Large-Scale Directed Graphical Models Learning

by

Gunwoong Park

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

Doctor of Philosophy
(Statistics)

at the
University of Wisconsin–Madison
2016

Date of final oral examination: July 25, 2016

The dissertation is approved by the following members of the Final Oral Committee:

Professor Garvesh Raskutti, Department of Statistics and Department of Computer Science
Professor Sunduz Keles, Department of Biostatistics & Medical Informatics and Department of Statistics
Professor Karl Rohe, Department of Statistics
Professor Bret Hanlon, Department of Statistics
Professor Rebecca Willet Department of Electrical and Computer Engineering

Abstract

Directed graphical models are a powerful statistical method to compactly describe directional or causal relationships among the set of variables in large-scale data. However, a number of statistical and computational challenges arise that make learning directed graphical models often impossible for large-scale data. These issues include: (1) model identifiability; (2) computational guarantee; (3) sample size guarantee; and (4) combining interventional experiments with observational data.

In this thesis, we focus on learning directed graphical models by addressing the above four issues. In Chapter 3, we discuss learning Poisson DAG models for modeling large-scale multivariate count data problems where each node is a Poisson random variable conditioning on its parents. We address the question of (1) model identifiability and learning algorithms with (2) computational complexity and (3) sample complexity. We prove that Poisson DAG models are fully identifiable from observational data using the notion of *overdispersion*, and present a polynomial-time algorithm that learns the Poisson DAG model under suitable regularity conditions.

Chapter 4 focuses on learning a broader class of DAG models in large-scale settings. We address the issue of (1) model identifiability and learning algorithms with (2) computational complexity and (3) sample complexity. We introduce a new class of identifiable DAG models which include many interesting classes of distributions such as Poisson, Binomial, Geometric, Exponential, Gamma, and many more, and prove that this class of DAG models is fully identifiable using the idea of *overdispersion*. Furthermore, we develop statistically consistent and computationally tractable learning algorithms for the new class of identifiable DAG models in high-dimensional settings. Our algorithms exploits the sparsity of the graphs and overdispersion property.

Chapter 5 concerns learning general DAG models using a combination of observational and interventional (or experimental) data. Prior work has focused on algorithms using Markov equivalence class (MEC) for the DAG and then using do-calculus rules

based on interventions to learn the additional directions. However it has been shown that existing passive and active learning strategies that rely on accurate recovery of the MEC do not scale well to large-scale graphs because recovering MEC for DAG models are not successful large-scale graphs. Hence, we prove (1) model identifiability using the notion of the *moralized* graphs, and develop passive and active learning algorithms (4) combining interventional experiments with observational data.

Lastly in Chapter 6, we concern learning directed cyclic graphical (DCG) models. We focus on (1) model identifiability for directed graphical models with feedback. We provide two new identifiability assumptions with respect to *sparsity* of a graph and the number of *d-separation rules*, and compare these new identifiability assumptions to the widely-held faithfulness and minimality assumptions. Furthermore we develop search algorithms for small-scale DCG models based on our new identifiability assumptions.

Acknowledgments

There are many people who have made this thesis possible.

First, I would like to express my deepest appreciation and gratitude to Professor Garvesh Raskutti, my advisor. He provided endless support, guidance, and research freedom to me, and his extensive knowledge and insight, enthusiasm have been very inspiring throughout my researches. All of the results presented in this thesis are based on projects with Garvesh, hence the use of 'we' throughout the thesis.

I would also like to thank Professor Sunduz Keles, the second member of my committee. Sunduz has always been extremely generous with her time. I had a number of interesting discussions with Sunduz on the analysis of gene sequence data. These discussions led to significant improvements in understanding real-world problem and helped sharpen my thoughts about the data analysis. I would like to thank Professor Karl Rohe, Professor Bret Hanlon, Professor Ming Yuan, Professor Ming Yuan, Professor Menggang Yu and Professor Rebecca Willet for suggesting interesting research directions and providing invaluable guidance.

Finally, I must thank to all of my family and friends for their constant support and encouragement. They have faithfully sustained and supported me through innumerable challenges. I dedicate this thesis to my family.

Contents

Chapter 1

Introduction

Analysis and modeling large-scale multivariate data is an important research problem, as massive amounts of data is available in the fields of statistics, machine learning, biology and many of their applications [5, 19, 28, 37]. For example, marketing companies such as Walmart, Target, and Amazon examine large data sets containing a variety of data types to uncover hidden patterns, market trends, customer preferences and other useful business information. Medical researchers are using entire human genome data to discover gene regulatory pathways so as to uncover causes of cancers or genetic diseases. Consequently, there is a huge demand to develop rich classes of statistical models that faithfully represent large-scale data with feasible learning methods.

In many real-world problems, there exist inherent conditional independence (CI) properties or directional relations between variables. CI properties in the underlying probability distribution can be explained by the structure which enables to factor the representation of the distribution into modular component. Hence, many recent works have attempted to adapt existing methods and develop new methods that exploit CI properties in the distribution to compactly and faithfully represent high-dimensional data.

One approach that has received significant attention is the graphical modeling framework. Graphical models provide a language to compactly describe large joint probability distributions using a set of non-directional or directional relation-

ships among neighboring variables in a graph. Graphical models includes a broad class of dependence models for various data types. Broadly speaking, there are two common sets of graphical models: (1) undirected graphical models (also called Markov random fields), (2) directed graphical models; acyclic graphical (DAG) models (also called Bayesian networks), and directed cyclic graphical (DCG) models.

Directed graphical models are a popular class of statistical models that model directional or causal relationships between variables. Such directional relationships naturally arise in many applications including biology, neuroscience, astronomy and others [16, 20, 38]. The presence of directed graph structure enables the compact representation of rich classes of probability models and efficient algorithms for model learning [6, 9, 13, 29, 59, 68, 70]. Moreover, the structure of a directed graphical model can describe which variables have direct influence on other variables in an understandable and visual manner [48, 52, 50, 68]. Therefore learning directed graphical model is roughly speaking equivalent to finding fundamental information about which variables influence each other.

However, a number of statistical and computational challenges arise that make learning directed graphical models often impossible for large-scale datasets, even when variables have a natural causal or directional structure. These issues are: (1) model identifiability; (2) computational guarantee; (3) sample size guarantee; and (4) combining interventional experiments with observational data.

Regarding the (1) model identifiability issue, directed graphical models are often not possible to be inferred or can only be identified up to their Markov equivalent graphs [68] where they represent the same collection of conditional independence properties. Recent works propose that it is possible to fully identify the DAG structure including directions by exploiting characterization of the node probability distribution. For example, Shimizu et al. [64] proved identifiability for linear non-Gaussian structural equation models, and Peters et al. [54] proved identifiability for non-parametric structural equation models with additive independent noise. Peters and Bühlmann [53]

proved identifiability for Gaussian DAG models based on structural equation models with known or the same variance of errors. However the identifiability issue for many DAG models have not yet been extensively studied.

Learning directed graphical models from observational data is an *NP-Hard* problem because it is necessary to search over the space of directed graphs which is super-exponential to the number of variables [8, 10]. Therefore computationally feasible methods for learning directed graphical models are very important. Its difficulty is perhaps best captured in the following quote. "In our view, inferring complete causal models is essentially impossible in large-scale data mining applications with thousands of variables"(Silverstein et al., 2000 [65]).

Many algorithms can recover the directed graphical models up to its Markov equivalent class assuming the faithfulness assumption (see e.g., [12, 59, 60, 68]). However, the faithfulness assumption often require extremely large sample sizes to be satisfied even when the number of nodes is small [73]. Furthermore, many algorithms for learning directed graphical models which do not require the faithfulness assumption are often statistically not consistent to identifying directed graphs or need impractical or restrictive additional assumptions (e.g., [11, 29, 31, 33, 35, 43, 71]).

Lastly experimental interventions that take control of (the distribution of) one or more variables in a system is a popular method to infer causal system or directed graphical models. Roughly speaking, we force one (or more) of the variables into a particular state, and we see how the probability distribution of the other variables is affected. The best scenario is when a set of data are collected where variables we are interested in are intervened. However, we often cannot intervene a lot of variables in a system due to cost, impracticality, ethics, and many reasons. Therefore it is important to uncover the connections between observational and interventional (experimental) data, which enables us to learn directed graphical models much more efficiently.

1.1 Contributions

In this thesis we focus mostly on learning directed graphical models and addressing the above four issues. The main contributions of this thesis are to (1) introduce new identifiability assumptions for broad class of directed graphical models, and (2) develop new algorithms using our new identifiability assumptions, which are able to more accurately learn the true structure of a directed graphical model than state-of-the-art algorithms and are at the same time computationally tractable. In the remainder of the thesis, we provide additional motivation for our new approaches to learning graphical models and prove how our algorithms can identify directed graphical models with significantly fewer errors than existing algorithms. We start by providing a more detailed introduction to directed graphical models and introduce the overview of the algorithm for learning directed graphical models in Chapter 2.

Chapter 3 concerns learning Poisson DAG models for modeling large-scale multivariate count data problems where each node is a Poisson random variable conditioning on its parents in the underlying DAG. We prove that Poisson DAG models are identifiable from observational data, and present a polynomial-time algorithm that learns the Poisson DAG model under suitable regularity conditions. The main idea behind our algorithm is based on *overdispersion*, in that variables that are conditionally Poisson are overdispersed relative to variables that are marginally Poisson. Our algorithms exploits overdispersion along with methods for learning sparse Poisson undirected graphical models for faster computation. We provide both theoretical guarantees and simulation results for both small and large-scale DAGs.

Chapter 4 addresses the problem of learning large-scale or high-dimensional DAG models. First, we introduce a new class of identifiable DAG models which include many interesting classes of distributions such as Poisson, Binomial, Geometric, Exponential, Gamma and many more. We prove that our class of DAG models is fully identifiable using the notion of *overdispersion*. Next, we develop a new theoretically consistent and computationally tractable algorithm for learning large-scale count DAG models

belonging to our class of DAG models. We provide theoretical results and simulations that our algorithm is statistically consistent in the high-dimensional setting provided the degree of the moralized graph is bounded. Furthermore, we provide a different algorithm for special cases of our class of DAG model where each conditional distribution given its parents belongs to *natural exponential family with quadratic variance function* (NEF-QVF) [44]. This algorithm can recover DAG models with continuous variables and is more accurate and faster than the algorithm we initially provide exploiting the characterization of the natural exponential family.

In Chapter 5, we study the problem of learning DAG models using a combination of observational and experimental data. Prior work has focused on algorithms involving first using observational data to learn the Markov equivalence class (MEC) for the DAG and then using do-calculus rules based on interventions to learn additional directions. However it has been shown that for DAG models where the number of nodes is large, errors are often made in determining the MEC. Hence existing passive and active learning strategies that rely on accurate recovery of the MEC does not scale well to large graphs. Therefore we introduce both a passive and an active learning strategy using a combination of learning the *moralized* graph and the do-calculus rules based on interventional graphs. Since there already exists many algorithms for learning large-scale moralized or undirected graphs that are known to be reliable, we show empirically that our passive learning algorithm makes significantly less errors in terms of recovering the true DAG model compared to the state-of-the-art GIES algorithm which relies on accurate recovery of the MEC. We also show empirically that our active learning algorithm has reliable performance in high-dimensional settings.

Lastly, in Chapter 6 we consider learning directed cyclic graphical (DCG) models for multivariate data where there exist directed cycles or feedback. we address the issue of model identifiability for general DCG models satisfying the Markov assumption. In particular, in addition to the faithfulness assumption which has already been introduced for cyclic models, we introduce two new identifiability assumptions, one

based on selecting the model with the fewest edges and the other based on selecting the DCG model that entails the maximum number of d-separation rules. We provide theoretical results comparing these assumptions which show that: (1) selecting models with the largest number of d-separation rules is strictly weaker than the faithfulness assumption; (2) unlike for DAG models, selecting models with the fewest edges does not necessarily result in a milder assumption than the faithfulness assumption. We also provide connections between our two new principles and minimality assumptions. We use our identifiability assumptions to develop search algorithms for small-scale DCG models. Our simulation study supports our theoretical results, showing that the algorithms based on our two new principles generally out-perform algorithms based on the faithfulness assumption in terms of selecting the true skeleton for DCG models.

Chapter 2

Background

In this chapter we provide a brief introduction to directed graphical models, including factorizations of probability distributions, their representations by graphs, and the Markov assumption. We begin with basic concepts of directed graphical models in Section 2.1. We also summarize three approaches to learning directed graphical models in Section 2.2: (1) score-based algorithms; (2) constraint-based algorithms; and (3) hybrid algorithms.

2.1 Directed Graphical Models

A directed graph $G = (V, E)$ consists of a set of vertices V and a set of directed edges E. The *structure* of a directed graph refers to as the collection of edges. Suppose that $V = \{1, 2, \ldots, p\}$ and there exists a random vector $(X_1, X_2, ..., X_p)$ with probability distribution $\mathbb{P}$ over the vertices in G. A directed edge from a vertex j to k is denoted by (j, k) or $j \to k$. The set $\text{pa}(k)$ of *parents* of a vertex k consists of all nodes j such that $(j, k) \in E$. If there is a directed path $j \to \cdots \to k$, then k is called a *descendant* of j and j is an *ancestor* of k. The set $\text{de}(k)$ denotes the set of all descendants of a node k. The *non-descendants* of a node k are $\text{nd}(k) = V \setminus (\{k\} \cup \text{de}(k))$. For a subset $S \subset V$, we define $\text{an}(S)$ to be the set of nodes k that are in S or are ancestors of some nodes in S. Two nodes that are connected by an edge are called *adjacent*. A triple of nodes (j, k, ℓ) is an *unshielded triple* if j and k are adjacent to

ℓ but j and k are not adjacent. An unshielded triple (j, k, ℓ) forms a *v-structure* if $j \to \ell$ and $k \to \ell$. In this case ℓ is called a *collider*. Another important property of DAGs is that there exists a (possibly non-unique) *causal ordering* π^* of a directed graph represents directions of edges such that for every directed edge $(j, k) \in E$, j comes before k in the causal ordering. Without loss of generality, we assume the true causal ordering $\pi^* = (1, 2, \cdots, p)$. Now we discuss probabilistic directed graphical models for multivariate distributions. Let $(X_1, X_2, , X_p)$ be p random variables with joint distribution $f(X_1, X_2, \cdots, X_p)$. A probabilistic DAG model has the following factorization [39]:

$$f(X_1, X_2, \cdots, X_p) = \prod_{j=1}^{p} f_j(X_j \mid X_{\text{pa}(j)}),$$

where $f_j(X_j \mid X_{\text{pa}(j)})$ refers to the conditional distribution of a variable X_j in terms of its set of parents $X_{\text{pa}(j)}$. However for directed graphical models with directed cycles may not have the factorization property. The joint distributions for directed graphical models with directed cycles will be discussed later in Chapter 6.

2.1.1 d-separation

Furthermore, let U be an undirected path between j and k. If every collider on U is in $\text{an}(S)$ and every non-collider on an undirected path U is not in S, an undirected path U from j to k *d-connects* j and k given $S \subset V \setminus \{j, k\}$ and j is *d-connected* to k given S. If a directed graph G has no undirected path U that d-connects j and k given a subset S, then j is *d-separated* from k given S:

Definition 2.1 (d-connection/separation [51, 66])**.** *For vertices $j, k \in V$ and $S \subset V \setminus \{j, k\}$, j is* d-connected *to k given S if and only if there is an undirected path U between j and k, such that*

(1) If there is an edge between a and b on U and an edge between b and c on U, and $b \in S$, then b is a collider between a and c relative to U.

(2) If b is a collider between a and c relative to U, then there is a descendant d of b and $d \in S$

2.1.2 Causal Markov Condition

Let $X_j \perp\!\!\!\perp X_k \mid X_S$ with $S \subset V \setminus \{j,k\}$ denote the conditional independence (CI) statement that X_j is conditionally independent (as determined by $\mathbb{P}$) of X_k given the set of variables $X_S = \{X_\ell \mid \ell \in S\}$, and let $X_j \not\perp\!\!\!\perp X_k \mid X_S$ denote conditional dependence. The *Causal Markov condition* associates CI statements of $\mathbb{P}$ with a directed graph G:

Definition 2.2 (Causal Markov condition (CMC) [68])**.** *A probability distribution $\mathbb{P}$ over a set of vertices V satisfies the* Causal Markov condition *with respect to a (acyclic or cyclic) graph $G = (V, E)$ if for all (j, k, S), j is d-separated from k given $S \subset V \setminus \{j,k\}$ in G, then*

$$X_j \perp\!\!\!\perp X_k \mid X_S \quad \textit{according to } \mathbb{P}.$$

The CMC applies to both acyclic and cyclic graphs (see e.g., [66, 68]).

2.1.3 Markov Equivalence Class

In general, there are many directed graphs entailing the same d-separation rules. These graphs are *Markov equivalent* and the set of Markov equivalent graphs is called a *Markov equivalence class* (MEC) [68, 60, 72, 75]. For example, consider two 2-node graphs, $G_1 : X_1 \to X_2$ and $G_2 : X_1 \leftarrow X_2$. Then both graphs are Markov equivalent because they both entail no d-separation rules. Hence, G_1 and G_2 belong to the same MEC and hence it is impossible to distinguish two graphs by d-separation rules. The precise definition of MEC is provided here:

Definition 2.3 (Markov Equivalence [60])**.** *Directed graphs G_1 and G_2 are* Markov equivalent *if any distribution which satisfies the CMC with respect to one graph satisfies it with respect to the other, and vice versa. The set of graphs which are Markov equivalent to G is denoted by $\mathcal{M}(G)$.*

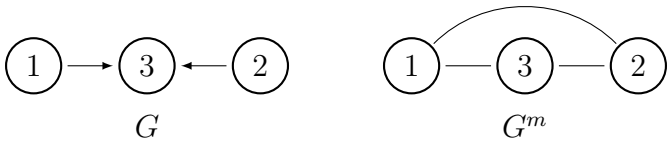


Figure 2.1:: Moralized graph G^m for DAG G

For DAG models, Verma and Pearl [72] developed an elegant characterization of Markov equivalence classes defined by the *skeleton* and *v-structures*. The skeleton of a DAG model consists of the edges without directions:

Theorem 2.1 (Local Markov property, Theorem 1 in [72]). *Two DAGs G_1 and G_2 belong to the same Markov equivalence class if and only if they have the same skeleton and v-structures.*

However the presence of directed cycle means the characterization of the Markov equivalence classes for DCGs is considerably more involved. Richardson [58, 60] extended the notion of unshielded triple to DCG models and provide a characterization of Markov equivalence. Since it is quite involved, we do not include here.

2.1.4 Moral Graph

A *moral* graph is an undirected graphical model representation of a DAG (see e.g., [14]). The moralized graph G^m for a DAG $G = (V, E)$ is an undirected graph where $G^m = (V, E^m)$ where E^m includes edge set E without directions plus edges between any nodes that are parents of a common child. Figure 2.1 demonstrates concepts of a moralized graph for a simple 3-node example where $E = \{(1,3),(2,3)\}$ for the DAG G. Since nodes 1 and 2 are parents of a common child 3, the additional edge $(1,2)$ arises, and therefore $E^m = \{(1,2),(1,3),(2,3)\}$. The neighborhood set of a node j refers to the adjacent nodes to j in the moralized graph $\mathcal{N}(j) := \{k \in V : (j,k) \text{ or } (k,j) \in E^m\}$.

2.2 Overview of Structure Learning

Given a observational data containing independent and identically distributed (iid) instances sampled from a probability distribution $\mathbb{P}$ corresponding to a graph G, the ultimate goal of learning is to recover the structure of the graph G. In general there are two main strategies for graph structure learning: (1) *scoring-based* algorithms and (2) *constraint-based* algorithms.

2.2.1 Scoring-based Algorithms

Scoring-based algorithms search over a possible space of directed graphs to find the graph with the highest score given the observations. Typical examples of scoring functions are the BIC [61], AIC [2], and modified Bayesian Dirichlet equivalent (mBDe) [29]. A popular score-based algorithm for DAG models is Greedy Equivalence Search (GES) algorithm [9]. Scoring-based algorithms are in general flexible and choose high-likelihood graph structure but do not enforce CI statements and often do not accurately recover the true graph [69, 1]. Another challenge for scoring-based methods is that searching over the space of DAGs is NP-hard due to exponential growth in graph structures [8]. Since an exhaustive search algorithm is not possible, existing structure learning algorithms either solve a restricted problem (i.e., choose the best graph or find the Markov equivalence class in the restricted space of directed graphs).

2.2.2 Constraint-based Algorithms

Constraint-based algorithms learn the structure of a directed graph by using the estimated CI statements from observational data. The estimated CI statements are viewed as constraints on the final graph structure, and constraint-based algorithms select a graph that is consistent with those constraints. The most widely used constraint-based algorithms are the SGS algorithm [22] and the PC algorithm [68] for DAG models and CCD algorithm [59] and FCI+ algorithm [12] for DCG models. In contrast to score-based algorithms, a lot of constraint-based algorithms have

been proven to be theoretically consistent. However, the set of CI statements according to $\mathbb{P}$ in general do not entail a unique graph. Hence accurately identifying CI statements present in observational data may only be able to identify Markov equivalence class of a graph rather than a graph including directions of edges. Furthermore constraint-based algorithms often require very strong assumptions such as the faithfulness assumption [68].

2.2.3 Hybrid Algorithms

Hybrid algorithms are also introduced to take advantage of both constraint-based algorithms and score-based algorithms. Two of the most widely used hybrid algorithms are Sparse Candidate algorithm [21] and the Max-Min Hill-Climbing (MMHC) algorithm [71]. Both algorithms first estimate a *skeleton* (which is a structure without directions) using CI statements and then perform a greedy search over graph structure space that respect the skeleton output. However, hybrid algorithms also suffer from disadvantages of both constraint-based algorithms and score-based algorithms where algorithms require strong assumptions and identify a graph up to Markov equivalence class.

Chapter 3

Learning Poisson DAG Model

3.1 Introduction

Modeling large-scale multivariate count data is an important challenge that arises in numerous applications such as neuroscience, systems biology and many others. One approach that has received significant attention is the graphical modeling framework since graphical models include a broad class of dependence models for different data types. Broadly speaking, there are two sets of graphical models: (1) undirected graphical models or Markov random fields and (2) directed acyclic graphical (DAG) models or Bayesian networks.

Between undirected graphical models and DAGs, undirected graphical models have generally received more attention in the large-scale data setting since both learning and inference algorithms scale to larger datasets. In particular, for multivariate count data Yang et al. [77] introduce undirected Poisson graphical models. Yang et al. [77] define undirected Poisson graphical models so that each node is a Poisson random variable with rate parameter depending only on its neighboring nodes in the graph. As pointed out in Yang et al. [77] one of the major challenges with Poisson undirected graphical models is ensuring global normalizability.

Directed acyclic graphs (DAGs) or Bayesian networks are a different class of generative models that model directional or causal relationships (see e.g. [72, 68] for

details). Such directional relationships naturally arise in most applications but are difficult to model based on observational data. One of the benefits of DAG models is that they have a straightforward factorization into conditional distributions [39], and hence no issues of normalizability arise as they do for undirected graphical models as mentioned earlier. However a number of challenges arise that make learning DAG models often impossible for large datasets even when variables have a natural causal or directional structure. These issues are: (1) identifiability since inferring causal directions from data is often not possible; (2) computational complexity since it is often computationally infeasible to search over the space of DAGs [8]; (3) sample size guarantee since fundamental identifiability assumptions such as faithfulness are often required extremely large sample sizes to be satisfied even when the number of nodes p is small (see e.g., [73]).

In this paper, we define Poisson DAG models and address these 3 issues. In Section 3.3 we prove that Poisson DAG models are identifiable and in Section 3.4 we introduce a polynomial-time DAG learning algorithm for Poisson DAGs which we call OverDispersion Scoring (ODS). The main idea behind proving identifiability is based on the *overdispersion* of variables that are conditionally Poisson but not marginally Poisson. Using overdispersion, we prove that it is possible to learn the causal ordering of Poisson DAGs using a polynomial-time algorithm and once the ordering is known, the problem of learning DAGs reduces to a simple set of neighborhood regression problems. While overdispersion with conditionally Poisson random variables is a well-known phenomena that is exploited in many applications (see e.g. [15, 81, 7]), using overdispersion has never been exploited in DAG model learning to our knowledge.

