. 2015. Accurate liability estimation improves power in ascertained case-control studies. *Nature methods* 12(4):332.

Widmer, Christian, Christoph Lippert, Omer Weissbrod, Nicolo Fusi, Carl Kadie, Robert Davidson, Jennifer Listgarten, and David Heckerman. 2014. Further improvements to linear mixed models for genome-wide association studies. *Scientific reports* 4:6874.

Wikipedia. 2017. Jay Rockefeller— Wikipedia.

Witten, Daniela M, Jerome H Friedman, and Noah Simon. 2011. New insights and faster computations for the graphical lasso. *Journal of Computational and Graphical Statistics* 20(4):892–900.

Wright, Stephen J. 2015. Coordinate descent algorithms. *Mathematical Programming*.

Wright, Stephen J, Robert D Nowak, and Mário A T Figueiredo. 2009. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing* 57(7):2479–2493.

Xiang, Zhen James, Yun Wang, and Peter J Ramadge. 2016. Screening tests for lasso problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP(99):1.

Xu, Hongteng, Mehrdad Farajtabar, and Hongyuan Zha. 2016. Learning granger causality for hawkes processes. In *International conference on machine learning*, 1717–1726.

Xu, Hua, Melinda C Aldrich, Qingxia Chen, Hongfang Liu, Neeraja B Peterson, Qi Dai, Mia Levy, Anushi Shah, Xue Han, Xiaoyang Ruan, and Others. 2014. Validating drug repurposing signals using electronic health records: a case study of metformin associated with reduced cancer mortality. *Journal of the American Medical Informatics Association* amiajnl—-2014.

Xu, Stanley, Chan Zeng, Sophia Newcomer, Jennifer Nelson, and Jason Glanz. 2012. Use of fixed effects models to analyze self-controlled case series data in vaccine safety studies. *Journal of Biometrics & Biostatistics*.

Yang, Eunho, and Pradeep Ravikumar. 2011. On the use of variational inference for learning discrete graphical model. In *International conference on machine learning*, 1009–1016.

Yang, Eunho, Pradeep Ravikumar, Genevera I Allen, and Zhandong Liu. 2015a. Graphical models via univariate exponential family distributions. *Journal of Machine Learning Research* 16(1):3813–3847.

Yang, Eunho, Pradeep K Ravikumar, Genevera I Allen, and Zhandong Liu. 2013. On Poisson graphical models. In *Advances in neural information processing systems*, 1718–1726.

Yang, Sen, Zhaosong Lu, Xiaotong Shen, Peter Wonka, and Jieping Ye. 2015b. Fused multiple graphical lasso. *SIAM Journal on Optimization* 25(2):916–943.

Zhao, Tuo, Mo Yu, Yiming Wang, Raman Arora, and Han Liu. 2014. Accelerated mini-batch randomized block coordinate descent method. In *Advances in neural information processing systems*.

Zhu, F, and D Wang. 2011. Estimation and testing for a Poisson autoregressive model. *Metrika* 73(2):211–230.