Statistical guarantees for learning the causal ordering are provided in Section 3.4.2 and we provide numerical experiments on both small DAGs and large-scale DAGs with node-size up to 5000 nodes. Our theoretical guarantees prove that even in the setting where the number of nodes p is larger than the sample size n, it is possible to learn the causal ordering under the assumption that the degree of the so-called moralized

graph of the DAG has small degree. Our numerical experiments support our theoretical results and show that our ODS algorithm performs well compared to other state-of-the-art DAG learning methods. Our numerical experiments confirm that our ODS algorithm is one of the few DAG-learning algorithms that performs well in terms of statistical and computational complexity in the high-dimensional $p > n$ setting.

3.2 Poisson DAG Models

In this section, we define general Poisson DAG models. A DAG $G = (V, E)$ consists of a set of vertices V and a set of directed edges E with no directed cycle. We usually set $V = \{1, 2, \ldots, p\}$ and associate a random vector $(X_1, X_2, \ldots, X_p)$ with probability distribution $\mathbb{P}$ over the vertices in G. A directed edge from vertex j to k is denoted by (j, k) or $j \to k$. The set $\mathrm{pa}(k)$ of *parents* of a vertex k consists of all nodes j such that $(j, k) \in E$. One of the convenient properties of DAG models is that the joint distribution $f(X_1, X_2, \cdots, X_p)$ factorizes in terms of the conditional distributions as follows [39]:

$$f(X_1, X_2, \cdots, X_p) = \prod_{j=1}^{p} f_j(X_j \mid X_{\mathrm{pa}(j)}),$$

where $f_j(X_j \mid X_{\mathrm{pa}(j)})$ refers to the conditional distribution of node X_j in terms of its parents. The basic property of Poisson DAG models is that each conditional distribution $f_j(x_j \mid x_{\mathrm{pa}(j)})$ has a Poisson distribution. More precisely, for Poisson DAG models:

$$X_j \mid X_{\{1,2,\cdots,p\}\setminus\{j\}} \sim \mathrm{Poisson}(g_j(X_{\mathrm{pa}(j)})), \tag{3.1}$$

where $g_j(\cdot)$ is an arbitrary function of $X_{\mathrm{pa}(j)}$. To take a concrete example, $g_j(\cdot)$ can represent the link function for the univariate Poisson generalized linear model (GLM) or $g_j(X_{\mathrm{pa}(j)}) = \exp(\theta_j + \sum_{k \in \mathrm{pa}(j)} \theta_{jk} X_k)$ where $(\theta_{jk})_{k \in \mathrm{pa}(j)}$ represent the linear weights.

Using the factorization (3.1), the overall joint distribution is:

$$f(X_1, \cdots, X_p) = \exp\Big(\sum_{j \in V} \theta_j X_j + \sum_{(k,j) \in E} \theta_{jk} X_k X_j - \sum_{j \in V} \log X_j! - \sum_{j \in V} e^{\theta_j + \sum_{k \in \text{pa}_{(j)}} \theta_{jk} X_k} \Big). \tag{3.2}$$

To contrast this formulation with the Poisson undirected graphical model in Yang et al. [77], the joint distribution for undirected graphical models has the form:

$$f(X_1, X_2, \cdots, X_p) = \exp\Big(\sum_{j \in V} \theta_j X_j + \sum_{(k,j) \in E} \theta_{jk} X_k X_j - \sum_{j \in V} \log X_j! - A(\theta) \Big), \tag{3.3}$$

where $A(\theta)$ is the log-partition function or the log of the normalization constant. While the two forms (3.2) and (3.3) look quite similar, the key difference is the normalization constant of $A(\theta)$ in (3.3) as opposed to the term $\sum_{j \in V} e^{\theta_j + \sum_{k \in \text{pa}_{(j)}} \theta_{kj} X_k}$ in (3.2) which depends on X. To ensure the undirected graphical model representation in (3.3) is a valid distribution, $A(\theta)$ must be finite which guarantees the distribution is normalizable and Yang et al. [77] prove that $A(\theta)$ is normalizable if and only if all θ values are less than or equal to 0.

3.3 Identifiability

In this section, we prove that Poisson DAG models are identifiable under a very mild condition. In general, DAG models can only be defined up to their Markov equivalence class (see e.g. [68]). However in some cases, it is possible to identify the DAG by exploiting specific properties of the distribution. For example, Peters and Bühlmann prove that for Gaussian DAGs based on structural equation models with known or the same variance, the models are identifiable [53], Shimizu et al. [64] prove identifiability for linear non-Gaussian structural equation models, and Peters et al. [54] prove identifiability of non-parametric structural equation models with additive independent noise. Here we show that Poisson DAG models are also identifiable using the idea of overdispersion.

To provide intuition, we begin by showing the identifiability of a two-node Poisson DAG model in Figure 3.1. The basic idea is that the relationship between nodes

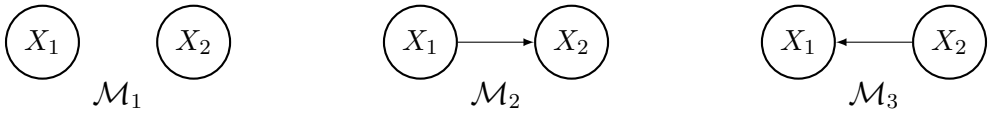


Figure 3.1:: Directed graphs of $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$

X_1 and X_2 generates the overdispersed child variable. To be precise, consider all three models: $\mathcal{M}_1 : X_1 \sim \text{Poisson}(\lambda_1)$, $X_2 \sim \text{Poisson}(\lambda_2)$, where X_1 and X_2 are independent; $\mathcal{M}_2 : X_1 \sim \text{Poisson}(\lambda_1)$ and $X_2 \mid X_1 \sim \text{Poisson}(g_2(X_1))$; and $\mathcal{M}_3 : X_2 \sim \text{Poisson}(\lambda_2)$ and $X_1 \mid X_2 \sim \text{Poisson}(g_1(X_2))$. Our goal is to determine whether the underlying DAG model is $\mathcal{M}_1, \mathcal{M}_2$ or $\mathcal{M}_3$.

Now we exploit the fact that for a Poisson random variable X, $\text{Var}(X) = \mathbb{E}(X)$, while for a distribution which is a conditionally Poisson, the variance is overdispersed relative to the mean. For $\mathcal{M}_1$, $\text{Var}(X_1) = \mathbb{E}(X_1)$ and $\text{Var}(X_2) = \mathbb{E}(X_2)$. For $\mathcal{M}_2$, $\text{Var}(X_1) = \mathbb{E}(X_1)$, while

$$\text{Var}(X_2) = \mathbb{E}[\text{Var}(X_2 \mid X_1)] + \text{Var}[\mathbb{E}(X_2 \mid X_1)] = \mathbb{E}[g_2(X_1)] + \text{Var}[g_2(X_1)] > \mathbb{E}(X_2),$$

as long as $\text{Var}(g_2(X_1)) > 0$.

Similarly under $\mathcal{M}_3$, it is clear that $\text{Var}(X_2) = \mathbb{E}(X_2)$ and $\text{Var}(X_1) > \mathbb{E}(X_1)$ as long as $\text{Var}(g_1(X_2)) > 0$. Hence we can identify model $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$ by testing whether the variance is greater than the expectation or equal to the expectation. With finite sample size n, the quantities $\mathbb{E}(\cdot)$ and $\text{Var}(\cdot)$ can be estimated from data and we consider the finite sample setting in Sections 3.4 and 3.4.2. Now we extend this idea to provide an identifiability condition for general Poisson DAG models.

The key idea to extending identifiability from the bivariate to multivariate scenario involves condition on parents of each node and then testing overdispersion. The general p-variate result is as follows:

Theorem 3.1. *Assume that for any $j \in V$, $K \subset pa(j)$ and $S \subset \{1, 2, .., p\} \setminus K$,*

$$Var(g_j(X_{pa(j)}) \mid X_S) > 0,$$

the Poisson DAG model is identifiable.

We defer the proof to the supplementary material. Once again, the main idea of the proof is overdispersion. To explain the required assumption note that for any $j \in V$ and $S \subset \text{pa}(j)$, $\text{Var}(X_j \mid X_S) - \mathbb{E}(X_j \mid X_S) = \text{Var}(g_j(X_{\text{pa}(j)}) \mid X_S)$. Note that if $S = \text{pa}(j)$ or $\{1, \cdots, j-1\}$, $\text{Var}(g_j(X_{\text{pa}(j)}) \mid X_S) = 0$. Otherwise $\text{Var}(g_j(X_{\text{pa}(j)}) \mid X_S) > 0$ by our assumption.

3.4 Algorithm

Our algorithm which we call OverDispersion Scoring (ODS) consists of three main steps: 1) estimating a candidate parents set [77, 71, 3] using existing learning undirected graph algorithms; 2) estimating a causal ordering using overdispersion scoring; and 3) estimating directed edges using standard regression algorithms such as Lasso. Steps 3) is a standard problem in which we use off-the-shelf algorithms. Step 1) allows us to reduce both computational and sample complexity by exploiting sparsity of the moralized or undirected graphical model representation of the DAG which we introduce shortly. Step 2) exploits overdispersion to learn a causal ordering.

Let $\{X^{(i)}\}_{i=1}^n$ denote iid n samples drawn from the Poisson DAG model G. Let $\pi : \{1, 2, \cdots, p\} \to \{1, 2, \cdots, p\}$ be a bijective function corresponding to a permutation or a causal ordering. We will also use the convenient notation $\widehat{\cdot}$ to denote an estimate based on the data. For ease of notation for any $j \in \{1, 2, \cdots, p\}$, and $S \subset \{1, 2, \cdots, p\}$ let $\mu_{j|S}$ and $\mu_{j|S}(x_S)$ represent $\mathbb{E}(X_j \mid X_S)$ and $\mathbb{E}(X_j \mid X_S = x_S)$, respectively. Furthermore let $\sigma^2_{j|S}$ and $\sigma^2_{j|S}(x_S)$ denote $\text{Var}(X_j \mid X_S)$ and $\text{Var}(X_j \mid X_S = x_S)$, respectively. We also define $n(x_S) = \sum_{i=1}^n \mathbf{1}(X_S^{(i)} = x_S)$ and $n_S = \sum_{x_S} n(x_S)\mathbf{1}(n(x_S) \geq c_0 \cdot n)$ for an arbitrary $c_0 \in (0, 1)$.

The computation of the score $\widehat{s}_{jk}$ in Step 2) of our ODS algorithm 3.1 involves the following equation:

$$\widehat{s}_{jk} = \sum_{x \in \mathcal{X}(\widehat{C}_{jk})} \frac{n(x)}{n_{\widehat{C}_{jk}}} \big(\widehat{\sigma}^2_{j|\widehat{C}_{jk}}(x) - \widehat{\mu}_{j|\widehat{C}_{jk}}(x)\big) \tag{3.4}$$

Algorithm 3.1 OverDispersion Scoring (ODS)

Input: n samples from the given Poisson DAG model. $X^{(1)}, \cdots, X^{(n)} \in \{\{0\} \cup \mathbb{N}\}^p$
Output: A causal ordering $\widehat{\pi} \in \mathbb{N}^p$ and a graph structure, $\widehat{E} \in \{0,1\}^{p \times p}$
Step 1: Estimate the undirected edges $\widehat{E}_u$ corresponding to the moralized graph with neighborhood set $\widehat{\mathcal{N}}(j)$
Step 2: Estimate causal ordering using overdispersion score
for $i \in \{1, 2, \cdots, p\}$ **do**
    $\widehat{s}_i = \widehat{\sigma}_i^2 - \widehat{\mu}_i$
end for
The first element of a causal ordering $\widehat{\pi}_1 = \arg\min_j \widehat{s}_j$
for $j = \{2, 3, \cdots, p-1\}$ **do**
    for $k \in \mathcal{N}(\widehat{\pi}_{j-1}) \cap \{1, 2, \cdots, p\} \setminus \{\widehat{\pi}_1, \cdots, \widehat{\pi}_{j-1}\}$ **do**
        The candidate parents set $\widehat{C}_{jk} = \widehat{\mathcal{N}}(k) \cap \{\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_{j-1}\}$
        Calculate $\widehat{s}_{jk}$ using (3.4);
    end for
    The j^{th} element of a causal ordering $\widehat{\pi}_j = \arg\min_k \widehat{s}_{jk}$
    Step 3: Estimate directed edges toward $\widehat{\pi}_j$, denoted by $\widehat{D}_j$
end for
The p^{th} element of the causal ordering $\widehat{\pi}_p = \{1, 2, \cdots, p\} \setminus \{\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_{p-1}\}$
The directed edges toward $\widehat{\pi}_p$, denoted by $\widehat{D}_p = \widehat{\mathcal{N}}(\widehat{\pi}_p)$
Return: $\widehat{\pi} = (\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_p)$ and $\widehat{E} = \{\widehat{D}_2, \widehat{D}_3, \cdots, \widehat{D}_p\}$

where $\widehat{C}_{jk}$ refers to an estimated candidate set of parents specified in Step 2) of our ODS algorithm 3.1 and $\mathcal{X}(\widehat{C}_{jk}) = \{x \in \{X^{(1)}_{\widehat{C}_{jk}}, X^{(2)}_{\widehat{C}_{jk}}, \cdots, X^{(n)}_{\widehat{C}_{jk}}\} : n(x) \geq c_0.n\}$ so that we ensure we have enough samples for each element we select. In addition, c_0 is a tuning parameter of our algorithm that we specify in our main Theorem 3.3 and our numerical experiments.

We can use a number of standard algorithms for Step 1) of our ODS algorithm since it boils down to finding a candidate set of parents. The main purpose of Step 1) is to reduce both computational complexity and the sample complexity by exploiting sparsity in the moralized graph. In Step 1) a candidate set of parents is generated for each node which in principle could be the entire set of nodes. However since Step 2)

requires computation of a conditional mean and variance, both the sample complexity and computational complexity depend significantly on the number of variables we condition on as illustrated in Section 3.4.1 and 3.4.2. Hence by making the set of candidate parents for each node as small as possible we gain significant computational and statistical improvements by exploiting the graph structure. A similar step is taken in the MMHC [70] and SC algorithms [21]. The way we choose a candidate set of parents is by learning the moralized graph G^m and then using the neighborhood set $\mathcal{N}(j)$ for each j. Hence Step 1) reduces to a standard undirected graphical model learning algorithm. A number of choices are available for Step 1) including the neighborhood regression approach of Yang et al. [77] as well as standard DAG learning algorithms which find a candidate parents set such as HITON [3] and MMPC [70].

Step 2) learns the causal ordering by assigning an overdispersion score for each node. The basic idea is to determine which nodes are overdispersed based on the sample conditional mean and conditional variance. The causal ordering is determined one node at a time by selecting the node with the smallest overdispersion score which is representative of a node that is least likely to be conditionally Poisson and most likely to be marginally Poisson. Finding the causal ordering is usually the most challenging step of DAG learning, since once the causal ordering is learnt, all that remains is to find the edge set for the DAG. Step 3), the final step finds the directed edge set of the DAG G by finding the parent set of each node. Using Steps 1) and 2), finding the parent set of node j boils down to selecting which variables are parents out of the candidate parents of node j generated in Step 1) intersected with all elements before node j of the causal ordering in Step 2). Hence we have p regression variable selection problems which can be performed using GLMLasso [18] as well as standard DAG learning algorithms.

3.4.1 Computational Complexity

Steps 1) and 3) use existing algorithms with known computational complexity. Clearly the computational complexity for Steps 1) and 3) depend on the choice of algorithm. For example, if we use the neighborhood selection GLMLasso algorithm [18] as is used in Yang et al. [77], the worst-case complexity is $O(\min(n,p)np)$ for a single Lasso run but since there are p nodes, the total worst-case complexity is $O(\min(n,p)np^2)$. Similarly if we use GLMLasso for Step 3) the computational complexity is also $O(\min(n,p)np^2)$. As we show in numerical experiments, DAG-based algorithms for Step 1) tend to run more slowly than neighborhood regression based on GLMLasso.

For Step 2) where we estimate the causal ordering has $(p-1)$ iterations and each iteration has a number of overdispersion scores $\widehat{s}_j$ and $\widehat{s}_{jk}$ computed which is bounded by $O(|K|)$ where K is a set of candidates of each element of a causal ordering, $\mathcal{N}(\widehat{\pi}_{j-1}) \cap \{1,2,\cdots,p\} \setminus \{\widehat{\pi}_1, \cdots \widehat{\pi}_{j-1}\}$, which is also bounded by the maximum degree of the moralized graph d. Hence the total number of overdispersion scores that need to be computed is $O(pd)$. Since the time for calculating each overdispersion score which is the difference between a conditional variance and expectation is proportional to n, the time complexity is $O(npd)$. In worst case where the degree of the moralized graph is p, the computational complexity of Step 2) is $O(np^2)$. As we discussed earlier there is a significant computational saving by exploiting a sparse moralized graph which is why we perform Step 1) of the algorithm. Hence Steps 1) and 3) are the main computational bottlenecks of our ODS algorithm. The addition of Step 2) which estimates the causal ordering does not significantly add to the computational bottleneck. Consequently our ODS algorithm, which is designed for learning DAGs is almost as computationally efficient as standard methods for learning undirected graphical models.

3.4.2 Statistical Guarantees

In this section, we show statistical guarantees for recovering the causal ordering of our algorithm under suitable regularity conditions. We begin by stating the assumptions we impose on DAG models.

Assumption 3.2.

(A1) *For all $j \in V$, $K \subset pa(j)$ and all $S \subset \{1, 2.., p\} \setminus K$, there exists an $m > 0$ such that $Var(g_j(X_{pa(j)}) \mid X_S) > m$.*

(A2) *For all $j \in V$, there exists an $M < \infty$ such that $\mathbb{E}[exp(g_j(X_{pa(j)}))] < M$.*

(A1) is a stronger version of the identifiability assumption in Theorem 3.1 where since we are in the finite sample setting, we need the conditional variance to be lower bounded by a constant bounded away from 0. (A2) is a condition on the tail behavior of $g_j(\text{pa}(j))$ for controlling tails of the score $\widehat{s}_{jk}$ in Step 2 of our ODS algorithm. To take a concrete example for which (A1) and (A2) are satisfied, it is straightforward to show that the GLM DAG model (3.2) with non-positive values of $\{\theta_{kj}\}$ satisfies both (A1) and (A2). The non-positivity constraint on the θ's is sufficient but not necessary and ensures that the parameters do not grow too large.

Now we present the main result under Assumptions (A1) and (A2). For general DAGs, the true causal ordering π^* is not unique. Therefore let $\mathcal{E}(\pi^*)$ denote all the causal orderings that are consistent with the true DAG G^*. Further recall that d denotes the maximum degree of the moralized graph G^*_m.

Theorem 3.3 (Recovery of a causal ordering)**.** *Consider a Poisson DAG model as specified in (3.1), with a set of true causal orderings $\mathcal{E}(\pi^*)$ and the rate function $g_j(\cdot)$ satisfies assumptions 3.2. If the sample size threshold parameter $c_0 \leq n^{-1/(5+d)}$, then there exist positive constants, C_1, C_2, C_3 such that*

$$\mathbb{P}(\hat{\pi} \notin \mathcal{E}(\pi^*)) \leq C_1 exp(-C_2 n^{1/(5+d)} + C_3 \log \max\{n, p\}).$$

We defer the proof to the supplementary material. The main idea behind the proof uses the overdispersion property exploited in Theorem 3.1 in combination with concentration bounds that exploit Assumption (A2). Note once again that the maximum degree d of the undirected graph plays an important role in the sample complexity which is why Step 1) is so important. This is because the size of the conditioning set depends on the degree of the moralized graph d. Hence d plays an important role in both the sample complexity and computational complexity.

Theorem 3.3 can be used in combination with sample complexity guarantees for Steps 1) and 3) of our ODS algorithm to prove that our output DAG $\widehat{G}$ is the true DAG G^* with high probability. Sample complexity guarantees for Steps 1) and 3) depend on the choice of algorithm but for neighborhood regression based on the GLMLasso, provided $n = \Omega(d \log p)$, Steps 1) and 3) should be consistent.

For Theorem 3.3 if the triple (n, d, p) satisfies $n = \Omega((\log p)^{5+d})$, then our ODS algorithm recovers the true DAG. Hence if the moralized graph is sparse, ODS recovers the true DAG in the high-dimensional $p > n$ setting. DAG learning algorithms that apply to the high-dimensional setting are not common since they typically rely on faithfulness or similar assumptions or other restrictive conditions that are not satisfied in the $p > n$ setting. Note that if the DAG is not sparse and $d = \Omega(p)$, our sample complexity is extremely large when p is large. This makes intuitive sense since if the number of candidate parents is large, we would need to condition on a large set of variables which is very sample-intensive. Our sample complexity is certainly not optimal since the choice of tuning parameter $c_0 \leq n^{-1/(5+d)}$. Determining optimal sample complexity remains an open question.

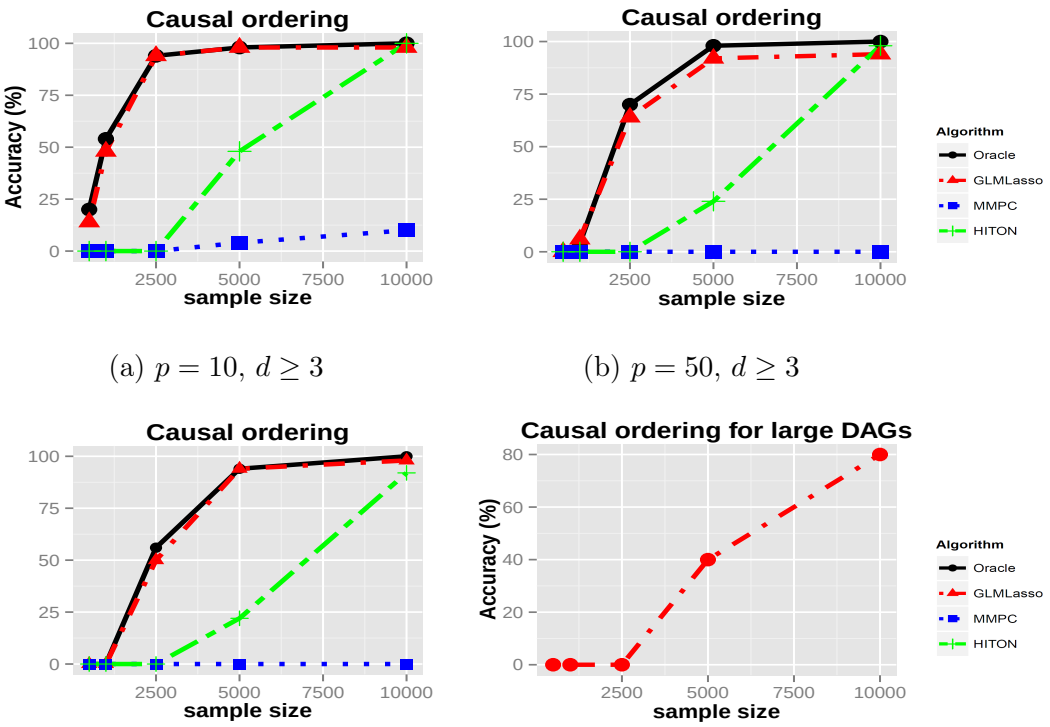
The larger sample complexity of our ODS algorithm relative to undirected graphical models learning is mainly due to the fact that DAG learning is an intrinsically harder problem than undirected graph learning when the causal ordering is unknown. Furthermore note that Theorem 3.3 does not require any additional identifiability assumptions such as faithfulness which severely increases the sample complexity for

large-scale DAGs [73].

3.5 Numerical Experiments

In this section, we support our theoretical results with numerical experiments and show that our ODS algorithm performs favorably compared to state-of-the-art DAG learning methods. The simulation study was conducted using 50 realizations of a p-node random Poisson DAG that was generated as follows. The $g_j(\cdot)$ functions for the general Poisson DAG model (3.1) was chosen using the standard GLM link function (i.e.$g_j(X_{\text{pa}(j)}) = \exp(\theta_j + \sum_{k\in\text{pa}(j)} \theta_{jk}X_k)$) resulting in the GLM DAG model (3.2). We experimented with other choices of $g_j(\cdot)$ but only present results for the GLM DAG model (3.2). Note that our ODS algorithm works well as long as Assumption 3.2 is satisfied regardless of choices of $g_j(\cdot)$. In all results presented (θ_{jk}) parameters were chosen uniformly at random in the range $\theta_{jk} \in [-1, -0.7]$ although any values far from zero and satisfying the assumption 3.2 work well. In fact, smaller values of θ_{jk} are more favorable to our ODS algorithm than state-of-the-art DAG learning methods because of weak dependency between nodes. DAGs are generated randomly with a fixed unique causal ordering $\{1, 2 \cdots, p\}$ with edges randomly generated while respecting desired maximum degree constraints for the DAG. In our experiments, we always set the thresholding constant $c_0 = 0.005$ although any value below 0.01 seems to work well.

In Fig. 3.2, we plot the proportion of simulations in which our ODS algorithm recovers the correct causal ordering in order to validate Theorem 3.3. All graphs in Fig. 3.2 have exactly 2 parents for each node and we plot how the accuracy in recovering the true π^* varies as a function of n for $n \in \{500, 1000, 2500, 5000, 10000\}$ and for different node sizes (a) $p = 10$, (b) $p = 50$, (c) $p = 100$, and (d) $p = 5000$. As we can see, even when $p = 5000$, our ODS algorithm recovers the true causal ordering about 40% of the time even when n is approximately 5000 and for smaller DAGs accuracy is 100%. In each sub-figure, 3 different algorithms are used for Step 1): GLMLasso [18] where we choose $\lambda = 0.1$; MMPC [70] with $\alpha = 0.005$; and HITON [3]



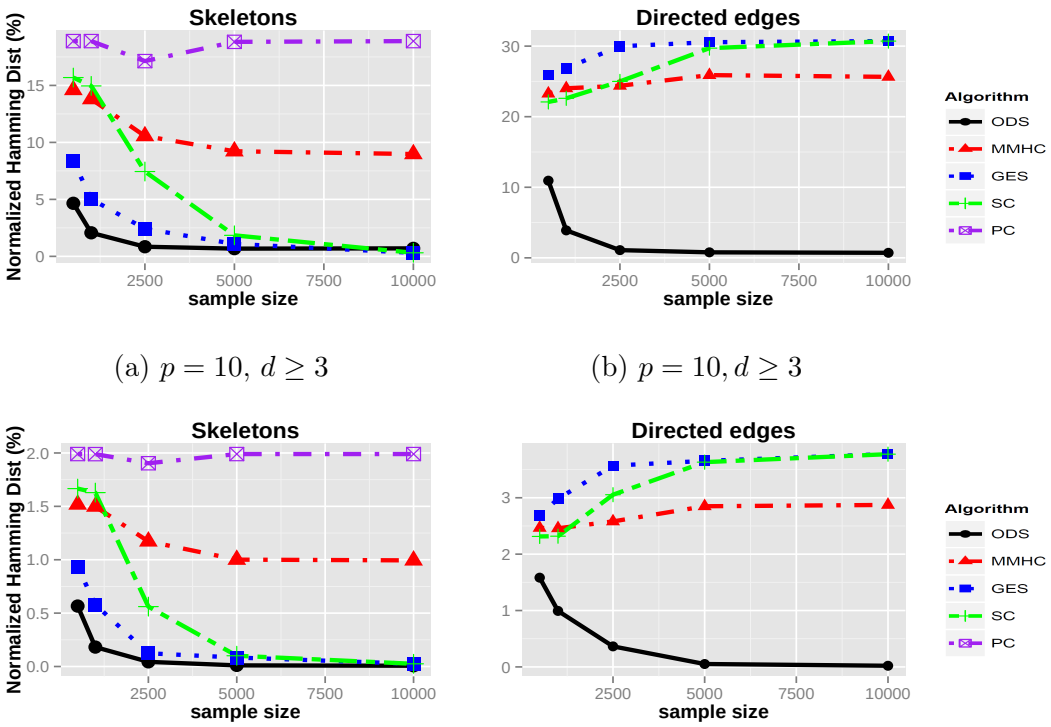
(a) $p = 10$, $d \geq 3$ (b) $p = 50$, $d \geq 3$

(c) $p = 100$, $d \geq 3$ (d) $p = 5000$, $d \geq 3$

Figure 3.2:: Accuracy rates of successful recovery for a causal ordering via our ODS algorithm using different base algorithms

again with $\alpha = 0.005$ and an oracle where the edges for the true moralized graph is used. As Fig. 3.2 shows, the GLMLasso seems to be the best performing algorithm in terms of recovery so we use the GLMLasso for Steps 1) and 3) for the remaining figures. GLMLasso was also the only algorithm that scaled to the $p = 5000$ setting. However, it should be pointed out that GLMLasso is not necessarily consistent and it is highly depending on the choice of $g_j(\cdot)$. Recall that the degree d refers to the maximum degree of the moralized DAG.

Fig. 3.3 provides a comparison of how our ODS algorithm performs in terms of Hamming distance compared to the state-of-the-art PC [68], MMHC [70], GES [9], and SC [21] algorithms. For the PC, MMHC and SC algorithms, we use $\alpha = 0.005$ while for the GES algorithm we use the mBDe [29] (modified Bayesian Dirichlet equivalent)

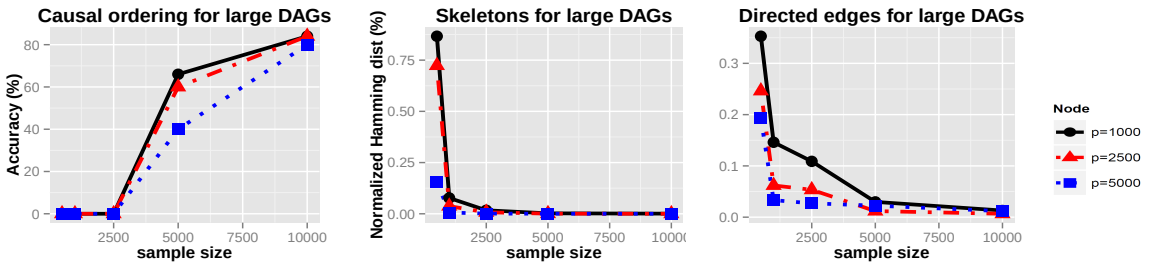


(a) $p = 10,\ d \geq 3$ (b) $p = 10, d \geq 3$

(c) $p = 100, d \geq 3$ (d) $p = 100, d \geq 3$

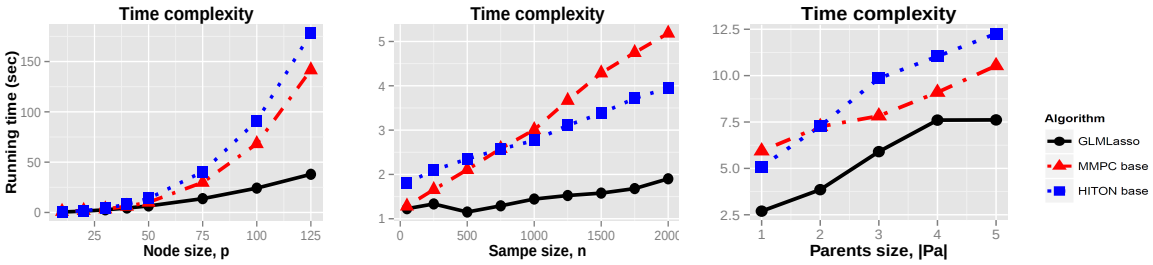
Figure 3.3:: Comparison of our ODS algorithm (black) and PC, GES, MMHC, SC algorithms in terms of Hamming distance to skeletons and directed edges.

score since it performs better than other score choices. We consider node sizes of $p = 10$ in (a) and (b) and $p = 100$ in (c) and (d) since many of these algorithms do not easily scale to larger node sizes. We consider two Hamming distance measures: in (a) and (c), we only measure the Hamming distance to the skeleton of the true DAG, which is the set of edges of the DAG without directions; for (b) and (d) we measure the Hamming distance for the edges with directions. The reason we consider the skeleton is because the PC does not recover all directions of the DAG. We normalize the Hamming distance by dividing by the total number of edges $\binom{p}{2}$ and $p(p-1)$, respectively so that the overall score is a percentage. As we can see our ODS algorithm significantly out-performs the other algorithms. We can also see that as the sample size n grows, our algorithm recovers the true DAG which is consistent with our theoretical results.



(a) $d \geq 3$ (b) $d \geq 3$ (c) $d \geq 3$

Figure 3.4:: Performance of our ODS algorithm for large-scale DAGs with $p = 1000, 2500, 5000$



(a) $n = 100, d \geq 3$ (b) $p = 20, d \geq 3$ (c) $n = 5000, p = 20$

Figure 3.5:: Time complexity of our ODS algorithm with respect to node size p, sample size n, and parents size |pa|

It must be pointed out that the choice of DAG model is suited to our ODS algorithm while these state-of-the-art algorithms apply to more general classes of DAG models.

Now we consider the statistical performance for large-scale DAGs. Fig. 3.4 plots the statistical performance of ODS for large-scale DAGs in terms of (a) recovering the causal ordering; (b) Hamming distance to the true skeleton; (c) Hamming distance to the true DAG with directions. All graphs in Fig. 3.4 have exactly 2 parents for each node and accuracy varies as a function of n for $n \in \{500, 1000, 2500, 5000, 10000\}$ and for different node sizes $p = \{1000, 2500, 5000\}$. Fig. 3.4 shows that our ODS algorithm accurately recovers the causal ordering and true DAG models even in high dimensional

setting, supporting our theoretical results 3.3.

Fig. 3.5 shows run-time of our ODS algorithm. We measure the running time (a) by varying node size p from 10 to 125 with the fixed $n = 100$ and 2 parents; (b) sample size n from 100 to 2500 with the fixed $p = 20$ and 2 parents; (c) the number of parents of each node $|\text{pa}|$ from 1 to 5 with the fixed $n = 5000$ and $p = 20$. Fig. 3.5 (a) and (b) support the section 3.4.1 where the time complexity of our ODS algorithm is at most $O(np^2)$. Fig. 3.5 (c) shows running time is proportional to a parents size which is a minimum degree of a graph. It agrees with the time complexity of Step 2) of our ODS algorithm is $O(npd)$. We can also see that the GLMLasso has the fastest run-time amongst all algorithms that determine the candidate parent set.

Chapter 4

Learning QVF DAG Models

4.1 Introduction

Probabilistic directed acyclic graphical (DAG) models or Bayesian networks are a widely used framework for representing causal, directional or dependence relationships between multiple variables. These models have applications in various areas such as genomics, neuroimaging, statistical physics, spatial statistics and many others (see e.g. []). One of the fundamental problems associated with DAG models is learning DAG models from observational data.

However, a number of challenges arise that make learning DAG models often impossible for large-scale data even when variables have a natural causal or directional structure. These issues are: (1) identifiability since inferring causal directions from data is often not possible; (2) computational complexity since it is often computationally infeasible to search over the space of DAGs [8]; (3) sample size guarantee since fundamental identifiability assumptions, such as the faithfulness [68] often requires an extremely large sample size n to be satisfied even when the number of nodes p is small (see e.g., [73]).

Regarding the identifiability issue, DAG models can only be identified up to their Markov equivalence class (see e.g., [68]). However, recent work shows that it is possible to fully identify the DAG structure including directions by exploiting characterization

of the node probability distribution. Shimizu et al. [64] proved identifiability for linear non-Gaussian structural equation models, and Peters et al. [54] proved identifiability for non-parametric structural equation models with additive independent noise. Peters and Bühlmann [53] proved identifiability for Gaussian DAG models based on structural equation models with known or the same variance of errors, and Park and Raskutti [46] proved identifiability for Poisson DAG models using the notion of overdispersion.

The major contributions of our paper are to (i) introduce a new class of identifiable directed graphical models where each node has a quadratic variance function (QVF) conditional distribution; (ii) introduce a general OverDispersion Scoring (ODS) algorithm that applies to our class of QVF DAG models; (iii) provide theoretical guarantees for our ODS algorithm which proves that our algorithm is consistent in the high-dimensional setting $p > n$ provided there is underlying sparse structure; and (iv) show through a simulation study that our ODS algorithm has favorable performance to a number of state-of-the-art algorithms for both low-dimensional and high-dimensional DAG models.

The remainder of the paper is organized as follows: In Section 4.2, we describe how we define DAG models with a given probability distribution and we prove the identifiability for our class of DAG models. In Section 4.3, we introduce a polynomial-time DAG learning algorithm for our class of identifiable DAG models which we call generalized OverDispersion Scoring (ODS). The main idea behind proving identifiability is based on the *overdispersion* of variables. As Park and Raskutti [46] discussed about Poisson DAG models, *overdispersion* of variables has potential as a score for recovering the causal ordering of a DAG. However most distributions in general do not satisfy *equidispersion*, therefore we provide a transformation such that each variable is conditionally *equidispersed* and marginally *overdispersed*. While overdispersion is a well-known phenomena that is exploited in many applications (see e.g. [15, 81]), using overdispersion as a score has never been exploited in learning our class of DAG models to our knowledge. Statistical guarantees for learning a DAG model are pro-

vided in Section 4.3.2, and we provide numerical experiments on both small DAGs and large-scale DAGs with node-size up to 5000 nodes in Section 4.5.1. Our theoretical guarantees prove that even in the setting where the number of nodes p is larger than the sample size n, it is possible to learn the DAG structure under the assumption that the degree of the so-called moralized graph of a DAG is small. Our numerical experiments provided in Section 4.5.1 support the theoretical results and show that our algorithm performs well compared to other state-of-the-art DAG learning methods. Our numerical experiments confirm that our algorithm is one of the few DAG-learning algorithms that performs well in terms of statistical and computational complexity in high-dimensional $p > n$ settings.

4.2 Quadratic Variance Function (QVF) DAG models and Identifiability

One of the main objectives of learning DAG models is to determine causal or directional relationships between variables. Therefore, we are interested in determining the conditions that make the DAG models fully identifiable in terms of their edges and directions from observational data. Recent studies proved identifiability of a special class of DAG models using the characterization of the given probability distribution. For example, Peters and Bühlmann [53] proved the identifiability of Gaussian DAG models using the property of Gaussian distribution and the known error variances. Here we introduce a new class of identifiable DAG models using the idea of overdispersion.

We introduce quadratic variance function (QVF) DAG models as models where the conditional distribution of each node given its parents satisfies the following quadratic variance function:

$$\mathrm{Var}(X_j \mid X_{\mathrm{pa}(j)}) = \beta_0 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) + \beta_1 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)})^2. \tag{4.1}$$

Furthermore, this quadratic variance property does not hold for other conditional distributions of a node given variables without some parents.

A popular example is a natural parameter exponential family distribution with quadratic variance function (NEF-QVF) [44] which includes Poisson, Binomial, Negative Binomial, Gamma, and Gaussian distributions. However, Gaussian DAG models do not belong to our QVF DAG models because the variance and expectation of a Gaussian distribution are independent. It is consistent that if all variances are known Gaussian DAG models are identifiable [53].

As a special case, if conditional distribution of each node given its parents is a member of NEF-QVF, then by the factorization property, the joint distribution is given as:

$$P(X) = \exp\left(\theta_j X_j + \sum_{(k,j)\in E} \theta_{jk} X_k X_j - \sum_{j\in V} C_j(X_j) - \sum_{j\in V} D_j(\theta_j + \langle \theta_{\mathrm{pa}(j)}, X_{\mathrm{pa}(j)}\rangle)\right). \tag{4.2}$$

where $C_j(\cdot)$ is the base measure, and $D(\cdot)$ is the log-normalization constant determined by a chosen exponential family distribution. In addition $\theta_{\mathrm{pa}(j)} \in \mathbb{R}^{|\mathrm{pa}(j)|}$ is a parameter vector corresponding the parents of a node j, and $\langle\cdot,\cdot\rangle$ refers to the inner product.

Provided the quadratic variance function (4.1), we can find a transformation $T_j(X_j) = \omega_j X_j$ where $\omega_j = (\beta_0 + \beta_1 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}))^{-1}$ such that $\mathrm{Var}(T_j(X_j) \mid X_{\mathrm{pa}(j)}) = \mathbb{E}(T_j(X_j) \mid X_{\mathrm{pa}(j)})$ for any node $j \in V$. We present some examples of conditional distribution for our QVF DAG models with the triple $(\beta_0, \beta_1, \omega)$ in the following Table 4.1.

To provide intuition, we begin by showing the identifiability of a two-node Poisson DAG model [46]. The basic idea is that the relationship between variables X_1 and X_2 generates the overdispersed child variable. To be precise, consider all three models: $\mathcal{M}_1 : X_1 \sim \mathrm{Poisson}(\lambda_1),\ \ X_2 \sim \mathrm{Poisson}(\lambda_2)$, where X_1 and X_2 are independent; $\mathcal{M}_2 : X_1 \sim \mathrm{Poisson}(\lambda_1)$ and $X_2 \mid X_1 \sim \mathrm{Poisson}(g_2(X_1))$; and $\mathcal{M}_3 : X_2 \sim \mathrm{Poisson}(\lambda_2)$ and $X_1 \mid X_2 \sim \mathrm{Poisson}(g_1(X_2))$ for arbitrary positive functions $g_1, g_2 : \mathbb{R} \to \mathbb{R}^+$. Our goal is to determine whether the underlying DAG model is $\mathcal{M}_1, \mathcal{M}_2$ or $\mathcal{M}_3$.

We exploit the fact that for a Poisson random variable X, $\mathrm{Var}(X) = \mathbb{E}(X)$,

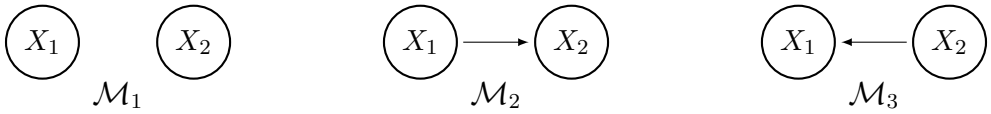


Figure 4.1:: Directed graphical models of $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$

while for a distribution which is a conditionally Poisson, the marginal variance is overdispersed relative to the marginal expectation, $\text{Var}(X) > \mathbb{E}(X)$. Hence for $\mathcal{M}_1$, $\text{Var}(X_1) = \mathbb{E}(X_1)$ and $\text{Var}(X_2) = \mathbb{E}(X_2)$. For $\mathcal{M}_2$, $\text{Var}(Y_1) = \mathbb{E}(Y_1)$, while

$$\text{Var}(X_2) = \mathbb{E}[\text{Var}(X_2 \mid X_1)] + \text{Var}[\mathbb{E}(X_2 \mid X_1)] = \mathbb{E}[\mathbb{E}[X_2 \mid X_1]] + \text{Var}[g_2(X_1)] > \mathbb{E}(X_2),$$

as long as $\text{Var}(g_2(X_1)) > 0$.

Similarly under $\mathcal{M}_3$, $\text{Var}(X_2) = \mathbb{E}(X_2)$ and $\text{Var}(X_1) > \mathbb{E}(X_1)$ as long as $\text{Var}(g_1(X_2)) > 0$. Hence we can distinguish models $\mathcal{M}_1$, $\mathcal{M}_2$, and $\mathcal{M}_3$ by testing whether the variance is greater than the expectation or equal to the expectation. For other QVF DAG models, we can see the same relationship between the variance and expectation after the transformation $T_j(\cdot)$ we discussed (see examples in Table 4.1). With finite sample size, the quantities $\mathbb{E}(\cdot)$ and $\text{Var}(\cdot)$ can be estimated from data and we consider the finite sample setting in Sections 4.3 and 4.3.2.

We extend this idea of overdispersion to provide an identifiability condition for general p-variate DAG models. The key idea to extending identifiability from the

Distribution	β_0	β_1	ω
Binomial, Bin(N, p)	1	$-\frac{1}{N}$	$\frac{N}{N-\mathbb{E}(X)}$
Poisson, Poi(λ)	1	0	1
Generalized Poisson, GPoi(λ_1, λ_2)	$\frac{1}{(1-\lambda_2)^2}$	0	$\frac{1}{(1-\lambda_2)^2}$
Geometric, Geo(p)	1	1	$\frac{1}{1+\mathbb{E}(X)}$
Negative Binomial, NB(R, p)	1	$\frac{1}{R}$	$\frac{R}{R+\mathbb{E}(X)}$
Exponential, Exp(λ)	0	1	$\frac{1}{\mathbb{E}(X)}$
Gamma, Gamma(α, β)	0	$\frac{1}{\alpha}$	$\frac{\alpha}{\mathbb{E}(X)}$

Table 4.1:: Some distributions with β_0, β_1 and ω in our new class of identifiable DAG distributions $\mathcal{C}_2$

bivariate to multivariate scenario involves condition on parents of each node and then testing overdispersion. The general p-variate DAG model result is as follows:

Theorem 4.1 (Identifiability). *Let* $(X_1, X_2, \cdots, X_p)$ *be a random vector associated with a QVF DAG model* $(G, \mathbb{P})$ *with quadratic variance coefficients* (β_0, β_1) *in* (4.1). *Suppose that* $\beta_1 > -1$. *Then for any node* $j \in V$, $K \subset pa(j)$, *and* $S \subset V \setminus K$ *if*

$$Var(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_S) > 0, \tag{4.3}$$

the DAG G *is identifiable.*

We defer the proof to Appendix B.1.1. Theorem 4.1 claims that a QVF DAG model is identifiable if all parents of a node j contribute to the variability of a node j. The identifiable condition is equivalent to transformed variables are overdispersed since $\mathrm{Var}(T_j(X_j) \mid X_S) - \mathbb{E}(T_j(X_j) \mid X_S) = c(1+\beta_1)\mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_S)$ for some constants c (explained in Appendix B.1.1). If $\mathrm{pa}(j) \subseteq S$, $\mathrm{Var}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) \mid X_S) = 0$ and therefore the conditional variance is the same as the conditional expectation. Otherwise, a transformed variable is overdispersed by the identifiability assumption in Theorem 4.1. The condition $\beta_1 > -1$ is important since it rules out DAG models with Bernoulli and Multinomial distributions which are not identifiable.

4.3 Algorithm for QVF DAG Models

We develop a new DAG learning algorithm for count data called generalized OverDispersion Scoring (ODS) algorithm. Our generalized ODS algorithm consists of three main steps: 1) estimating the moralized graph of the DAG using undirected graph learning algorithms; 2) estimating the causal ordering of the DAG using overdispersion scoring; and 3) estimating the DAG using standard regression algorithms such as Lasso. Both Step 1) and Step 3) are standard neighborhood estimation problems in which we use not only regression algorithms, but also off-the-shelf graph learning algorithms (e.g. [77, 71, 3]). Step 1) allows us to reduce both computational and

sample complexity by exploiting the sparsity of the moralized or undirected graphical model representation of a DAG.

Let $\{X^{(1)}, X^{(2)}, \cdots, X^{(n)}\}$ denote iid n samples drawn from a given QVF DAG model $(G, \mathbb{P})$ with a quadratic variance coefficients (β_0, β_1). For any node $j \in V$ and $S \subset V \setminus \{j\}$, let $\mu_{j|S}$ and $\sigma^2_{j|S}$ represent $\mathbb{E}(T_j(X_j) \mid X_S)$ and $\mathrm{Var}(T_j(X_j) \mid X_S)$, respectively. Furthermore for some realizations of $x_S \in X_S$, let $\mu_{j|S}(x_S)$ and $\sigma^2_{j|S}(x_S)$ denote $\mathbb{E}(T_j(X_j) \mid X_S = x_S)$ and $\mathrm{Var}(T_j(X_j) \mid X_S = x_S)$, respectively. We will also use the convenient notation $\widehat{\cdot}$ to denote an estimate based on the data. We use $n(x_S) = \sum_{i=1}^{n} \mathbf{1}(X_S^{(i)} = x_S)$ to denote a total conditional sample size, and $n_S = \sum_{x_S} n(x_S)\mathbf{1}(n(x_S) \geq c_0.n)$ for an arbitrary $c_0 \in (0, 1)$ to denote a truncated conditional sample size. For notational convenience, we use $1 : j = \{1, 2, \cdots, j\}$ and $1 : 0 = \emptyset$. With those notations, let $c_{jm} = (\beta_0 + \beta_1 \mu_{j|1:m-1})^{-1}$ for $m \in V \setminus \{1\}$ and $j \in \{m, m+1, \cdots, p\}$, and $c_{j1} = \beta_0 + \beta_1 \mu_j)^{-1}$. The idea of c_{jm} is from the ω_j of the transformation $T_j(\cdot)$. Since we do not know the parents of a node j, here we consider a candidate set of parents C_{jm} of a node j for the m^{th} element of the causal ordering. A candidate set of parents of a node j is an intersection of an neighbors of a node j and first $m-1$ elements of the causal ordering because the parents of a node j must be in the $\mathcal{N}(j)$ and appear in the causal ordering before a node j. It is estimated in Step 2) of the generalized ODS algorithm 4.1.

The computation of overdispersion scores in Step 2) of the generalized ODS algorithm 4.1 involves the following equations:

$$\widehat{\mathcal{S}}(1, k) = \left[\left(\frac{\widehat{\sigma}_j}{\beta_0 + \beta_1 \widehat{\mu}_j} \right)^2 - \frac{\widehat{\mu}_j}{\beta_0 + \beta_1 \widehat{\mu}_j} \right] \tag{4.4}$$

$$\widehat{\mathcal{S}}(j, k) = \sum_{x \in \mathcal{X}(\widehat{C}_{jk})} \frac{n(x)}{n_{\widehat{C}_{jk}}} \left[\left(\frac{\widehat{\sigma}_{j|\widehat{C}_{jk}}(x)}{\beta_0 + \beta_1 \widehat{\mu}_{j|\widehat{C}_{jk}}(x)} \right)^2 - \frac{\widehat{\mu}_{j|\widehat{C}_{jk}}(x)}{\beta_0 + \beta_1 \widehat{\mu}_{j|\widehat{C}_{jk}}(x)} \right] \tag{4.5}$$

where $\mathcal{X}(\widehat{C}_{jk}) = \{x_{jk} \in \{X^{(1)}_{\widehat{C}_{jk}}, X^{(2)}_{\widehat{C}_{jk}}, \cdots, X^{(n)}_{\widehat{C}_{jk}}\} : n(x_{jk}) \geq c_0.n\}$ to ensure we have enough samples for each element of an overdispersion score. An overdispersion score is

Algorithm 4.1 Generalized OverDispersion Scoring (ODS)

Input: iid n samples from the QVF DAG model
Output: A causal ordering $\widehat{\pi} \in \mathbb{N}^p$ and a graph structure $\widehat{E} \in \{0,1\}^{p \times p}$
Step 1: Estimate the neighborhood of each node $\widehat{\mathcal{N}}(j)$ in the moralized graph
Step 2: Estimate the causal ordering using overdispersion scores
for $k \in \{1, 2, \cdots, p\}$ **do**
    Calculate overdispersion scores $\widehat{\mathcal{S}}(1, k)$ using Equation (4.4)
end for
The first element of a causal ordering $\widehat{\pi}_1 := \arg\min_k \widehat{\mathcal{S}}(1, k)$
for $j = \{2, 3, \cdots, p-1\}$ **do**
    for $k \in \widehat{\mathcal{N}}(\widehat{\pi}_{j-1}) \cap \{1, 2, \cdots, p\} \setminus \{\widehat{\pi}_1, \cdots, \widehat{\pi}_{j-1}\}$ **do**
        Find candidate parents set $\widehat{C}_{jk} := \widehat{\mathcal{N}}(k) \cap \{\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_{j-1}\}$
        Calculate overdispersion scores $\widehat{\mathcal{S}}(j, k)$ using Equation (4.5)
    end for
    The j^{th} element of a causal ordering $\widehat{\pi}_j := \arg\min_k \widehat{\mathcal{S}}(j, k)$
    Step 3: Estimate the directed edges toward $\widehat{\pi}_j$, denoted by $\widehat{D}_j$
end for
The last element of the causal ordering $\widehat{\pi}_p := \{1, 2, \cdots, p\} \setminus \{\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_{p-1}\}$
The directed edges toward $\widehat{\pi}_p$, denoted by $\widehat{D}_p := \{(z, \widehat{\pi}_p) \mid z \in \widehat{\mathcal{N}}(\widehat{\pi}_p)\}$
Return: $\widehat{\pi} := (\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_p)$ and $\widehat{E} := \cup_{j=\{2,3,\cdots,p\}} \widehat{D}_j$

the weighted average of differences between conditional sample means and variances. In addition, c_0 is a tuning parameter of our algorithm that we specify in Theorem 4.8 and our numerical experiments.

The main purpose of Step 1) is to reduce both computational complexity and sample complexity by exploiting the sparsity of the moralized graph. In Step 1), the neighborhood set for each node is estimated which is a superset of a candidate parents set for each node. A candidate parents set is used for a condition set for an overdispersion score in Step 2). In principle, a size of a condition set for an overdispersion score could be $p-1$ if the moralized graph is not applied. Since Step 2) requires computation of a conditional mean and variance, both the computational complexity and sample complexity depend significantly on the number of variables we condition on as

illustrated in Subsection 4.3.1 and 4.3.2. Therefore by making the condition set of for the overdispersion score of each node as small as possible, we gain significant computational and statistical improvements. Furthermore, Step 1) reduces the number of overdispersion scores to be compared in Step 2). If j^{th} element of the causal ordering is estimated, it is sufficient for $j+1^{th}$ element of the causal ordering to consider neighborhood of j^{th} element of the causal ordering in the moralized graph. Since Step 2) compares overdispersion scores of nodes for each component of the causal ordering, by minimizing the number of overdispersion scores to be compared, we obtain significant statistical and computational improvements. A similar step is taken by Loh et al. [40], the MMHC [71] and SC algorithms [21]. Since the moralized graph is an undirected graph, a number of choices are available for Step 1) including standard undirected graph learning algorithms such as the HITON [3] and MMPC algorithms [71] as well as GLMLasso [18]. In addition, standard DAG learning algorithms such as GES [9] and MMHC algorithms [71] can be applied and the moralized graph can be found from the estimated DAG.

The novelty of our generalized ODS algorithm is Step 2) which learns the causal ordering by comparing overdispersion scores of nodes. The basic idea is to determine which nodes are overdispersed based on the sample conditional mean and conditional variance. The causal ordering is determined one node at a time by selecting the node with the smallest overdispersion score which is representative of a node that is least likely to be overdispersed. Finding the causal ordering is usually the most challenging step of DAG learning since once the causal ordering is learned, all that remains is to find the edge set of the DAG.

By using Step 2) of the generalized ODS algorithm, finding the set of parents of a node j boils down to selecting the parents out of all elements before a node j in the estimated causal ordering. Hence, Step 3) can be reduced to p neighborhood estimation problems which can be performed using GLMLasso [18] as well as standard DAG learning algorithms such as the PC [68], GES [9], and MMHC algorithms [71].

4.3.1 Computational Complexity

Steps 1) and 3) of the generalized ODS algorithm use any off-the-shelf algorithms with known computational complexity. Clearly, the computational complexity for Steps 1) and 3) depends on the choice of algorithms. For example, if we use the neighborhood selection GLMLasso algorithm [18] as is used in Yang et al. [77], the worst-case complexity is $O(\min(n,p)np)$ for a single Lasso run but since there are p nodes, the total worst-case complexity is $O(\min(n,p)np^2)$. Similarly, if we use GLMLasso for Step 3) the computational complexity is also $O(\min(n,p)np^2)$.

For Step 2) where we estimate the causal ordering of a DAG, there are $(p-1)$ iterations and each iteration has a number of overdispersion scores $\widehat{S}(j,k)$ to be computed which is bounded by $O(d)$ where d is the maximum degree of the moralized graph. Hence the total number of overdispersion scores that need to be computed is $O(pd)$. Since the time for calculating each overdispersion score which is the difference between a conditional variance and expectation is proportional to the sample size n, the time complexity is $O(npd)$.

In worst case where the degree of the moralized graph is p, the computational complexity of Step 2) is $O(np^2)$. As we discussed earlier, there is a significant computational saving by exploiting the sparsity of the moralized graph which is why we perform Step 1) of the generalized ODS algorithm. Hence, Steps 1) and 3) are the main computational bottlenecks of the generalized ODS algorithm. The addition of Step 2) which estimates the causal ordering does not significantly add to the computational bottleneck. Consequently, the generalized ODS algorithm, which is designed for learning DAGs, is almost as computationally efficient as standard methods for learning graphical models. As we show in numerical experiments, the ODS algorithm using GLMLasso in both Steps 1) and 3) is not slower than state-of-the-art GES algorithm.

4.3.2 Statistical Guarantees

In this section, we study the theoretical guarantees of recovering the structure of a DAG via our generalized ODS algorithm. Although we can use any off-the-shelf algorithms in Steps 1) and 3), we only provide theoretical guarantees of learning the moralized graph and the DAG structure via surrogate GLMLasso in Sections 4.3.2.1 and 4.3.2.3. In addition, we also provide statistical guarantees for learning the causal ordering of a DAG in Section 4.3.2.2. All three main results concern conditions on the triple (n, p, d), sample size n regarding to complexity of the graphical model which are specifically the number of nodes p and the maximum degree of the moralized graph d, ensuring that the generalize ODS algorithm recovers a DAG structure consistently.

We introduce an important lemma to ensure that the true parents of each node are same as the estimated parents via surrogate GLMLasso. To make the definition of the estimated parents via surrogate GLMLasso precise, suppose that $\theta_D^* \in \Theta_D$ denotes the solution of the surrogate GLM problem where $\Theta_D = \{\theta \in \mathbb{R}^{p-1} : \theta_k = 0 \text{ for } k \notin \text{pa}(j)\}$.

$$\theta_D^* := \arg\min_{\theta \in \mathbb{R}^{j-1}} \mathbb{E}\left(-X_j \langle \theta, X_{1:j-1} \rangle + D(\langle \theta, X_{1:j-1} \rangle)\right), \tag{4.6}$$

where $D(\cdot)$ is the log-normalization constant determined by a chosen GLM. Then, the parents of a node j via surrogate GLM is defined as $\widetilde{\text{pa}}(j) := \{k \in V \setminus \{j\} : [\theta_D^*]_k \neq 0\}$ where $[\cdot]_k$ denotes a parameter corresponding to a variable X_k.

In a special case of NEF-QVF DAG models in (4.2), clearly θ_D^* is the same as the true parameters θ where $\theta_{jk} \neq 0$ for all $k \in \text{pa}(j)$. However θ_D^* is in general not the same as the true parameters.

Lemma 4.1. *Consider a DAG model $(G, \mathbb{P})$. For any node $j \in V$ and $k \in pa(j)$, if*

$$Cov(X_j, X_k) \neq Cov(X_k, D'(\langle [\theta_D^*]_{pa(j)\setminus k}, X_{pa(j)\setminus j} \rangle)),$$

the true parents of each node is equivalent to the estimated parents of each node via surrogate GLM. In other words, $\widetilde{pa}(j) = pa(j)$ for any $j \in V$.

If X_k and $X_{\text{pa}(j)\setminus k}$ are independent, the condition in Lemma 4.1 is equivalent to $\text{Cov}(X_j, X_k) \neq 0$ which is milder than the widely held faithfulness assumption. However, since it is possible that parents are correlated, we require that the covariance between X_j and its parent X_k is not a covariance between the parents X_k and $D'(\langle[\theta^*_D]_{\text{pa}(j)\setminus k}, X_{\text{pa}(j)\setminus j}\rangle)$.

Lemma 4.1 explains that recovering the structure of a DAG is equivalent to solving the p-surrogate GLMLasso if the solution of GLMLasso is sufficiently close to the solution of GLM. Hence in Section 4.3.2.3, we provide the theoretical guarantee that solution of GLMLasso is sufficiently close to the solution θ^*_D of GLM.

For the moralized graph estimation, we also require a similar condition to ensure that the true neighborhood of each node are same as the estimated neighborhood via surrogate GLMLasso. For the precise definition of the estimated neighborhood via surrogate GLMLasso, we define $\theta^*_M \in \Theta_M$ as the solution of the convex optimization problem of GLM where $\Theta_M = \{\theta \in \mathbb{R}^{p-1} : \theta_k = 0 \text{ for } k \notin \mathcal{N}(j)\}$.

$$\theta^*_M := \arg\min_{\theta \in \mathbb{R}^{p-1}} \mathbb{E}\left[-X_j \langle \theta_{V\setminus j}, X_{V\setminus j}\rangle + D(\langle \theta_{V\setminus j}, X_{V\setminus j}\rangle)\right]. \tag{4.7}$$

where $D(\cdot)$ is the log-normalization constant determined by a chosen GLM. Then the estimated neighborhood via surrogate GLM is defined as $\widetilde{\mathcal{N}}(j) := \{k \in V \setminus \{j\} : [\theta^*_M]_k \neq 0\}$.

Corollary 4.1. *Consider a DAG model $(G, \mathbb{P})$. For any node $j \in V$ and $k \in \mathcal{N}(j)$, if*

$$Cov(X_j, X_k) \neq Cov(X_k, D'(\langle [\theta^*_M]_{\mathcal{N}(j)\setminus k}, X_{\mathcal{N}(j)\setminus j}\rangle)),$$

the true neighborhood of each node is equivalent to the estimated neighborhood of each node via surrogate GLM. In other words, $\widetilde{\mathcal{N}}(j) = \mathcal{N}(j)$ for all $j \in V$.

This Corollary 4.1 guarantees that recovering the moralized graph structure is equivalent to solving the p-surrogate GLMLasso if the solution of GLMLasso is sufficiently close to the solution of GLM. Hence in Section 4.3.2.1, we provide the

theoretical guarantee that solution of GLMLasso is sufficiently close to the solution θ^*_M of the GLM.

4.3.2.1 Step 1): Recovery of the Moralized Graph of a DAG via Surrogate GLMLasso

We first focus on Step 1) of the generalized ODS algorithm; theoretical guarantee of recovering the moralized graph of a DAG. We approach this problem via neighborhood estimation where we estimate the neighborhood of each node $\widehat{\mathcal{N}}(j)$ individually. Here we consider surrogate GLMLasso to estimate the neighborhood of each node because a conditional distribution of a node given the rest of nodes in a DAG is in general not equivalent to the likelihood of GLM, therefore our problem is not same as the regular GLMLasso but surrogate GLMLasso.

We assume that there are n iid samples $x = \{X^{(1)}, X^{(2)}, \cdots, X^{(n)}\}$ and for any $i \in \{1, 2, \cdots, n\}$, $X^{(i)} = \{X_1^{(i)}, X_2^{(i)} \cdots, X_p^{(i)}\}$ from a given DAG model $(G, \mathbb{P})$. Then for any variable X_j, the negative surrogate conditional log-likelihood of GLM is as follows.

$$\ell(\theta; x) := \frac{1}{n} \sum_{i=1}^{n} \left(-X_j^{(i)} \langle \theta, X_{V \setminus j}^{(i)} \rangle + D(\langle \theta, X_{V \setminus j}^{(i)} \rangle) \right) \tag{4.8}$$

where $\theta \in \mathbb{R}^{p-1}$ and $D(\cdot)$ is the log-normalization constant determined by a chosen GLM.

We solve the negative surrogate conditional log-likelihood with ℓ_1 norm penalty for each variable X_j:

$$\hat{\theta}_M := \arg \min_{\theta \in \mathbb{R}^{p-1}} \ell(\theta; x) + \lambda_n \|\theta\|_1. \tag{4.9}$$

With the solution $\hat{\theta}_M$, we estimate the neighborhood of a node j, $\widehat{\mathcal{N}}(j) := \{k \in V \setminus j : [\hat{\theta}_M]_k \neq 0\}$ where $[\cdot]_k$ is a value corresponding to a variable X_k. Recall that $\widetilde{\mathcal{N}}(j) = \mathcal{N}(j)$ under the assumption that $[\theta^*_M]_k$ is non-zero for any $k \in \mathcal{N}(j)$. Hence if the solution of surrogate GLMLasso for each variable $\hat{\theta}_M$ is sufficiently close to the solution of GLM θ^*_M in (4.7), we can conclude that $\widehat{\mathcal{N}}(j) = \mathcal{N}(j)$. In the following we show the theoretical guarantee that the solution of surrogate GLMLasso for each

variable is close to the solution of GLM θ^*_M in (4.7).

We begin by discussing the assumptions we impose on the graphical model which are also used in learning graphical models [77, 42, 76, 57]. Since Steps 1) and 3) require similar assumptions, for simplicity, let Q be the Hessian matrix of the negative surrogate conditional log-likelihood of a variable X_j given either the rest of the nodes (4.8) or the nodes before j in the causal ordering (4.10) we discuss later in Section 4.3.2.3. Furthermore, let $\S = \mathcal{N}(j)$ or $\mathrm{pa}(j)$, and $Q_{\S\S}$ be the sub-matrix of Q corresponding to variables $X_\S$.

Assumption 4.2 (Dependency condition)**.** *There exists a constant* $\lambda_{\min} > 0$ *such that* $\lambda_{\min}(Q_{\S\S}) \geq \lambda_{\min}$*. Moreover, there exists a constant* $\lambda_{\max} < \infty$ *such that* $\lambda_{\max}(\frac{1}{n}\sum_{i=1}^n X_\S^{(i)}(X_\S^{(i)})^T) \leq \lambda_{\max}$ *where* $\lambda_{\min}(A)$ *and* $\lambda_{\max}(A)$ *are the smallest and biggest eigenvalues of a matrix* A*, respectively.*

These condition can be interpreted as ensuring that the relevant variables are not overly dependent.

Assumption 4.3 (Incoherence condition)**.** *There exists a constant* $\alpha \in (0, 1]$ *such that*

$$\max_{t \in \S^c} \|Q_{t\S}(Q_{\S\S})^{-1}\|_1 \leq 1 - \alpha.$$

Incoherence condition can be understood that the large number of irrelevant variables cannot strongly affect neighboring variables.

One of the main assumptions of Ising, multinomial, linear, and generalized linear models in learning undirected graphical models [77, 42, 36, 57] is that random variables $(X_1, X_2, \cdots, X_p)$ are bounded with high probability. We also need a similar condition to control a tail behavior of a distribution of each node.

Assumption 4.4 (Concentration Bound condition)**.** *For any node* $j \in V$*, there exists a constant* $M > 0$ *such that* $\mathbb{E}(exp(|X_j|)) < M$*.*

Assumption 4.4 enables surrogate GLMLasso to recover the structures of the moralized graph and directed graph in high-dimensional settings. In addition, it reduces the number of overdispersion scores to be calculated in Step 2) by controlling the cardinality of a condition set. Assumption 4.4 is stronger than other relevant assumptions in learning undirected graphical models [77, 42, 36, 57] because we use surrogate GLMLasso.

We need an assumption on the log-partition constants $D(\cdot)$ similar to leaning undirected graphical models with exponential family distributions via GLMLasso [77].

Assumption 4.5. *The log-partition function $D(\cdot)$ of the likelihood function (4.8) or (4.10) holds the following condition. There exist constants κ_1 and κ_2 such that $\max\{|D'(a)|, |D'''(a)|\} \leq n^{\kappa_2}$ for $a \in [0, \kappa_1 \log(\max\{n, p\}))$, $\kappa_1 \geq 8\max(\|\theta^*_M\|_1, \|\theta^*_D\|_1)$ and $\kappa_2 \in [0, 1/4]$ where $D'(\cdot)$ is the first derivative of $D(\cdot)$ and $D'''(\cdot)$ is the third derivative of $D(\cdot)$.*

Our assumption is a stronger version of the assumption on the log-partition function in [77] because the assumed graph is a directed graph rather than a undirected graph and learning the moralized graph via surrogated GLMLasso is a more difficult than learning undirected graphical models via standard GLMLasso. However we can find exponential family distributions satisfying this assumption. For example Poisson distribution has one of the steepest log-partition function; $D(\cdot) = \exp(\cdot)$. Hence, in order to satisfy Assumption 4.5, we require $\|\theta^*_M\|_1 \leq \frac{\log n}{64 \log p}$ with $\kappa_2 = \frac{1}{8}$. For other distributions such as Binomial, Multinomial, or Gaussian, Assumption 4.5 is satisfied with $\kappa_2 = 0$ because the log-partition function $D(\cdot)$ is bounded.

Putting Assumptions 4.2 4.3, 4.4, and 4.5 together, we reach the following main result that surrogate GLMLasso can recover the moralized graph in high dimensional settings.

Theorem 4.6 (Learning the moralized graph structure via surrogate GLMLasso)**.** *Consider a QVF DAG model $(G, \mathbb{P})$ with the maximum degree of the moralized graph*

d. Suppose that Assumptions 4.2, 4.3, 4.4 and 4.5 are satisfied. Choose the regularization parameter $\frac{(9\log(\max\{n,p\})^2}{n^a} \leq \lambda_n \leq \frac{\lambda_{\min}^2}{30n^{\kappa_2}\log(\max\{n,p\})d\lambda_{\max}}$ *for some* $a \in (\kappa_2, 1/2)$ *where* $\lambda_{\min}, \lambda_{\max}$ *are the minimum and maximum eigenvalue of the Hessian matrix in Assumption 4.2 and* κ_2 *is a constant in Assumptions 4.5. If* $\min_{j\in V}\min_{t\in\mathcal{N}(j)}|[\theta_M^*]_t| \geq \frac{10}{\lambda_{\min}}\sqrt{d}\lambda_n$*, for any constant* $\epsilon > 0$ *there exists a positive constant* C_ϵ *such that for sample size* $n \geq C_\epsilon(d\log(\max\{n,p\})^3)^{\frac{1}{a-\kappa_2}}$*,*

$$P(\widehat{G^m} = G^m) \geq 1 - \epsilon.$$

We defer the proof to Appendix B.1.3. The key technique of the proof is *primal-dual witness* method. Theorem 4.6 shows that surrogate GLMLasso recovers the structure of the moralized graph in high-dimensional ($p > n$) settings with high probability.

Compared to learning undirected graphical models with exponential family distributions via standard GLMLasso, the learning moralized graph requires stronger assumptions and more samples. Yang et al. [77] proved that the require sample size for learning undirected graphical models with exponential family distributions is $n = \Omega(\{d^2\log(\max\{n,p\})^3\}^{\frac{1}{1-3\kappa_2}})$. This makes sense because we apply the surrogate GLMLasso while Yang et al. [77] used the standard GLMLasso.

4.3.2.2 Step 2): Recovery of the Causal Ordering of a DAG

We show theoretical guarantee of recovering the causal ordering of a DAG via our generalized ODS algorithm under suitable regularity conditions. We begin by stating assumptions we impose on the graphical model.

Assumption 4.7. *For all* $j \in V$*,* $K \subset pa(j)$ *and all* $S \subset V \setminus K$*, there exists an* $M_0 > 0$ *such that*

$$Var(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_S) > M_0.$$

This assumption is a stronger version of the identifiability assumption in Theorem 4.1, $\mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_S) > 0$. Since we are in the finite sample setting, we need a lower bound away from 0 for all overdispersion scores.

The concentration bound, Assumption 4.4 is also important because the overdispersion score is sensitive to bias of conditional mean and variance of each variable, and therefore the overdispersion score is sensitive to both the size of a condition set and cardinality of each variable. Therefore by controlling the tail behavior of each random variable, we reduce the total number of overdispersion scores to be calculated in Step 2).

We present the theoretical result given the true moralized graph. Recall that for general DAGs, the true causal ordering π^* may not be unique. Therefore, let $\mathcal{E}(\pi^*)$ denote the set of all the causal orderings that are consistent with the true DAG G^*.

Theorem 4.8 (Recovery of the causal ordering of a QVF DAG)**.** *Consider a QVF DAG model $(G, \mathbb{P})$ with quadratic variance coefficients (β_0, β_1) and the maximum degree of the moralized graph d. Suppose that $\beta_1 > -1$ and the moralized graph G^m is known. Furthermore, suppose that Assumption 4.4 and 4.7 are satisfied. Then for any $\epsilon > 0$ and some $c_0 \geq (\log(\max\{n, p\}))^d$, there exists a positive constant K_ϵ such that for sample size $n \geq K_\epsilon(\log(\max\{n, p\}))^{5+d}$,*

$$P(\widehat{\pi} \in \mathcal{E}(\pi^*)) \geq 1 - \epsilon.$$

The detail of the proof is provided in Appendix B.1.4. The main idea of the proof is the overdispersion property exploited in Theorem 4.1. Note that estimated overdispersion scores converge to the true overdispersion scores $\widehat{\mathcal{S}}(j, k) \to \mathcal{S}(j, k)$ as sample size increases because each entry of a overdispersion score is the difference between sample conditional sample mean and variance which consistently converge to true values, respectively. Hence a comparison of overdispersion scores enable us to detect the parents of each node in limited data settings.

Theorem 4.8 claims that if the triple (n, d, p) satisfies $n = \Omega((\log p)^{5+d})$ and $d < \Omega(p)$, then our generalized ODS algorithm correctly estimates the true causal ordering. Therefore if the moralized graph is sparse, our generalized ODS algorithm recovers the true casual ordering in high-dimensional $(p > n)$ settings. DAG learning

algorithms that apply to high-dimensional setting are not common since they typically rely on the faithfulness [68], or other restrictive conditions that are often not satisfied in high-dimensional settings. Note that if the moralized graph is not sparse and $d = \Omega(p)$, the generalized ODS algorithm fails to work in high-dimensional settings. This also makes sense since if the number of neighbors of each node is large, we would need to condition on a large set of variables which is very sample-intensive.

Our sample complexity is certainly not optimal since a sample cut-off parameter c_0 is chosen for the worst case which is $\log(\max\{n, p\})^{-d}$. In addition, the power term of the sample complexity $n = \Omega((\log p)^{5+d})$ is associated with Assumption 4.4. If we have a stronger assumption $\max_j \mathbb{E}(\exp(4X_j)) < M$, it can be reduced to $n = \Omega((\log p)^{2+d})$. Determining an optimal sample complexity remains an open question.

4.3.2.3 Step 3): Recovery of the Structure of a DAG via Surrogate GLM-Lasso

In this section, we focus on Step 3) of our generalized ODS algorithm; theoretical guarantees of recovering the structure of a DAG given its causal ordering. Our approach in Step 3) is the same as in Step 1) except that we estimate the parents of each node over the possible parents according to the causal ordering. Without loss of generality, assume that the true causal ordering is $\pi^* = (1, 2, \cdots, p)$. Then, we estimate the parents of a node j over the set of nodes $\{1, 2, \cdots, j-1\}$.

Again we consider the surrogate GLMLasso for estimating the parents of each node because a conditional distribution of a node given its parents in a DAG may not correspond to the likelihood of GLM, therefore our problem is not the regular GLMLasso but surrogate GLMLasso like Step 1). For notational convenience, we use $X_{1:j} = (X_1, X_2, \cdots, X_j)$. Then for any variable X_j, the negative surrogate conditional log-likelihood of GLM is as follows.

$$\ell_D(\theta; x) := \frac{1}{n} \sum_{i=1}^{n} \left(-X_j^{(i)} \langle \theta, X_{1:j-1}^{(i)} \rangle + D(\langle \theta, X_{1:j-1}^{(i)} \rangle) \right) \tag{4.10}$$

where $\theta \in \mathbb{R}^{j-1}$ and $D(\cdot)$ is the log-normalization constant determined by a chosen

GLM.

We solve the negative surrogate conditional log-likelihood with ℓ_1 norm penalty for each variable X_j:

$$\hat{\theta}_D := \arg\min_{\theta \in \mathbb{R}^{j-1}} \ell_D(\theta; x) + \lambda_n \|\theta\|_1. \tag{4.11}$$

With the solution $\hat{\theta}_D$, we estimate the parents of a node j, $\widehat{\text{pa}}(j) = \{k \in V \setminus j : [\hat{\theta}_D]_k \neq 0\}$ where $[\cdot]_k$ is a value corresponding to a variable X_k. Recall that $\widehat{\text{pa}}(j) = \text{pa}(j)$ under the assumption that $[\theta^*_M]_k$ is non-zero for all $k \in \mathcal{N}(j)$. Hence if the solution of surrogate GLMLasso for each variable $\hat{\theta}_D$ and the solution of GLM θ^*_D in (4.6) are close, we can conclude that $\widehat{\mathcal{N}}(j) = \mathcal{N}(j)$. As in the moralized graph learning, it is sufficient to show that the solution of surrogate GLMLasso $\hat{\theta}_D$ is close to the solution of GLM θ^*_D.

Theorem 4.9 (Learning the structure of a DAG via surrogate GLMLasso)**.** *Consider a QVF DAG model $(G, \mathbb{P})$ with the maximum degree of the moralized graph d. Suppose that the true causal ordering is known. Furthermore, suppose that Assumptions 4.2, 4.3, 4.4 and 4.5 are satisfied. Choose the regularization parameter $\frac{(9\log(\max\{n,p\}))^2}{n^a} \leq \lambda_n \leq \frac{\lambda^2_{\min}}{30n^{\kappa_2}\log(\max\{n,p\})d\lambda_{\max}}$ for some $a \in (\kappa_2, 1/2)$ where $\lambda_{\min}, \lambda_{\max}$ are the minimum and maximum eigenvalue of the Hessian matrix in Assumption 4.2 and κ_2 is a constant in Assumptions 4.5. If $\min_{j\in V} \min_{t\in\mathcal{N}(j)} |[\theta^*_D]_t| \geq \frac{10}{\lambda_{\min}}\sqrt{d}\lambda_n$, for any constant $\epsilon > 0$ there exists a positive constant C_ϵ such that for sample size $n \geq C_\epsilon(d\log(\max\{n,p\})^3)^{\frac{1}{a-\kappa_2}}$,*

$$P(\widehat{G} = G) \geq 1 - \epsilon.$$

The detail of the proof is provided in Appendix B.1.5. The main idea of the proof for Theorem 4.9 is again *primal-dual witness* method. Theorem 4.9 explains that surrogate GLMLasso successfully recovers the structure of a DAG in high-dimensional $(p > n)$ settings given the true causal ordering. As in learning the moralized graph, learning the DAG requires stronger assumptions and more samples compared to learning undirected graphical models with exponential family distributions via standard

GLMLasso [77]. However if a QVF-DAG model consist of NEF-QVF distributions, it requires similar assumptions and sample complexity.

We present the consistency of all three steps of the generalized ODS algorithm. In combination of Theorems 4.6, 4.8, and 4.9, we reach our final main result that the generalized ODS algorithm successfully recovers the true structure of a QVF DAG with high probability even in high-dimensional settings.

Corollary 4.2 (Learning a DAG structure via our generalized ODS algorithm)**.** *Consider a QVF DAG model $(G, \mathbb{P})$ with quadratic variance coefficients (β_0, β_1) and the maximum degree of the moralized graph d. Suppose that κ_2 is a constant in Assumptions 4.5. Under the regularity conditions and if the triple (n, p, d) satisfies $n = \Omega(\max\{(d\log(p)^3)^{\frac{1}{a-\kappa_2}}, (\log p)^{5+d}\})$ for some $a \in (\kappa_2, 1/2)$, then our generalized ODS algorithm recovers the structure of the DAG with high probability.*

4.4 Algorithm for NEF-QVF DAG Models

In this section, we develop a new DAG learning algorithm for NEF-QVF DAG models (4.2), called NEF-QVF ODS algorithm, which is an adapted version of the generalized ODS algorithm 4.1. Like the generalized ODS algorithm, our NEF-QVF ODS algorithm consists of three main steps: Step 1) is to estimate the moralized graph of the DAG, Step 2) is to estimate the causal ordering using overdispersion property, and Step 3) is to estimate the DAG structure. Step 1) and Step 3) can exploit off-the-shelf graph structure learning algorithms (e.g., [71, 77, 3]) as well as neighborhood selection algorithm such as GLMLasso. Step 1) allows us to reduce both computational and sample complexity by exploiting the sparsity of the moralized or undirected graphical model representation of a DAG.

The novelty of this paper is Step 2) of the NEF-QVF ODS algorithm. In general, it is very difficult problem that determining a node is conditionally overdispersed since overdispersion tests require computing the number of occurrences of all different possible patterns of variables of a conditioning set. For example, if a conditioning set

X_S contains five ternary variables then the number of overdispersion test is 3^5. This implies that the number of samples required to accurately estimate the conditional expectation and variance is exponential to the size of the conditioning set and sample space.

Here we introduce an important property of natural exponential family with quadratic variance function (NEF-QVF) which reduces a massive number of estimations of conditional expectation to only one regression problem. More precisely, we estimate the generalized linear model (GLM) (4.12) for estimating conditional expectation. For a node j and a conditioning set $S \subset V \setminus \{j\}$,

$$\theta^* := \arg\min_{\theta \in \mathbb{R}^{|S|}} \{-\langle \theta, X_S \rangle X_j + D_j(\langle \theta, X_S \rangle)\}. \tag{4.12}$$

where $D_j(\cdot)$ is the log-normalization constant determined by a given GLM.

By the first order optimality condition, we obtain $\mathbb{E}[X_j \mid X_S] = D_j'(\langle \theta^*, X_S \rangle)$ where $D_j'(\cdot)$ is the first derivative of $D_j(\cdot)$. This implies that learning the parameters of a GLM is sufficient to estimating conditional expectations for all different possible patterns of variables of a conditioning set.

A conditional variance of a node given its parents is clearly $\mathrm{Var}(X_j \mid X_{\mathrm{pa}(j)}) = \beta_0 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) + \beta_1 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)})^2$. However it is unclear that how a conditional variance is related to a conditional expectation for a general conditioning set $S \subset V$. Hence we provide an important lemma which represents the relationship between a conditional variance and expectation for any conditioning set $S \subset V$.

Lemma 4.2. *Let $(X_1, X_2, \cdots, X_p)$ be a random vector associated with a NEF-QVF DAG model $(G, \mathbb{P})$ with quadratic variance coefficients (β_0, β_1) in (4.1). Suppose that $\beta_1 > -1$ and the identifiability assumption in Theorem 4.1 is satisfied. Then for any $j \in V$, $K \subset pa(j)$, and $S \subset V \setminus (K \cup \{j\})$*

$$\frac{Var(X_j \mid X_S)}{\beta_0 \mathbb{E}(X_j \mid X_S) + \beta_0 \mathbb{E}(X_j \mid X_S)} > 1.$$

Proof.

$$\begin{aligned}\mathrm{Var}(X_j \mid X_S) &\overset{(a)}{=} \mathbb{E}(\mathrm{Var}(X_j \mid X_{\mathrm{pa}(j)}) \mid X_S) + \mathrm{Var}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) \mid X_S) \\ &\overset{(b)}{=} \mathbb{E}(\beta_0 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)} + \beta_1 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)})^2 \mid X_S) + \mathrm{Var}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) \mid X_S) \\ &\overset{(c)}{=} \beta_0 \mathbb{E}(X_j \mid X_S) + \beta_1 \mathbb{E}(X_j \mid X_S)^2 + (1+\beta_1)\mathrm{Var}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) \mid X_S).\end{aligned}$$

(a) follows from the variance decomposition formula $\mathrm{Var}[Y] = \mathbb{E}(\mathrm{Var}[Y \mid X]) + \mathrm{Var}(\mathbb{E}[Y \mid X])$ for some random variables X and Y. In addition, (b) follows from the variance quadratic property (4.1) and (c) is directly from the definition of a conditional variance.

Therefore, we have

$$\frac{\mathrm{Var}(X_j \mid X_S)}{\beta_0 \mathbb{E}(X_j \mid X_S) + \beta_1 \mathbb{E}(X_j \mid X_S)^2} = 1 + (1+\beta_1)\mathrm{Var}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) \mid X_S) > 1. \quad (4.13)$$

□

Lemma 4.2 claims that if a condition set S contains all parents then the ratio a conditional variance to the quadratic function of a conditional expectation is one, otherwise greater than one. This implies that a new random variable $Y_j = \frac{X_j}{\sqrt{\beta_0 \mathbb{E}(X_j|X_S) + \beta_1 \mathbb{E}(X_j|X_S)^2}}$ have a conditional variance one if a conditioning set S contains all parents, otherwise greater than one. Hence testing whether variance of Y_j is equal to one is equivalent to testing whether a conditioning set includes all parents of a node j. We will use the conditional variance of transformed variable Y_j as a overdispersion score in Step 2).

In a finite data setting, we assume that there are n iid samples drawn from a given QVF DAG model which is referred to as $\{X^{(1)}, X^{(2)}, \cdots, X^{(n)}\}$ where $X^{(i)} = \{X_1^{(i)}, X_2^{(i)} \cdots, X_p^{(i)}\}$ for all $i \in \{1, 2, \cdots, n\}$. Then, we estimate the GLM model (4.12) for overdispersion scores in Step 2). We find the minimizer of the negative conditional log-likelihood of GLM for a node j given a conditioning set $S \subset V \setminus \{j\}$:

$$\hat{\theta} = \arg\min_{\theta \in \mathbb{R}^{|S|}} \frac{1}{n} \sum_{i=1}^{n} \left\{ -\langle \theta, X_S^{(i)} \rangle X_j^{(i)} + D_j(\langle \theta, X_S^{(i)} \rangle) \right\}. \quad (4.14)$$

where $D_j(\cdot)$ is the log-normalization constant determined by a given GLM.

Then, we estimate a conditional expectation based on the estimated parameters in (4.14) and variance quadratic equation in (4.1). For any $i \in \{1, 2, \cdots, n\}$ and $j \in V$,

$$\widehat{\mathbb{E}}(X_j^{(i)} \mid X_S^{(i)}) \quad := \quad D_j(\langle \hat{\theta}, X_S^{(i)} \rangle). \tag{4.15}$$

For the marginal expectation of each variable, we use sample mean as an estimator.

As we discussed we will use the sample variance of the ratio the conditional expectation to conditional standard deviation as an overdispersion score. Then the true overdispersion score for each component of the causal ordering must be one, however other scores should be greater than one by Lemma 4.2. To be precise, we define overdispersion scores as following: For any $j, k \in V$,

$$\widehat{S}(1,k) \quad := \quad \frac{1}{n-1} \sum_{i=1}^{n} \frac{(X_k^{(i)} - \widehat{\mathbb{E}}(X_k))^2}{\beta_0 \widehat{\mathbb{E}}(X_k) + \beta_1 \widehat{\mathbb{E}}(X_k)^2} \tag{4.16}$$

$$\widehat{S}(j,k) \quad := \quad \frac{1}{n-1} \sum_{i=1}^{n} \frac{(X_k^{(i)} - \widehat{\mathbb{E}}(X_k^{(i)} \mid X_{\widehat{C}_{jk}}^{(i)}))^2}{\beta_0 \widehat{\mathbb{E}}(X_j^{(i)} \mid X_{\widehat{C}_{jk}}^{(i)}) + \beta_1 \widehat{\mathbb{E}}(X_j^{(i)} \mid X_{\widehat{C}_{jk}}^{(i)})^2} \tag{4.17}$$

where $\widehat{C}_{jk}$ is an estimated candidate parents set which is an intersection of estimated neighbors of $j-1^{th}$ component of an estimated causal ordering and first $j-1$ components of an estimated causal ordering because only first $j-1$ components of a causal ordering can be parents of j^{th} component of a causal ordering, and the set of neighbors of $j-1^{th}$ component of an estimated causal ordering includes j^{th} component of a causal ordering. A candidate parents set is estimated in Step 2) of NEF-QVF ODS algorithm.

As Park and Raskutti [46] explained, the main purpose of Step 1) is to reduce both computational complexity and sample complexity by using the sparsity of the moralized graph. The moralized graph provides a candidate parents set for each node. Using a candidate parents set reduces the number of variables of a GLM (4.14) to be fitted for Step 2), and therefore it improves our algorithm in terms of prediction and

Algorithm 4.2 NEF-QVF ODS algorithm

Input: iid n samples from the DAG model
Output: A causal ordering $\widehat{\pi} \in \mathbb{N}^p$ and a graph structure $\widehat{E} \in \{0,1\}^{p \times p}$
Step 1: Estimate the neighborhood of each node $\widehat{\mathcal{N}}(j)$ in the moralized graph
Step 2: Estimate the causal ordering using overdispersion scores
for $k \in \{1, 2, \cdots, p\}$ **do**
    Calculate overdispersion scores $\widehat{\mathcal{S}}(1, k)$ using Equation (4.16)
end for
The first element of a causal ordering $\widehat{\pi}_1 := \arg\min_k \widehat{\mathcal{S}}(1, k)$
for $j = \{2, 3, \cdots, p-1\}$ **do**
    for $k \in \widehat{\mathcal{N}}(\widehat{\pi}_{j-1}) \cap \{1, 2, \cdots, p\} \setminus \{\widehat{\pi}_1, \cdots, \widehat{\pi}_{j-1}\}$ **do**
        Find candidate parents set $\widehat{C}_{jk} := \widehat{\mathcal{N}}(k) \cap \{\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_{j-1}\}$
        Calculate overdispersion scores $\widehat{\mathcal{S}}(j, k)$ using Equation (4.17)
    end for
    The j^{th} element of a causal ordering $\widehat{\pi}_j := \arg\min_k \widehat{\mathcal{S}}(j, k)$
    Step 3: Estimate the directed edges toward $\widehat{\pi}_j$, denoted by $\widehat{D}_j$
end for
The last element of the causal ordering $\widehat{\pi}_p := \{1, 2, \cdots, p\} \setminus \{\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_{p-1}\}$
The directed edges toward $\widehat{\pi}_p$, denoted by $\widehat{D}_p := \{(z, \widehat{\pi}_p) \mid z \in \widehat{\mathcal{N}}(\widehat{\pi}_p)\}$
Return: $\widehat{\pi} := (\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_p)$ and $\widehat{E} := \cup_{j=\{2,3,\cdots,p\}} \widehat{D}_j$

computation. Furthermore, Step 1) reduces the number of overdispersion scores to be compared in Step 2). Since the edge set of the moralized graph includes the edge set of a DAG, j^{th} component of the causal ordering is a neighbor of $(j-1)^{th}$ component of the causal ordering. Therefore, we only compare overdispersion scores of neighbors of $(j-1)^{th}$ component of the causal ordering. By minimizing the number of overdispersion scores to be compared, we also obtain significant statistical and computational improvements. A similar step is taken by Loh et al. [40], the MMHC [71] and SC algorithms [21]. Since the moralized graph is an undirected graph, a number of choices are available for Step 1) including standard undirected graph learning algorithms such as the HITON [3] and MMPC algorithms [71] as well as GLMLasso [18]. In addition, standard DAG learning algorithms such as GES [9] and MMHC algorithms [71] can

be applied and the moralized graph can be found from the estimated DAG.

The novelty of our algorithm is Step 2) which estimates the causal ordering. The main idea is to determine which nodes are conditionally overdispersed. The causal ordering is determined one node at a time by selecting the node with the minimum overdispersion score which is representative of a node that is least likely to be overdispersed. The main difference between our NEF-QVF ODS algorithm and the ODS algorithm is overdispersion scores. The ODS algorithm uses the weighted average of sample conditional variance minus conditional expectation as a overdispersion score, and therefore the score is sensitive to the number of patterns of a conditioning set. In contrast, our NEF-QVF ODS algorithm calculate a overdispersion score based on based on the estimated parameter in the given GLM. Therefore, the NEF-QVF ODS algorithm is favorable to the ODS algorithm in limited data settings, and our algorithm can be applied to continuous distribution such as Gamma.

Step 3) which recovers the set of parents of each node j is reduced to selecting the parents out of all elements before a node j in the estimated causal ordering from Step 2). Therefore, Step 3) can be reduced to p-neighborhood estimation problems which can be performed using GLMLasso [18] as well as standard DAG learning algorithms such as the PC [68], GES [9], and MMHC algorithms [71].

4.5 Numerical Experiments

4.5.1 The Generalized ODS Algorithm

In this section, we support our theoretical results with numerical experiments and show that our generalized ODS algorithm performs favorably compared to state-of-the-art DAG learning algorithms. In order to authenticate the validation of Theorems 4.6, 4.8, and 4.9, the simulation study was conducted using 50 realizations of a p-node random Poisson and Binomial GLM DAG models in (4.2) where a conditional distribution of each node given its parents is Poisson and Binomial, respectively. In all the results we present non-zero parameters (θ_{jk}) in (4.2) were chosen uniformly at

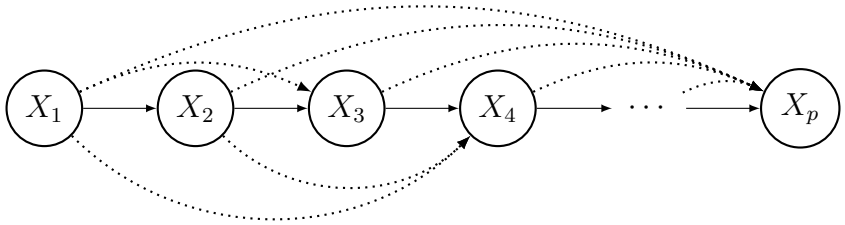


Figure 4.2:: Structure of the DAG we used in numerical experiments. Solid directed edges are always present and dotted directed edges are randomly chosen based on the given number of parents of each node constraints

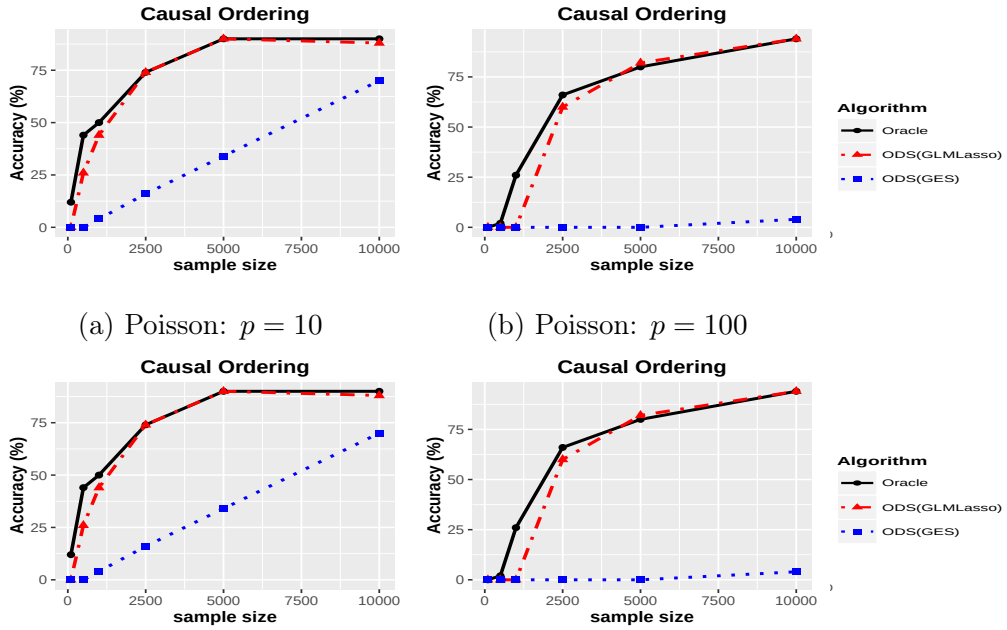
random in the range $\theta_{jk} \in [-1, -0.5]$ for Poisson DAG models and $\theta_{jk} \in [0.5, 1]$ for Binomial DAG models. These ranges of parameters are chosen to satisfy the assumptions of the generalized ODS algorithm although there is no restriction on parameters of DAG models unlike undirected graphical models. In addition, we fixed parameters $N_1, N_2, \cdots, N_p = 4$ for Binomial DAG models. We also used a special structure (see Figure 4.2) which has the fixed unique causal ordering $\pi^* = (1, 2, \cdots, p)$ with edges randomly generated while respecting the desired maximum number of parents constraints for the DAG. In our experiments, we always set the number of parents to two (the number of neighbors of each node is at least three, and therefore $d \in [3, p-1]$) and the thresholding constant to $c_0 = 0.005$ although any value below 0.01 seems to work well.

In Figure 4.3, we plot the proportion of simulations in which our generalized ODS algorithm recovers the correct causal ordering to validate Theorem 4.8. We plot the accuracy rates in recovering the true causal ordering π^* as a function of sample size ($n \in \{100, 500, 1000, 2500, 5000, 10000\}$) for different node sizes ($p = 10$ for (a) and (c), and $p = 100$ for (b) and (d)) and different distributions (Poisson for (a) and (b) and Binomial for (c) and (d)). In each sub-figure, two generalized ODS algorithms are used; (i) GLMLasso [18] is applied in Step 1) where we chose a regularization parameter $\lambda = \frac{0.75}{\log(\max(n,p))}$ for Poisson DAG models and $\lambda = \frac{.10}{\log(\max(n,p))}$ for Binomial

DAG models, and (ii) the GES algorithm [9] is applied in Step 1) where we used the mBDe [29] (modified Bayesian Dirichlet equivalent) score and then the moralized graph is generated based on the output of the GES algorithm. As we discussed any state-of-the-art algorithms can be applied, we chose the those two algorithms because they seem to work better in terms of recovering moralized graph in our simulation settings. We also showed an oracle where the undirected edges of the true moralized graph is used for comparison.

Figure 4.3 shows that both generalized ODS algorithms recover the true causal ordering better as sample size increases, which supports our theoretical result. In addition, we can see that the GLMLasso-base generalized ODS algorithm seems to be better than the GES-base generalized ODS algorithm in terms of the recovery of the causal ordering. Since GLMLasso is the only algorithm that scale to the setting ($p > 1000$), we used GLMLasso in Steps 1) and 3) of the generalized ODS algorithm for large-scale DAG models.

Figures 4.4 and 4.5 provide a comparison of how accurately the generalized ODS algorithm performs in terms of Hamming distance to two state-of-the-art directed graphical model learning algorithms (the MMHC and GES algorithms) for both Poisson DAG and Binomial DAG models. Similar to learning causal ordering, we used two generalized ODS algorithms exploiting GLMLasso in both Steps 1) and 3) and the GES algorithm with the mBDe score in both Steps 1) and 3). Furthermore, oracle where the undirected edges of the true moralized graph is used for comparison. We considered small-scale DAG models with $p = 10$ in sub-figures (a), (b), (e) and (f), and $p = 100$ in sub-figures (c), (d), (g) and (h). Then, we considered two Hamming distance measures. We measured the Hamming distance to the skeleton of the true DAG in sub-figures (a), (c), (e) and (g) which is the set of edges of the DAG without directions. In addition, we measured the Hamming distance for the edges with directions in sub-figures (b), (d), (f), and (h). The reason we consider the skeleton is that the comparison algorithms can recover up to the skeleton of the DAG. We



(a) Poisson: $p = 10$ (b) Poisson: $p = 100$

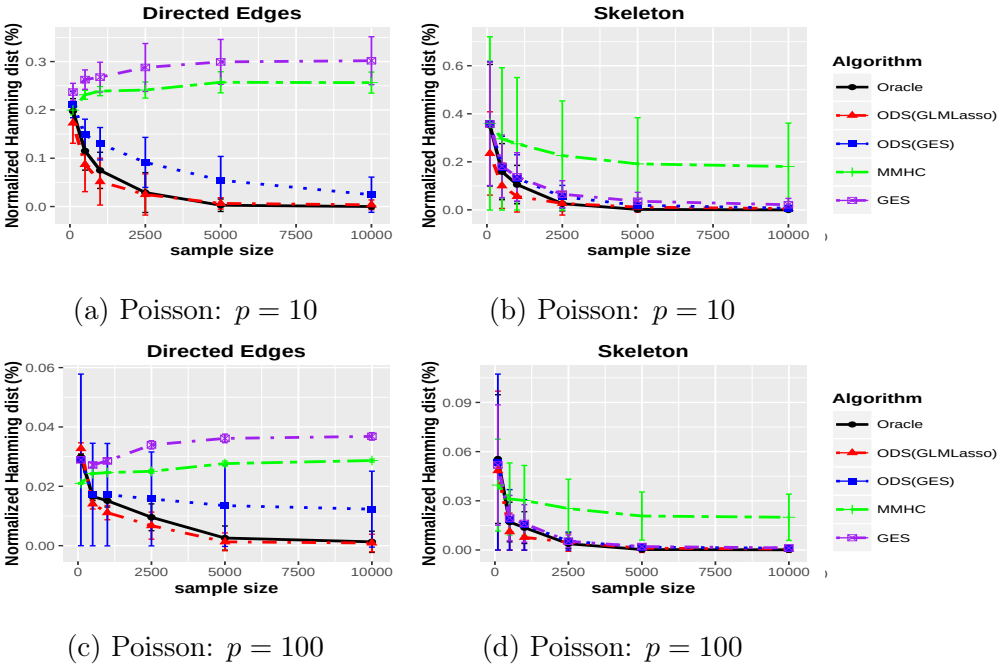
(c) Binomial: $p = 10$ (d) Binomial: $p = 100$

Figure 4.3:: Probability of recovering the causal ordering of a DAG via our generalized ODS algorithm using two different algorithms (GLMLasso and GES algorithm) in Step 1)

normalized the Hamming distances by dividing it by the maximum number of errors $\binom{p}{2}$ and $p(p-1)$, respectively. Therefore, the overall score is a percentage.

As we see in Figures 4.4 and 4.5, the both generalized ODS algorithms significantly out-performs state-of-the-art MMHC and GES algorithms in terms of both directed edges and skeleton. For small sample size cases, the both generalized ODS algorithms shows bad performance because it often fails to recover the causal ordering, however we can see that GES-base generalized ODS algorithm performs always better the GES algorithm. It is because the generalized ODS algorithm only adds directional information to the estimated skeleton via the GES algorithm and hence GES-base generalized ODS algorithm is always better than the GES algorithm in terms of recovering both directed edges and skeleton.

Furthermore Figures 4.4 and 4.5 show that as sample size increases, the both



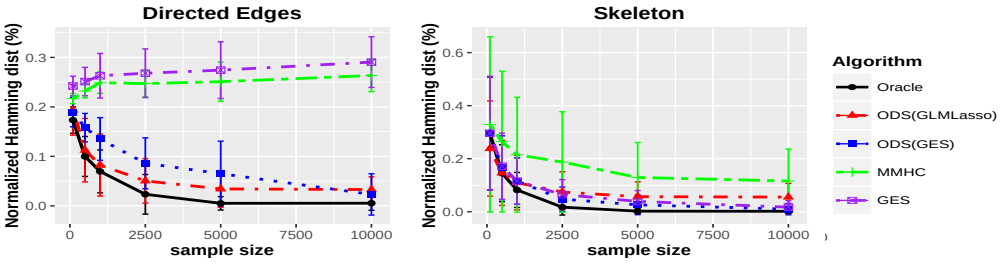
(a) Poisson: $p = 10$ (b) Poisson: $p = 10$

(c) Poisson: $p = 100$ (d) Poisson: $p = 100$

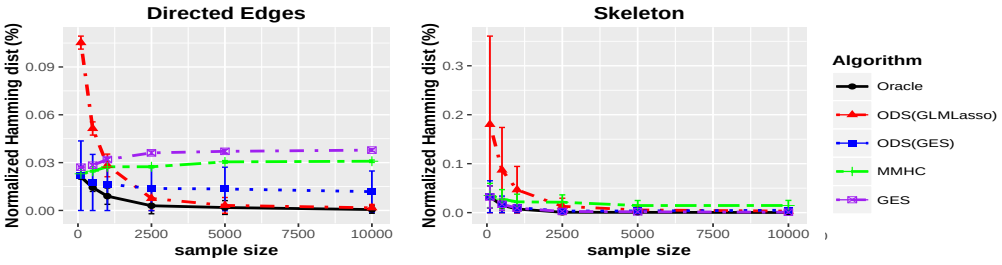
Figure 4.4:: Comparison of the generalized ODS algorithms using GLMLasso (in Steps 1) and 3)) and the GES algorithm (in Steps 1) and 3)) to two state-of-the-art DAG learning algorithms (the MMHC and the GES algorithms) in terms of Hamming distance to skeletons and directed edges of Poisson DAG models. The end of each bar corresponds to the average of the normalized hamming distance plus or minus its standard error

generalized ODS algorithms recover the true directed edges and skeleton of the DAG better, which is consistent with our theoretical results. It must be pointed out that the choice of the DAG models is suited to the generalized ODS algorithm while comparison algorithms are capable of being applied to more general classes of DAG models.

Now we consider the statistical performance for large-scale DAG models to show that the generalized ODS algorithm works in the high-dimensional setting. In all experiments we used the GLMLasso in Steps 1) and 3) of generalized ODS algorithm. Figure 4.6 plots the statistical performance of the generalized ODS algorithm for large-scale Poisson DAGs in sub-figures (a), (b), and (c) and Binomial DAGs in



(a) Binomial: $p = 10$ (b) Binomial: $p = 10$

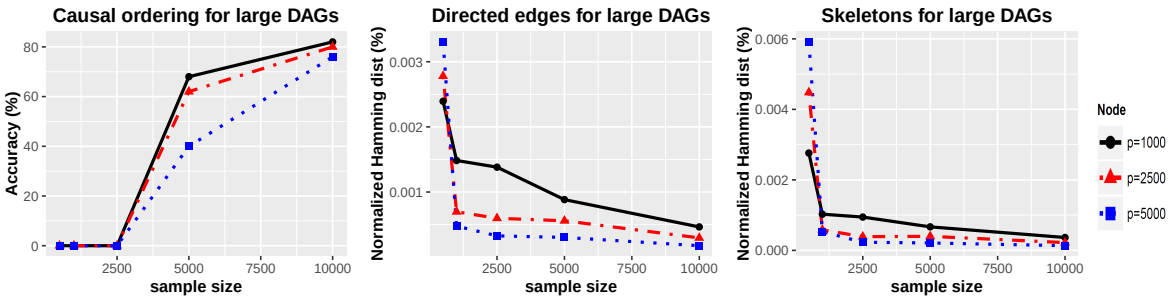


(c) Binomial: $p = 100$ (d) Binomial: $p = 100$

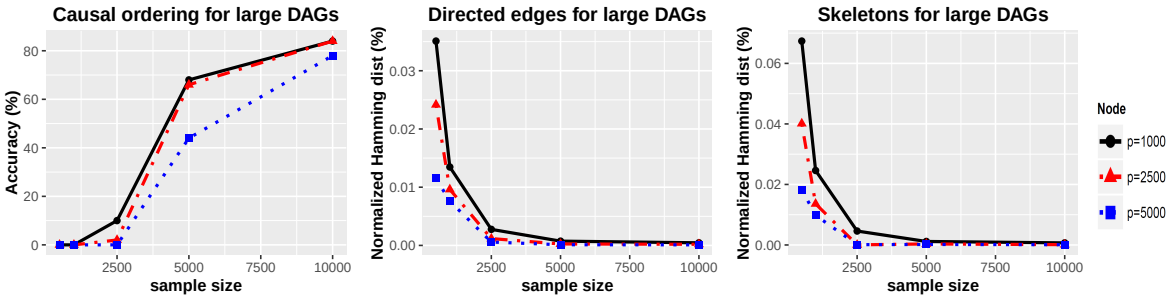
Figure 4.5:: Comparison of the generalized ODS algorithms using GLMLasso (in Steps 1) and 3)) and the GES algorithm (in Steps 1) and 3)) to two state-of-the-art DAG learning algorithms (the MMHC and the GES algorithms) in terms of Hamming distance to skeletons and directed edges of Binomial DAG models. The end of each bar corresponds to the average of the normalized hamming distance plus or minus its standard error

sub-figures (d), (e), and (f). Furthermore, (a) and (d) represent the accuracy rates of the recovering the causal ordering, (b) and (e) show the normalized Hamming distance to the true skeleton, and (c) and (f) show the normalized Hamming distance to the true edge set of the DAG. Accuracies vary as a function of sample size ($n \in \{500, 1000, 2500, 5000, 10000\}$) for each node size ($p = \{1000, 2500, 5000\}$). Figure 4.6 shows that the generalized ODS algorithm recovers the causal ordering and the true structure of a DAG even in high-dimensional settings.

In Figure 4.7, we compared the run-time of the generalized ODS algorithms using GLMLasso in Steps 1) and 3) to the run-time of the MMHC and the GES



(a) Poisson (b) Poisson (c) Poisson

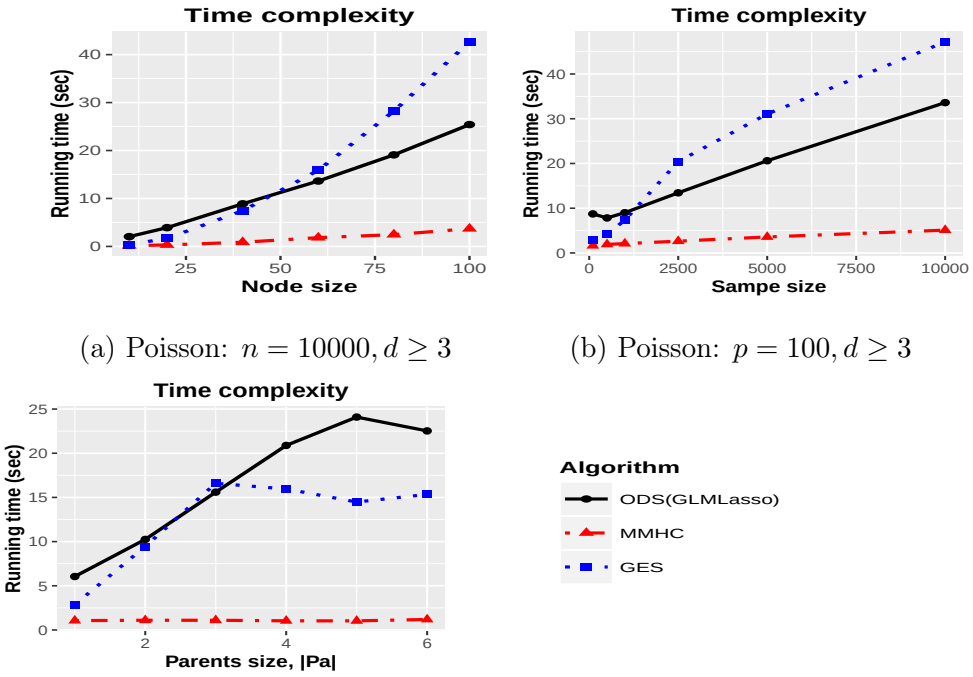


(d) Binomial (e) Binomial (f) Binomial

Figure 4.6:: Performance of the generalized ODS algorithm using GLMLasso in both Steps 1) and 3) for large-scale DAG models with the node size $p = \{1000, 2500, 5000\}$

algorithms. We measured the run-time for Poisson DAG models by varying (a) node size $p \in \{10, 20, 40, 60, 80, 100\}$ with the fixed sample size $n = 10000$ and exactly two parents of each node, (b) sample size $n \in \{100, 500, 1000, 2500, 5000, 10000\}$ with the fixed node size $p = 100$ and two parents of each node, and (c) the number of parents of each node $|\text{Pa}| \in \{1, 2, 3, 4, 5, 6\}$ with the fixed sample size $n = 10000$ and node size $p = 20$. Sub-figures (a) and (b) support the section 4.3.1 where the time complexity of our ODS algorithm using GLMLasso is at least $O(\min(n, p)np^2)$ which is computational complexity of p GLMLasso. Sub-figure (c) also shows run-time of the ODS algorithm is proportional to the number of parents of each node which is the lower bound of the degree of the moralized graph d.

We can also see that the generalized ODS algorithm is faster than the GES algorithm as either node size or sample size increases. Although the generalized ODS



(a) Poisson: $n = 10000, d \geq 3$

(b) Poisson: $p = 100, d \geq 3$

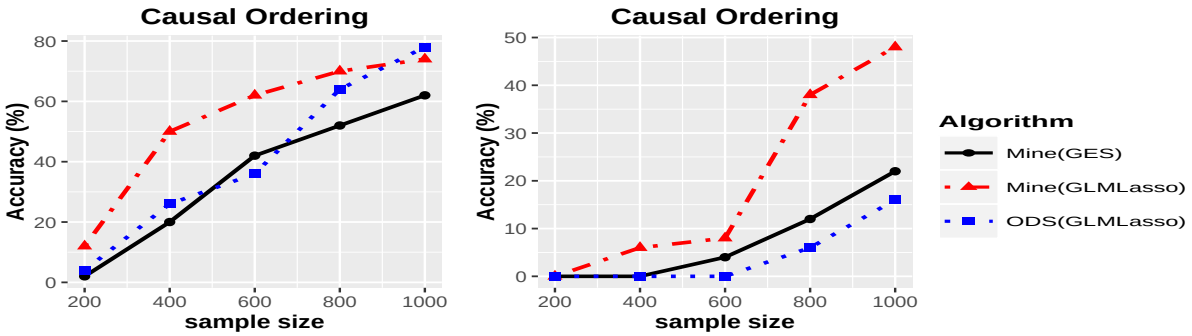
(c) Poisson: $n = 10000, p = 20$

Figure 4.7:: Comparison of the generalized ODS algorithms using GLMLasso in Steps 1) and 3) to two standard DAG learning algorithms (the MMHC and the GES algorithms) in terms of running time with respect to (a) node size p, (b) sample size n, and (c) number of parents of each node

algorithm seems slower than the MMHC algorithm, this is mainly because the MMHC algorithm often stops earlier before they reach the true DAG (see Figure 4.4 and 4.5).

4.5.2 The NEF-QVF Algorithm

In this section, we support our theoretical results with numerical experiments and show that our NEF-QVF algorithm performs better than state-of-the-art DAG learning algorithms and the generalized ODS algorithm 4.1 in terms of recovering structure of DAG. The simulation study was conducted using 50 realizations of a p-node random Poisson NEF-QVF DAG models where a conditional distribution of each node given its parents is Poisson with the canonical link function for count data. Furthermore we used random Exponential NEF-QVF DAG models where a conditional

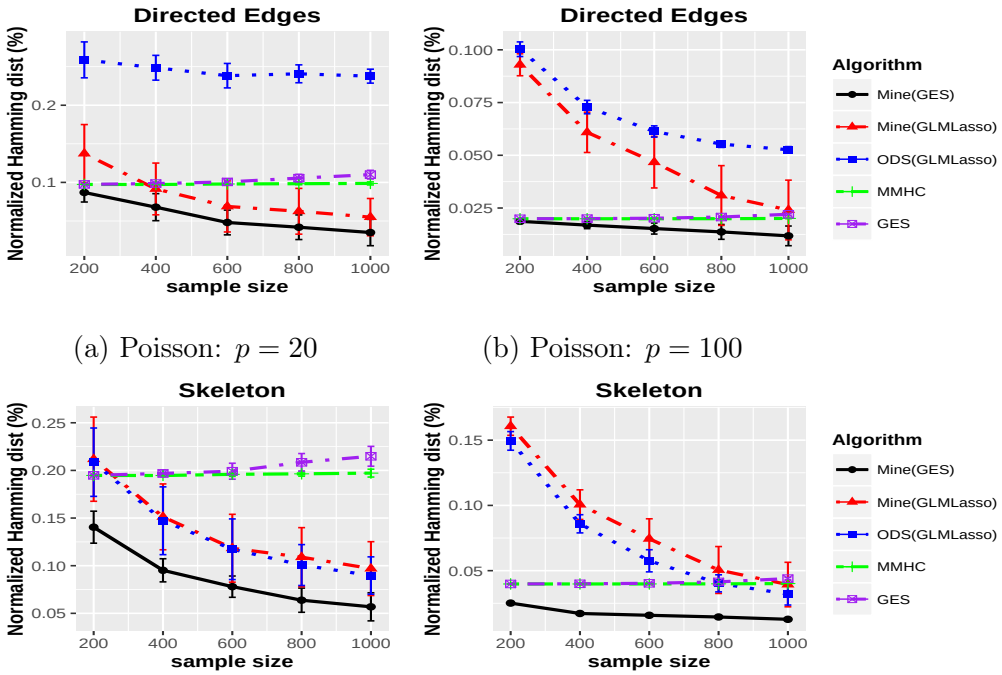


(a) Poisson: $p = 20$ (b) Poisson: $p = 100$

Figure 4.8:: Probability of recovering the causal ordering of a DAG via our generalized ODS algorithm using two different algorithms (GLMLasso and GES algorithm) in Step 1)

distribution of each node given its parents is Exponential with the canonical link function for continuous data. In all the simulation results we present, we used a special structure (see Figure 4.2) which has the fixed unique causal ordering $\pi^* = (1, 2, \cdots, p)$ with edges randomly generated while respecting the desired maximum number of parents constraints for the DAG. We always set the number of parents to two, and hence the number of neighbors of each node is at least three. However, we do not set the maximum degree of the moralized graph. We chose the present parameters (θ_{jk}) in (4.2) at random in the range $\theta_{jk} \in [-0.75, -0.25]$ for Poisson DAG models and $\theta_{jk} \in [0.25, 0.75]$ for Exponential DAG models to ensure the edge weights are bounded away from 0.

In Figure 4.8, we plot the proportion of simulations in which our generalized ODS algorithm recovers the correct causal ordering to validate that NEF-QVF ODS algorithm can fully recover DAG models. We plot the accuracy rates in recovering the true causal ordering π^* as a function of sample size ($n \in \{200, 400, 600, 800, 1000\}$) for different node sizes ($p = 10$ for (a) and (c), and $p = 50$ for (b) and (d)). In each sub-figure, two NEF-QVF ODS algorithms are used; (i) GLMLasso [18] is applied in Step 1) where we chose a tuning parameter 0.1 and (ii) the GES algorithm [9] is applied in Step



(a) Poisson: $p = 20$ (b) Poisson: $p = 100$

(c) Poisson: $p = 20$ (d) Poisson: $p = 100$

Figure 4.9:: Comparison of our algorithms using the GES algorithm (in Steps 1) and 3)) and GLMLasso (in Steps 1) and 3)) to ODS algorithm using GLMLasso (in Steps 1) and 3)) and two standard DAG learning algorithms (the MMHC and the GES algorithms) in terms of Hamming distance to skeletons and directed edges of Poisson DAG models. The end of each bar corresponds to each average normalized hamming distance plus or minus its standard error

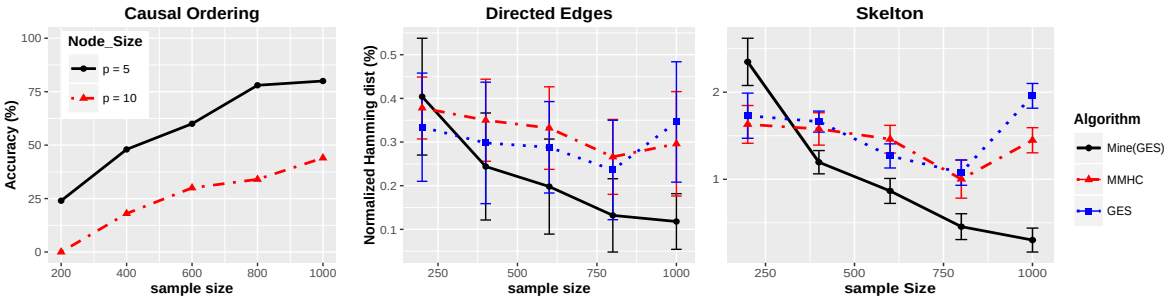
1) where we used the mBDe [29] (modified Bayesian Dirichlet equivalent) score and the moralized graph is generated from the output of the GES algorithm. Although any state-of-the-art algorithm can be applied, we chose the those two algorithms because they seem to work better in terms of recovering moralized graph in our simulation setting.

Figure 4.9 provides a comparison of how accurately our algorithm performs in terms of Hamming distance to the generalized ODS algorithm and two state-of-the-art directed graphical model learning algorithms (the MMHC and GES algorithms) for

Poisson DAG models. Similar to learning causal ordering, we used our two different algorithms using GLMLasso in both Steps 1) and 3) and the GES algorithm with the mBDe score in both Steps 1) and 3). For the generalized ODS algorithm, we used GLMLasso in both Steps 1) and 3). We considered small DAGs with $p = 10$ for (a) and (b), $p = 50$ for (c) and (d). We also considered two Hamming distance measures. We measured the Hamming distance to the skeleton of the true DAG for (a) and (c) which is the set of edges of the DAG without directions. In addition, we measured the Hamming distance for the edges with directions for (b) and (d). The reason we considered the skeleton is that the comparison methods recover up to the skeleton of the DAG. We normalize the Hamming distances by dividing it by the maximum number of errors $p(p-1)$ and $\binom{p}{2}$, respectively. Therefore, the overall score is a percentage.

As we see in Figure 4.9, our algorithm significantly out-performs the MMHC and GES algorithms in terms of both directed edges and skeleton when sample size is large enough. For small sample size cases, the GLMLasso-base our algorithm and the generalized ODS algorithm show bad performance because it frequently fails to recover the causal ordering. However, our GES-base algorithm is strictly better than the GES algorithm. It is because the GES-base NEF-QVF ODS algorithm only adds directional information to the estimated skeleton via the GES algorithm and hence GES-base NEF-QVF ODS algorithm is always better than the GES algorithm in terms of recovering both directed edges and skeleton. We can also see that our algorithms are better than the generalized ODS algorithm, which support our main contribution of this paper. As sample size increases, our algorithm recovers the true directed edges and skeleton of the DAG better.

Now we concern the statistical performance for exponential DAG models where a conditional distribution a node given its parents is exponential to show that our NEF-QVF ODS algorithm works for continuous data. In all experiments we used the GES algorithm in Steps 1) and 3) of the NEF-QVF ODS algorithm. Figure 4.10 rep-



(a) Expontial (b) Expontial: $p = 5$ (c) Expontial: $p = 5$

Figure 4.10:: Comparison of our NEF-QVF ODS algorithms using the GES algorithm (in Steps 1) and Step 3)) and GLMLasso (in Steps 1) and Step 3)) to two standard DAG learning algorithms (the MMHC and the GES algorithms) in terms of Hamming distance to skeletons and directed edges of Exponential DAG models. The end of each bar corresponds to each average normalized hamming distance plus or minus its standard error

resents the accuracy rates of the recovering the causal ordering for (a), the normalized Hamming distance to the true skeleton for (b), the normalized Hamming distance to the true edge set of the DAG for (c). Accuracies vary as a function of sample size ($n \in \{200, 400, 600, 800, 1000\}$) for each node size ($p = \{5, 10\}$). Figure 4.10 indicates that the NEF-QVF ODS algorithm recovers the causal ordering and the true structure of a DAG even for continuous data.

Chapter 5

Learning DAG Models Using Moralization and Interventions

5.1 Introduction

A popular framework for representing causal or directional relationships are directed acyclic graphical (DAG) models, also known as Bayesian networks. In such models parents of a vertex are *causes* and their edges are understood as *causal influences*. One of the major challenges associated with DAG models is that they are in general not identifiable from observational data alone and can be identified only up to their *Markov equivalence class* (MEC). Therefore if the goal is to learn all causal directions further information from experiments based on *interventions* are required. Here we focus on the practically relevant setting where the number of variables of interest p is potentially large, and our goal is to learn all directions of a DAG model using a combination of interventional experiments and observational data.

Recently, a number of DAG learning algorithms using a combination of observational data and interventions have been proposed (see e.g. [24, 25, 26, 27, 63]). More specifically, Hauser and Bühlmann [24] extended the notion of the (MEC) to the interventional case, and introduced the Greedy Interventional Equivalence Search (GIES) algorithm which is known to recover the DAG model provided algorithms for

learning the MEC are accurate. In other related work, Hauser and Bühlmann [25] and He and Geng [27] presented strategies for actively determining which nodes to intervene or experiment on by exploiting properties of the MEC for interventional graphs. However these approaches rely on accurate recovery of the MEC and many existing algorithms for learning the MECs are unreliable (see e.g. [73]). Therefore estimating the MEC based on observational may lead to errors which would lead to downstream errors in estimating other directions using interventions.

In this paper, we propose both passive and active learning strategies using the *moralized graphs* rather than the MEC. The advantage of using moralized graphs instead of the MEC is that recovering the moralized graph is more reliable since it does not require as strong assumptions as those needed for recovering the MEC. Furthermore, the moralized graph can be accurately estimated even in high-dimensional settings, where the number of nodes are larger than the measured sample size (see e.g. [4, 56, 57, 77]). Major contributions of our paper are to (1) introduce new rules for recovering directions of edges by comparing the moralized graphs from observational and interventional data and develop a passive learning strategy which we show out-performs the state-of-the-art GIES algorithm, and (2) develop an active learning algorithm for DAG models which reduces the number of interventions and allows reliable recovery in the high-dimensional settings.

Our passive and active learning strategies involve combining to basic concepts, moralized graphs and interventional graphs and developing new theory which guarantees their success for learning DAG models. The passive learning algorithm involves two iterative steps: (i) learn the *leaf* nodes by using the fact that interventions applied to leaf nodes have no neighbors in the moralized or undirected graph, and (ii) learning the parents of the leaf nodes by exploiting the fact that parents of the leaf nodes correspond exactly to neighbors in the moralized graph. Our experiments demonstrate the superior performance of our passive learning algorithm relative to the state-of-the-art GIES algorithm in terms of recovering the underlying DAG model. The active

learning strategy involves iteratively selecting which nodes to perform interventions on so that the moralized graph on the interventional data reveals the most information about the directions of the edges. Our active learning algorithm has three steps to be repeated iteratively: (i) choosing subsets of nodes to intervene on using moralized graph or input graph from previous step; (ii) learn the moralized graph based on the interventional data; (iii) use rules developed in this paper to determine directions of the DAG model based on the interventional moralized graph. Experimental results using our active learning strategy performs well even in the high-dimensional setting provided that the maximum degree of the moralized graph is bounded.

The remainder of this paper is organized is follows. In Section 2, we introduce two important concepts, interventional data and graphs and the moralized graph. In Section 3 we introduce the passive learning strategy along with theoretical guarantees on the sample size in terms of the number of nodes and the maximum degree of the moralized graph. In Section 4 we introduce the active learning strategy that applies to both small-scale and large-scale DAG models and we introduce addition theoretical results on modified Meek rules for moralized graphs that guarantee the success of our algorithm. Finally in Section 5 we present experimental results for both the passive and active learning strategies on a range of DAG models both in the low-dimensional and high-dimensional settings.

5.2 Background

Directed graphs. A DAG $G = (V, E)$ consists of a set of nodes V and a set of directed edges E with no directed cycle. We usually set $V = [p] := \{1, 2, ..., p\}$ and associate with the nodes a random vector $X := (X_1, X_2, ..., X_p)$ which takes values in some product measure $(\mathcal{X}, \mathcal{A}, \mu) = (\prod_{i=1}^{p} \mathcal{X}_i, \bigotimes_{i=1}^{p} \mathcal{A}_i, \bigotimes_{i=1}^{p} \mu_i)$ with $\mathcal{X}_i \subset \mathbb{R}\ \forall\ i$. For any subset of component indices $A \subset [p]$, we use the notation $\mathcal{X}_A := \prod_{a \in A} \mathcal{X}_a$, $X_A := (X_a)_{a \in A}$. We use $\mathrm{pa}(k)$ to denote the *parents*, $\mathrm{ch}(k)$ to denote the *children*, and $\mathrm{sp}(k)$ to denote the *spouses* of node k. (A node j is a spouse of node k if j and k have

a common child). Lastly, we use $\mathrm{an}(k)$ to denote the *parents* of k.

Interventions. We borrow the notation from [24]. We consider *stochastic interventions* modeling the effect of setting or forcing one or several random variables X_I, where $I \subset [p]$ is called the *intervention target*, to the value of *independent* random variables U_I. The joint product density of U_I on $\mathcal{X}_I$, called *level density*, is denoted by $\tilde{f}$. Extending the do() operator in [47] to stochastic interventions, we denote the density of X under such an intervention by $f(x \mid \mathrm{do}_D(X_I = U_I))$. Using truncated factorization and the assumption of independent intervention variables, this *interventional density* can be written as

$$f(x \mid \mathrm{do}_D(X_I = U_I)) = \prod_{i \notin I} f(x_i \mid x_{\mathrm{pa}(i)}) \prod_{i \in I} \tilde{f}(x_i) \ . \tag{5.1}$$

Intervention graph. For a DAG $G = (V, E)$ and an intervention target $I \subset [p]$, the *intervention graph* is a DAG $G_I = (V, E_I)$, where $E_I := \{(j, k) \mid (j, k) \in E, k \notin I\}$.

Moralized graph. For a DAG $G = (V, E)$, the *moralized graph* of G is an undirected graph $G^m = (V, E^m)$, where E^m is obtained by adding (1) an undirected edge $\{j, k\}$ to E^m for each $(j, k) \in E$, and (2) an undirected edge between (j, k) to E^m if j and k have a common child. We use $\mathcal{N}(j)$ to denote the neighbors of node j in G^m, also known as *Markov blanket* [51].

Interventional moralized graph. For a DAG $G = (V, E)$ and an intervention target $I \subset [p]$, the *interventional moralized graph* $G_I^m = (V, E_I^m)$ is the moralized graph of the intervention graph G_I. We use $\mathcal{N}_I(j)$ to denote the neighbors of node j in the interventional moralized graph G_I^m.

Interventional data. We consider *interventional data* $(\mathcal{T}, \mathbf{X})$ of sample size n, where

$$\mathcal{T} = \begin{pmatrix} T^{(1)} \\ \vdots \\ T^{(n)} \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} — X^{(1)} — \\ \vdots \\ — X^{(n)} — \end{pmatrix} \tag{5.2}$$

where for each $i \in [n]$, $T^{(i)}$ denoted the intervention target under which the sample $X^{(i)} = (X_1^{(i)}, X_2^{(i)}, ..., X_p^{(i)})$ was produced. Mathematically, $X^{(1)}, X^{(2)}, \ldots, X^{(n)}$ are

independent, and

$$X^{(i)} \sim f\big(\cdot \mid \mathrm{do}_D(X^{(i)}_{T^{(i)}} = U_{T^{(i)}})\big),\ U_{T^{(i)}} \sim \tilde{f}_{T^{(i)}}, \quad i = 1, \ldots, n.$$

A more thorough background on these relevant concepts is provided in the supplementary material.

5.3 Passive Learning

In this section, we present a passive learning algorithm CLMG (Causal Learning using Moral Graphs) for recovering the DAG structure using interventional data and moral graphs i.e. it uses pre-collected interventional data to recover the DAG structure. We assume that every node has been intervened at least once.

We first identify the leaf nodes by exploiting the fact that a leaf node l has no neighbors in the interventional moralized graph when l is intervened. We then learn the parents of the leaf node by using the fact that parents of leaf nodes correspond exactly to neighbors in the moralized graph. The leaf node is then removed and only the remaining subgraph is considered. This process is repeated till the subgraph is empty.

The CLMG algorithm uses a black-box FindNeighbors(data, target node, search set) function that estimates the neighborhood of a target node in the moralized graph from a set of search nodes, by using sampled data. Note that we can use a number of standard algorithms for FindNeighbors$(\cdot)$ since it is the same as learning neighborhoods of nodes in undirected graphs (see e.g. [4, 56, 57, 76, 77, 78, 79]). In our numerical experiments, a thresholding approach developed in [78] is applied.

A leaf node can be recovered as the node whose neighborhood is empty when it is intervened because an intervention eliminates edges between an intervened node and its parents. Let $\mathbf{X}_j = \{X^{(i)} : j \in T^{(i)}\}$ and $\mathbf{X}_{-j} = \{X^{(i)} : j \notin T^{(i)}\}$ denote the data where j was intervened and not intervened respectively. One can find the interventional neighborhood of a node j using FindNeighbors(data $= \mathbf{X}_j$, target node $= j$, search set $= V \setminus j$), and declaring j to be a leaf node if the neighborhood

returned is empty. However, the search set $V \setminus j$ is large, and hence FindNeighbors is likely to return false neighbors, making it difficult to recover the leaf nodes of a graph.

The CLMG algorithm cleverly solves this problem by exploiting the fact that the interventional neighborhood of a node j is a subset of $\mathcal{N}(j)$, because an intervention only eliminates edges incident on the intervened node. The CLMG algorithm first recovers $\mathcal{N}(j)$, and then searches for the interventional neighborhood of j only in $\mathcal{N}(j)$ (instead of $V \setminus j$). As before, it declares j to be a leaf node if FindNeighbors(data $= \mathbf{X}_j$, target node $= j$, search set $= \mathcal{N}(j)$) returns an empty set. This is better because $|\mathcal{N}(j)| \ll |V \setminus j|$ if the moralized graph is sparse. Furthermore, $\mathcal{N}(j)$ can be recovered from $\mathbf{X}_{-j}$ because although every measurement has a different set of intervened nodes, the conditional distribution of X_j given all other variables is the same as long as node j is not intervened. This estimation of $\mathcal{N}(j)$ will also be accurate because $|\mathbf{X}_{-j}|$ is likely to be large.

Algorithm 5.1 CLMG($\mathcal{T}, \mathbf{X}$): Causal Learning using Moral Graphs

Input: $(\mathcal{T}, \mathbf{X})$ interventional data

Output: $\widehat{G} = (V, \widehat{E})$ estimated graph structure

$\widehat{E} = \emptyset$

remainingNodes $= \{1, 2, ..., p\}$

while remainingNodes $\neq \emptyset$ **do**

    leaves = FindLeaves(N = remainingNodes, $\mathcal{T}, \mathbf{X}$)

    for l in leaves **do**

        $\mathbf{X}_l = \{X^{(i)} : l \in T^{(i)}\}$

        parents $\leftarrow$ FindNeighbors(target $= l$, search = remainingNodes, data $= \mathbf{X}_l$)

        for r in parents **do**

            $\widehat{E} = \widehat{E} \cup \{(r, l)\}$

        end for

    end for

    remainingNodes $\leftarrow$ remainingNodes $\setminus$ leaves

end while

return $\widehat{G} = (V, \widehat{E})$

FINDLEAVES($N, \mathcal{T}, \mathbf{X}$): Find leaf nodes among n given interventional data

Input: N set of nodes to search and $(\mathcal{T}, \mathbf{X})$ interventional data
Output: L set of leaf nodes
$L = \emptyset$
for s in N **do**
    $\mathbf{X}_{-s} = \{X^{(i)} : s \notin T^{(i)}\}$
    unbrs = FINDNEIGHBORS(target $= s$, search $= N$, data $= \mathbf{X}_{-s}$)
    if unbrs $\neq \emptyset$ **then**
        $\mathbf{X}_s = \{X^{(i)} : s \in T^{(i)}\}$
        children = FINDNEIGHBORS(target $= s$, search $= unbrs$, data $= \mathbf{X}_s$)
        if children $= \emptyset$ **then**
            $L = L \cup \{s\}$
        end if
    end if
end for
return L

5.3.1 Statistical Guarantees for the CLMG Algorithm

Here we provide statistical guarantees for the CLMG algorithm 5.1. For the purposes of this guarantee, we consider an intervention strategy where we perform an intervention at every node and collect n_0 samples per intervention. We consider single interventions because 1) they are simple, 2) they form a sufficient set to estimate the entire DAG [24], and 3) it is possible to determine the total joint effect of multiple interventions from single intervention effects [45]. We thus perform p single-node interventions. Our algorithm easily allows other intervention strategies and we use single-node interventions purely for illustration.

Theorem 5.1. *Consider a DAG $G = (V, E)$ with the maximum degree of the moralized graph, d. If single-node interventions are performed at every node and n_0 measurements are made per intervention, then Alg.5.1 recovers the true DAG with high*

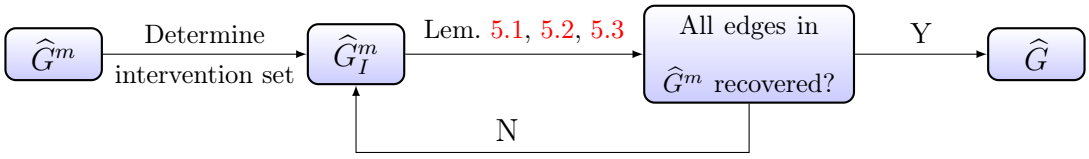


Figure 5.1:: Outline of our active learning algorithm

probability:

$$P(\widehat{G} \neq G) \leq \sum_{j=1}^{p} j \left\{ \delta_N(n_0(p-1), j-1, d) + \delta(n_0, \min(d, j-1), d) \right\}, \tag{5.3}$$

where $\delta_N(n, p-1, d)$ *is an error bound for estimating a moralized graph with sample size* n*, possible neighborhood size* $p-1$*, and the maximum degree of moralize graph* d*.*

The detail of the proof is in the supplementary material. $\delta_N(n_0, p-1, d)$ can be found using existing theoretical results for subset selection in regression which we are treating as a black box in this paper. For example using the GLM lasso approach, $\delta_N(n_0, p-1, d) \leq \frac{1}{p^4}$ provided $n_0 \geq c.d. \log p$ for an appropriately chosen constant c (see e.g. [56, 77]). The detail of the proof is in the supplementary material. Our experiments show that the CLMG algorithm 5.1 performs well in practice.

5.4 Active Learning Algorithm

In this section, we develop a new active learning algorithm for recovering the structure of a DAG using its moralized graph.

Our algorithm is outlined in Figure 5.1 and consists of four main steps: **(1)** estimating the moralized graph of a DAG using undirected graph learning algorithms, **(2)** determining a set of nodes to be intervened, **(3)** estimating the interventional moralized graph of the DAG from interventional data, and **(4)** determining the direction of many edges as possible by comparing the moralized graph from step 1 and the interventional moralized graph from step 3 and applying rules we develop below. We then repeat steps 2-3-4 till the entire DAG is recovered.

Steps 1 and 3 can be performed using standard undirected graph learning algorithms [4, 56, 57, 76, 77, 78, 79]. The novelty of our algorithm lies in steps 2 and 4. For step 2, we design an optimal algorithm for determining which nodes to intervene, so as to minimize the total number of interventions, thus enabling our algorithm to recover the structure of the DAG even in high-dimensional ($p > n$) settings. Step 4 of our algorithm is similar in flavor to existing Meek rules [41] used to determine the direction of an edge given V-structures in a Markov Equivalence Class (MEC). However, since we use moralized graphs instead of MECs, we require new methods to determine the direction of an edge from the moralized graph and interventional moralized graph. Using moralized graphs allows us to identify directed graphs without strong identifiability assumptions such as the faithfulness.

We begin by discussing step 4 of our algorithm. Note that for a node j, its moralized neighborhood $\mathcal{N}(j) = \mathrm{pa}(j) \cup \mathrm{ch}(j) \cup \mathrm{sp}(j)$. So step 4 boils down to distinguishing between $\mathrm{pa}(j), \mathrm{ch}(j)$ and $\mathrm{sp}(j)$. The following three lemmas allow us to do this.

Lemma 5.1. *For any $j, k \notin I$, suppose that $(j,k) \in E^m$ and $(j,k) \notin E^m_I$. Then $(j,k) \in E$. Furthermore, there exists at least one $i \in I$ such that i is a common child of j and k.*

Lemma 5.2. *Suppose that no nodes in I are adjacent in G^m. Let $S = \mathcal{N}(j) \cap \mathcal{N}_I(j)^c$. Then, for any $k \in S$, $(k,j) \in E$.*

The detail of the proof is in the supplementary material.

Lemma 5.3. *Suppose that no components of I are adjacent in G^m. Let $j \in I$, $S = \mathcal{N}(j) \cap \mathcal{N}_I(j)$, and $\ell = \mathcal{N}(j) \cap \mathcal{N}(k)$ for $k \in S$.*

(a) If $\ell = \emptyset$, $(j,k) \in E$.

(b) If for all $l \in \ell, (l,j) \in E$, $(j,k) \in E$.

(c) If for any $t \in V \setminus \ell$ and $j \in an(t)$, there exists $(t, k) \in E$, then the edge between (j, k) in the moralized graph is generated by a common child.

We defer the proof to the supplementary material.

Lemma 5.1

Input: I, $\widehat{G}^m$ and $\widehat{G}^m_I$
Output: $\widehat{G} = (V, \widehat{E})$
$\widehat{E} = \emptyset$
for $j, k \notin I$ **do**
    $\ell = \widehat{\mathcal{N}}(j) \cap \widehat{\mathcal{N}}(k) \cap I$
    if $(j, k) \in \widehat{E}^m, (j, k) \notin \widehat{E}^m_I$ **then**
        $\widehat{E}^m := \widehat{E}^m \setminus \{(j, k), (k, j)\}$
        if $|\ell| = 1$ **then**
            $\widehat{E} := \widehat{E} \cup \{(j, \ell), (k, \ell)\}$
        end if
    end if
end for
Return: $\widehat{E}$

Choice of nodes to be intervened

Input: $\widehat{G}^m$
Output: I set of nodes to intervene
while There is an unshielded triple (i_1, a, i_2) in $\widehat{G}^m$ such that i_1 or i_2 is connected to unidentified edges in G^m, and i_1, i_2 are not adjacent to all elements of I in $\widehat{G}^m$ **do**
    Add $i_1, i_2 \in I$.
    while For any $i \in I$, there is is unshielded triple (i, a, i_3) and i_3 is not adjacent to all elements of I in $\widehat{G}^m$ **do**
        Add $i_3 \in I$
    end while
end while
while There is a node i_4 not adjacent to all elements of I in $\widehat{G}^m$ **do**
    Add $i_4 \in I$
end while
Return: I

Lemma 5.2

Input: I, $\widehat{G}^m$ and $\widehat{G}^m_I$
Output: $\widehat{G} = (V, \widehat{E})$
for $j \in I$ **do**
    $S = \widehat{\mathcal{N}}(j) \cap \widehat{\mathcal{N}}_I(j)^c$.
    for $k \in S$ **do**
        $\widehat{E} := \widehat{E} \cup \{(k, j)\}$
    end for
end for
Return: $\widehat{E}$

Lemma 5.3

Input: I, $\widehat{G}^m$ and $\widehat{G}_I^m$
Output: $\widehat{G} = (V, \widehat{E})$
for $j \in I$ **do**
    $C = \widehat{\mathcal{N}}(j) \cap \widehat{\mathcal{N}}_I(j)$
    for $k \in C$ **do**
        $\ell = \widehat{\mathcal{N}}(j) \cap \widehat{\mathcal{N}}(k)$
        if $\ell = \emptyset$ **then** ▷ Lem. 5.3(a)
            $\widehat{E} := \widehat{E} \cup \{(j, k)\}$
        end if
        if For any $l \in \ell$, $(l, j) \in \widehat{E}$ **then**
            $\widehat{E} := \widehat{E} \cup \{(j, k)\}$ ▷ Lem. 5.3(b)
        end if
        if $(\ell, k) \in \widehat{E}$ and $j \in \text{an}(\ell)$ **then**
            $\widehat{E}^m := \widehat{E}^m \setminus \{(j, k), (k, j)\}$
            if $|\ell| = 1$ **then** ▷ Lem. 5.3(c)
                $\widehat{E} := \widehat{E} \cup \{(j, \ell), (\ell, j)\}$
            end if
        end if
    end for
end for
Return: $\widehat{E}$

For step 2 of our algorithm, an important question is how to determine an optimal set of nodes intervene. Taking a hint from the above lemmas, we consider two guiding principles while choosing which nodes to intervene. First, no adjacent nodes in the moralized graph should be intervened. Second, choose the maximum number of nodes that can be intervened while obeying the first principle. The intuition is as follows. Recall that we use the difference between the moralized graph and the interventional moralized graph to determine directions of edges in step 4. If two adjacent nodes are intervened, we do not gain any information about the direction of the edge between the two nodes which is why we avoid intervening adjacent nodes. Further, the higher the number of non-adjacent nodes we intervene, the more differences we are likely to

find between the moralized graph and interventional moralized graph.

Here is the main strategy for choosing nodes to be intervened. For ease of notation we use I_0 as the set of nodes to be estimated. **(1)** We first find an unshielded triple (i_1, a, i_2) in the moralized graph G^m such that i_1 or i_2 is connected to unidentified edges in G^m, and then choose the two end nodes for intervention. $I_0 := I_0 \cup \{i_1, i_2\}$. It guarantees at least two nodes are chosen in a multiple-nodes intervention. **(2)** Next, we find an unshielded triple such that one of the end nodes of an unshielded triple is an element of nodes to intervened (i_1, b, i_3) or (i_2, b, i_3) and i_3 is not adjacent to all elements of $I_0 = \{i_1, i_2\}$. We add i_3 to I_0, and repeat this procedure until we cannot find any unshielded triple such that one of the end nodes of an unshielded triple is an element of I_0. **(3)** Next, we choose a new unshielded triple (i_3, b, i_4) such that both i_3 and i_4 are not adjacent to all elements of I_0, and add i_3 and i_4 to I_0. We repeat the procedure (2) and (3) until there is no unshielded triple satisfying the conditions. **(4)** We find a node not adjacent to all elements of I_0, and then add the node to I_0. We repeat the procedure (4) until there is no node which is not adjacent to I_0.

Algorithm 5.2 Active Learning Algorithm

1: **Input**: **X** observational data
2: **Output:** $\widehat{G} = (V, \widehat{E})$
3: Step 1) Estimate the moralized graph G^m via a standard undirected graph learning algorithm
4: Step 2) Choose nodes to be intervened I.
5: Step 3) Generate X_I interventional data, and then estimate the interventional moralized graph $\widehat{G}_I^m$ via a standard undirected graph learning algorithm
6: Step 4) Estimate the structure of a DAG using Lem. 5.1, 5.2, 5.3 ,and Meek rules.
7: Repeat Step 2) - Step 4) until every direction of an edge in G^m is recovered.
8: **Return:** $\widehat{G} = (V, \widehat{E})$

We illustrate our selection strategy with an example. In Fig. 5.2, $(1, 2, 3)$,

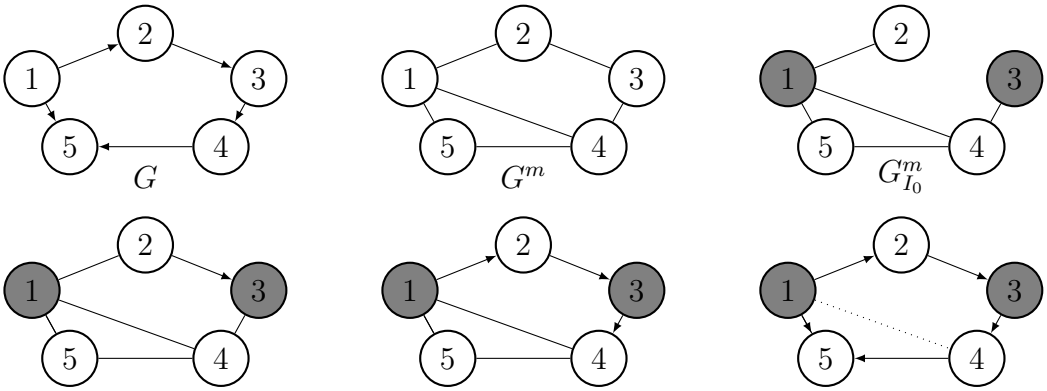


Figure 5.2:: Applying our algorithm to a 5-node cycle graph

$(1,4,3)$, $(2,3,4)$, $(2,1,5)$ and $(3,4,5)$ are unshielded triples, so that we can choose $\{1,3\}$, $\{2,4\}$, $\{2,5\}$ or $\{3,5\}$ for I_0, and do not need further steps since there are no more unshielded triples or nodes not-adjacent to I_0. Suppose $I_0 = \{1,3\}$. Since $(2,3) \in E^m$ and $\notin E^m_{I_0}$, $(2,3) \in E$ by Lem. 5.2. In addition, since $(1,2),(3,4) \in E^m$ and $\in E^m_{I_0}$ and both pairs do not consist triangles, $(1,2),(3,4) \in E$ by Lem. 5.3 (a). Lastly, $1 \in \text{an}(3)$, $(3,4) \in E$, and $(1,3,4)$ is an unshielded triple in G^m. It means that $1 \in \text{sp}(4)$, and therefore 5 is the common child of $(1,4)$ because $(1,4,5)$ is the only triangle in G^m.

Corollary 5.1. *Consider a DAG $G = (V,E)$ with the maximum degree of the moralized graph, d. Suppose that n_0 measurements are made per intervention and q interventions are required. Then our active learning algorithm recovers the true DAG with an error probability that is upper bounded by*

$$P(\widehat{G} \neq G) \leq (q+1) \cdot \delta_A(n_0, p, d) \tag{5.4}$$

where $\delta_A(n_0,p,d)$ is an error bound for estimating a moralized graph with sample size n_0, node size p, and the maximum degree of moralize graph d.

Cor. 5.1 shows that if the moralized graph is sparse and total number of interventions are bounded saying q, we can recover the structure of a DAG with $n = n_0 \cdot q$

total samples. For example if we use GLasso for Gaussian DAGs, there exist constants $c_1, c_2 > 0$ such that $\delta_A(n_0, p, d) = 1 - p^{c_1}$ if $n_0 > c_2 d^2 \log p$. Hence our active learning algorithm recovers a DAG structure with probability at least $1 - (q+1)\max(n_0, p)^{-c_1}$ if total sample size is $n = (q+1)n_0 > c_2(q+1)d^2 \log p$. Therefore our algorithm can recover a DAG in the high-dimensional ($n > p$) settings.

5.5 Experiments

In this section, we show that our passive algorithm performs better than the state-of-the-art GIES algorithm. We also show that our active learning algorithm recovers a DAG in high-dimensional settings if the moralized graph is sparse. We ran simulations using 100 realizations of a p-node random and some popular Gaussian linear DAGs such as bipartite, cycle, and chain (see e.g. in supplementary material) in which distribution $\mathbb{P}$ is defined by the following linear structural equations:

$$(X_1, X_2, ..., X_p)^T = B(X_1, X_2, ..., X_p)^T + \epsilon,$$

where $B \in \mathbb{R}^{p \times p}$ is an edge weight matrix with $B_{jk} = \beta_{jk}$ and β_{jk} is a weight of an edge from X_j to X_k and $\epsilon \sim \mathcal{N}(\mathbf{0}_p, I_p)$ where $\mathbf{0}_p = (0, 0, ..., 0)^T \in \mathbb{R}^p$ and $I_p \in \mathbb{R}^{p \times p}$ is the identity matrix. The matrix B encodes the DAG structure since if β_{jk} is non-zero, $k \to j$. For random graph we impose sparsity by assigning a probability that each coefficient of the matrix B is non-zero and we set the expected neighborhood size $\frac{p}{2}$. In addition for special structure graphs, we set β_{jk} to zero for non-edge and β_{jk} to non-zero edge weight for an edge. Non-zero β_{jk} were chosen uniformly at random from the range $\beta_{jk} \in [-0.75, -0.50] \cup [0.50, 0.75]$ for ensuring the edge weights are bounded away from 0. Furthermore, we used $U_I \sim \mathcal{N}(\mathbf{0}_{|I|}, I_{|I|})$ for an intervened variables.

We used the thresholding method provided by Yang et al [78] for recovering neighborhood of a node in the passive learning algorithm, and we used a combination of GLasso [57] and thresholding method [78] for recovering the moralized graph in the active learning algorithm.

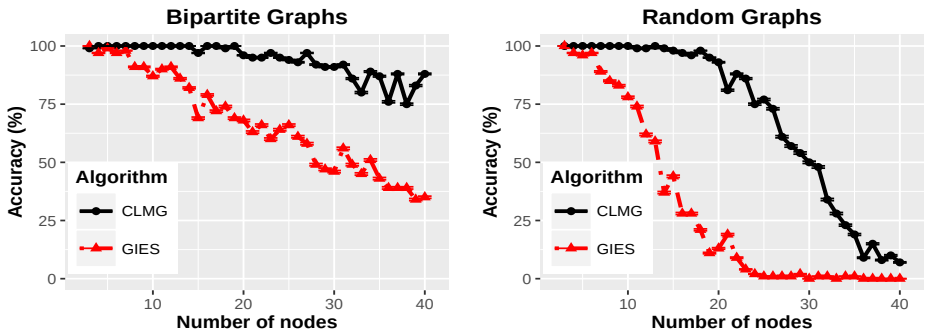


Figure 5.3:: Accuracy rates of recovering the structure of a DAG using our passive algorithm CLMG and GIES . The end of each bar corresponds to each accuracy rate plus or minus its standard error

5.5.1 Passive learning algorithm: CLMG

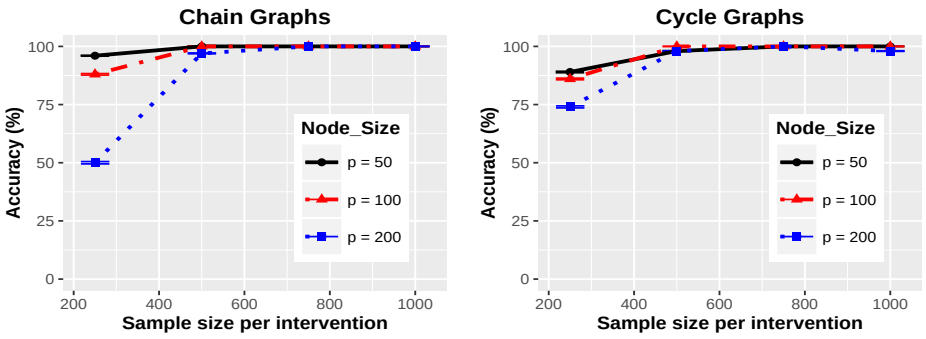
Fig. 5.3 provides a comparison of how accurately our passive learning algorithm 5.1 performs to the state-of-the-art DAG structure learning algorithm, GIES in [24] for both random and bipartite Gaussian linear DAGs. The data used was the same as the scheme described in Sec. 5.3.1 i.e. every node was intervened and $n_0 = 1000$ samples were collected per intervention. In addition, $n_0 = 1000$ samples were collected without any intervention. We plot the accuracy rates in recovering the structure of a DAG as a function of different node sizes $p \in \{3, ..., 40\}$. Fig. 5.3 shows that the CLMG algorithm significantly out-performs the GIES in terms of recovering the structure on average. This supports our main ideas that using the moralized graph instead of the MEC of a DAG is better in terms of recovering the structure although V-structures cannot be used if the moralized graph is used.

5.5.2 Active learning algorithm

In Fig. 5.4, we plot the proportion of simulations in which our active learning algorithm recovers the correct structure of a DAG to validate our main result in Sec 5.4 that our active learning algorithm recovers the structure of a DAG in the high-dimensional settings if the moralized graph is sparse. We used two popular DAGs (1)

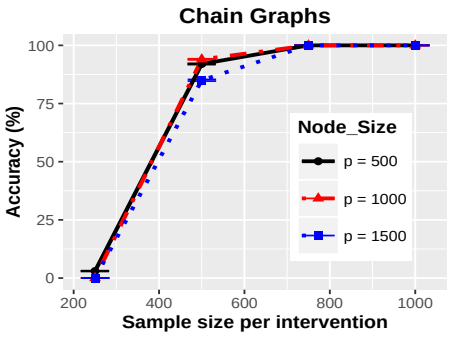
chain and (2) cycle to ensure the maximum degree of moralized graph is sparse. We plot the accuracy rates in recovering the structure of DAGs as a function of sample size per intervention $n_0 \in \{250, 500, 750, 1000\}$ for different node sizes, small-scale DAGs $p = \{50, 100, 200\}$ and large-scale DAGs $p = \{500, 1000, 1500\}$. For both cases, our active learning algorithm requires only one intervention, so the total sample sample size is $n = 2.n_0$.

Fig. 5.4 shows that our active learning algorithm recovers the structure of a DAG well as sample size increases for both chain and cycle DAGs. Fig. 5.4 also shows that our active learning algorithm accurately recovers the DAGs even in high-dimensional settings if the moralized graph is sparse, supporting our theoretical results in Section 5.4. We were unable to find an obvious way to compare our active learning strategy to existing strategies, since when different intervention schemes are used, accuracy comparisons do not make sense. However as far as we are aware, ours is the only strategy that scales to the high-dimensional setting.

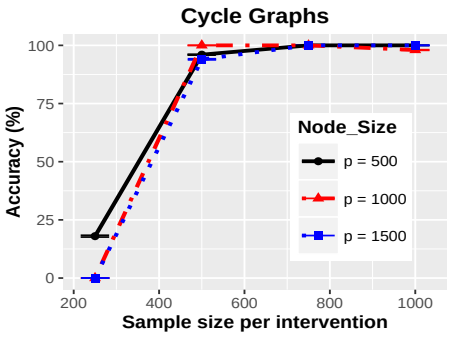


(a) Small-scale DAG

(b) Small-scale DAG



(c) Large-scale DAG



(d) Large-scale DAG

Figure 5.4:: Accuracy rates of recovering a DAG using our active algorithm for chain and cycle DAGs. The end of each bar corresponds to each accuracy rate plus or minus its standard error

Chapter 6

Learning Graphical Models with Feedback

6.1 Introduction

A fundamental goal in many scientific problems is to determine causal or directional relationships between variables in a system. A well-known framework for representing causal or directional relationships are directed graphical models. Most prior work on directed graphical models has focused on directed acyclic graphical (DAG) models, also referred to as Bayesian networks which are directed graphical models with no directed cycles. One of the core problems is determining the underlying DAG G given the data-generating distribution $\mathbb{P}$.

A fundamental assumption in the DAG framework is the *causal Markov condition* (CMC) (see e.g., [39, 68]). While the CMC is broadly assumed, in order for a directed graph G to be identifiable based on the distribution $\mathbb{P}$, additional assumptions are required. For DAG models, a number of identifiability and minimality assumptions have been introduced [23, 68] and the connections between them have been discussed [80]. In particular, one of the most widely used assumptions for DAG models is the *causal faithfulness condition* (CFC) which is sufficient for many search algorithms. However the CFC has been shown to be extremely restrictive, especially in the limited data setting [73]. In addition two minimality assumptions, the P-minimality and SGS-minimality assumptions have been introduced. These conditions are weaker

than the CFC but do not guarantee model identifiability [80]. On the other hand, the recently introduced sparsest Markov representation (SMR) and frugality assumptions [17, 55, 74] provide an alternative that is milder than the CFC and is sufficient to ensure identifiability. The main downside of the SMR and frugality assumptions relative to the CFC is that the SMR and frugality assumptions are sufficient conditions for model identifiability only when exhaustive searches over the DAG space are possible [55], while the CFC is sufficient for polynomial-time algorithms [23, 67, 68] for learning equivalence class of sparse graphs.

While the DAG framework is useful in many applications, it is limited since feedback loops are known to often exist (see e.g., [60, 59]). Hence, directed graphs with directed cycles [68] are more appropriate to model such feedback. However learning directed cyclic graphical (DCG) models from data is considerably more challenging than learning DAG models [60, 59] since the presence of cycles poses a number of additional challenges and introduces additional non-identifiability. Consequently there has been considerably less work focusing on directed graphs with feedback both in terms of identifiability assumptions and search algorithms. [66] discussed the CMC, and [60, 59] discussed the CFC for DCG models and introduced the polynomial-time cyclic causal discovery (CCD) algorithm [59] for recovering the Markov equivalence class for DCGs. Recently, Claassen et al. [12] introduced the FCI+ algorithm for recovering the Markov equivalence class for sparse DCGs, which also assumes the CFC. As with DAG models, the CFC for cyclic models is extremely restrictive since it is more restrictive than the CFC for DAG models. In terms of learning algorithms that do not require the CFC, additional assumptions are typically required. For example [43] proved identifiability for bivariate Gaussian cyclic graphical models with additive noise which does not require the CFC while many approaches have been studied for learning graphs from the results of interventions on the graph (e.g., [31, 32, 33, 34, 35]). However, these additional assumptions are often impractical and it is often impossible or very expensive to intervene many variables in the graph. This raises the question of

whether milder identifiability assumptions can be imposed for learning DCG models.

In this paper, we address this question in a number of steps. Firstly, we adapt the SMR and frugality assumptions developed for DAG models to DCG models. Next we show that unlike for DAG models, the adapted SMR and frugality assumptions are not strictly weaker than the CFC. Hence we consider a new identifiability assumption based on finding the Markovian DCG entailing the maximum number of d-separation rules (MDR) which we prove is strictly weaker than the CFC and recovers the Markov equivalence class for DCGs for a strict superset of examples compared to the CFC. We also provide a comparison between the MDR, SMR and frugality assumptions as well as the minimality assumptions for both DAG and DCG models. Finally we use the MDR and SMR assumptions to develop search algorithms for small-scale DCG models. Our simulation study supports our theoretical results by showing that the algorithms induced by both the SMR and MDR assumptions recover the Markov equivalence class more reliably than state-of-the art algorithms that require the CFC for DCG models. We point out that the search algorithms that result from our identifiability assumptions require exhaustive searches and are not computationally feasible for large-scale DCG models. However, the focus of this paper is to develop the weakest possible identifiability assumption which is of fundamental importance for directed graphical models.

The remainder of the paper is organized as follows: Section 6.2 provides the background and prior work for identifiability assumptions for both DAG and DCG models. In Section 6.3 we adapt the SMR and frugality assumptions to DCG models and provide a comparison between the SMR assumption, the CFC, and the minimality assumptions. In Section 6.4 we introduce our new MDR principle, finding the Markovian DCG that entails the maximum number of d-separation rules and provide a comparison of the new principle to the CFC, SMR, frugality, and minimality assumptions. Finally in Section 6.5, we use our identifiability assumptions to develop a search algorithm for learning small-scale DCG models, and provide a simulation study

that is consistent with our theoretical results.

6.2 Prior work on directed graphical models

The characterization of Markov equivalence classes is different for DAGs and DCGs. For DAGs, [72] developed an elegant characterization of Markov equivalence classes defined by the *skeleton* and *v-structures*. The skeleton of a DAG model consists of the edges without directions.

However for DCGs, the presence of feedback means the characterization of the MEC for DCGs is considerably more involved. [60] provides a characterization. The presence of directed cycles changes the notion of adjacency between two nodes. In particular there are *real* adjacencies that are a result of directed edges in the DCG and *virtual* adjacencies which are edges that do not exist in the data-generating DCG but can not be recognized as a non-edge from the data. The precise definition of real and virtual adjacencies are as follows.

Definition 6.1 (Adjacency [60]). *Consider a directed graph* $G = (V, E)$.

(a) For any $j, k \in V$, j *and* k *are* really adjacent *in* G *if* $j \to k$ *or* $j \leftarrow k$.

(b) For any $j, k \in V$, j *and* k *are* virtually adjacent *if* j *and* k *have a common child* ℓ *such that* ℓ *is an ancestor of* j *or* k.

Note that a virtual adjacency can only occur if there is a cycle in the graph. Hence, DAGs have only real edges while DCGs can have both real edges and virtual edges. Figure 6.1 shows an example of a DCG with a virtual edge. In Figure 6.1, a pair of nodes $(1, 4)$ has a virtual edge (dotted line) because the triple $(1, 4, 2)$ forms a v-structure and the common child 2 is an ancestor of 1. This virtual edge is created by the cycle, $1 \to 2 \to 3 \to 1$.

Virtual edges generate different types of relationships involving unshielded triples: (1) an unshielded triple (j, k, ℓ) (that is $j - \ell - k$) is called a *conductor* if ℓ is an ancestor

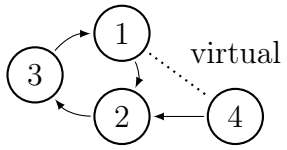


Figure 6.1:: 4-node example for a virtual edge

of j or k; (2) an unshielded triple (j, k, ℓ) is called a *perfect non-conductor* if ℓ is a descendant of the common child of j and k; and (3) an unshielded triple (j, k, ℓ) is called an *imperfect non-conductor* if the triple is not a conductor or a perfect non-conductor.

Intuitively, the concept of (1) a conductor is analogous to the notion of a non v-structure in DAGs because for example suppose that an unshielded triple (j, k, ℓ) is a conductor, then j is d-connected to k given any set S which does not contain ℓ. Moreover, (2) a perfect non-conductor is analogous to a v-structure because suppose that (j, k, ℓ) is a perfect non-conductor, then j is d-connected to k given any set S which contains ℓ. However, there is no analogous notion of an imperfect non-conductor for DAG models. We see throughout this paper that this difference creates a major challenge in inferring DCG models from the underlying distribution $\mathbb{P}$. As shown in [58] (Cyclic Equivalence Theorem), a necessary (but not sufficient) condition for two DCGs to belong to the same MEC is that they share the same real plus virtual edges and the same (1) conductors, (2) perfect non-conductors and (3) imperfect non-conductors. However unlike for DAGs, this condition is not sufficient for Markov equivalence. A complete characterization of Markov equivalence is provided in Richardson [58, 60] and since it is quite involved, we do not include here.

Even if we weaken the goal to inferring the MEC for a DAG or DCG, the CMC is insufficient for discovering the true MEC $\mathcal{M}(G^*)$ because there are many graphs satisfying the CMC, which do not belong to $\mathcal{M}(G^*)$. For example, any fully-connected graph always satisfies the CMC because it does not entail any d-separation rules. Hence, in order to identify the true MEC given the distribution $\mathbb{P}$, stronger identifiability assumptions that force the removal of edges are required.

6.2.1 Faithfulness and minimality assumptions

In this section, we discuss prior work on identifiability assumptions for both DAG and DCG models. To make the notion of identifiability and our assumptions precise, we need to introduce the notion of a true data-generating graphical model $(G^*, \mathbb{P})$. All we observe is the distribution (or samples from) $\mathbb{P}$, and we know the graphical model $(G^*, \mathbb{P})$ satisfies the CMC. Let $CI(\mathbb{P})$ denote the set of conditional independence statements corresponding to $\mathbb{P}$. The graphical model $(G^*, \mathbb{P})$ is *identifiable* if the Markov equivalence class of the graph $\mathcal{M}(G^*)$ can be uniquely determined based on $CI(\mathbb{P})$. For a directed graph G, let $E(G)$ denote the set of directed edges, $S(G)$ denote the set of edges without directions, also referred to as the skeleton, and $D_{sep}(G)$ denote the set of d-separation rules entailed by G.

One of the most widely imposed identifiability assumptions for both DAG and DCG models is the *causal faithfulness condition* (CFC) [68] also referred to as the stability condition in [51]. A directed graph is *faithful* to a probability distribution if there is no probabilistic independence in the distribution that is not entailed by the CMC. The CFC states that the graph is faithful to the true probability distribution.

Definition 6.2 (Causal Faithfulness condition (CFC) [68])**.** *Consider a directed graphical model* $(G^*, \mathbb{P})$*. A graph* G^* *is* faithful *to* $\mathbb{P}$ *if and only if for any* $j, k \in V$ *and any subset* $S \subset V \setminus \{j, k\}$*,*

$$j \text{ d-separated from } k \mid S \iff X_j \perp\!\!\!\perp X_k \mid X_S \text{ according to } \mathbb{P}.$$

While the CFC is sufficient to guarantee identifiability for many polynomial-time search algorithms [12, 23, 32, 59, 60, 68] for both DAGs and DCGs, the CFC is known to be a very strong assumption (see e.g., [17, 55, 73]) that is often not satisfied in practice. Hence, milder identifiability assumptions have been considered.

Minimality assumptions, notably the *P-minimality* [49] and SGS-minimality [23] assumptions are two such assumptions. The P-minimality assumption asserts that for directed graphical models satisfying the CMC, graphs that entail more d-separation

rules are preferred. For example, suppose that there are two graphs G_1 and G_2 which are not Markov equivalent. G_1 is *strictly preferred* to G_2 if $D_{sep}(G_2) \subset D_{sep}(G_1)$ and $D_{sep}(G_2) \neq D_{sep}(G_1)$. The P-minimality assumption asserts that no graph is strictly preferred to the true graph G^*. The SGS-minimality assumption asserts that there exists no proper sub-graph of G^* that satisfies the CMC with respect to the probability distribution $\mathbb{P}$. To define the term sub-graph precisely, G_1 is a sub-graph of G_2 if $E(G_1) \subset E(G_2)$ and $E(G_1) \neq E(G_2)$. [80] proved that the SGS-minimality assumption is weaker than the P-minimality assumption which is weaker than the CFC for both DAG and DCG models. While [80] states the results for DAG models, the result easily extends to DCG models.

Theorem 6.1 (Sections 4 and 5 in [80]). *If a directed graphical model* $(G^*, \mathbb{P})$ *satisfies*

(a) the CFC, it satisfies the P-minimality assumption.

(b) the P-minimality assumption, it satisfies the SGS-minimality assumption.

6.2.2 Sparsest Markov Representation (SMR) for DAG models

While the minimality assumptions are milder than the CFC, neither the P-minimality nor SGS-minimality assumptions imply identifiability of the MEC for G^*. Recent work by [55] developed the *sparsest Markov representation* (SMR) assumption and a slightly weaker version later referred to as *frugality* assumption [17] which applies to DAG models. The SMR assumption which we refer to here as the identifiable SMR assumption states that the true DAG model is the graph satisfying the CMC with the fewest edges. Here we say that a DAG G_1 is *strictly sparser* than a DAG G_2 if G_1 has *fewer* edges than G_2.

Definition 6.3 (Identifiable SMR [55]). *A DAG model* $(G^*, \mathbb{P})$ *satisfies the identifiable SMR assumption if* $(G^*, \mathbb{P})$ *satisfies the CMC and* $|S(G^*)| < |S(G)|$ *for every DAG* G *such that* $(G, \mathbb{P})$ *satisfies the CMC and* $G \notin \mathcal{M}(G^*)$.

The identifiable SMR assumption is strictly weaker than the CFC while also ensuring a method known as the Sparsest Permutation (SP) algorithm [55] recovers the true MEC. Hence the identifiable SMR assumption guarantees identifiability of the MEC for DAGs. A slightly weaker notion which we refer to as the weak SMR assumption does not guarantee model identifiability.

Definition 6.4 (Weak SMR (Frugality) [17]). *A DAG model* $(G^*, \mathbb{P})$ *satisfies the weak SMR assumption if* $(G^*, \mathbb{P})$ *satisfies the CMC and* $|S(G^*)| \leq |S(G)|$ *for every DAG* G *such that* $(G, \mathbb{P})$ *satisfies the CMC and* $G \notin \mathcal{M}(G^*)$.

A comparison of SMR/frugality to the CFC and the minimality assumptions for DAG models is provided in [55] and [17].

Theorem 6.2 (Theorems 2.5 and 2.8 in [55], and Theorem 3 in [17]). *If a DAG model* $(G^*, \mathbb{P})$ *satisfies*

(a) the CFC, it satisfies the identifiable SMR assumption and consequently weak SMR assumption.

(b) the weak SMR assumption, it satisfies the P-minimality assumption and consequently the SGS-minimality assumption.

(c) the identifiable SMR assumption, G^* *is identifiable up to the true MEC* $\mathcal{M}(G^*)$.

It is unclear whether the SMR/frugality assumptions apply naturally to DCG models since the success of the SMR assumption relies on the local Markov property which is known to hold for DAGs but not DCGs [58]. In this paper, we investigate the extent to which these identifiability assumptions apply to DCG models and provide a new principle for learning DCG models.

Based on this prior work, a natural question to consider is whether the identifiable and weak SMR assumptions developed for DAG models apply to DCG models and whether there are similar relationships between the CFC, identifiable and weak

SMR, and minimality assumptions. In this paper we address this question by adapting both identifiable and weak SMR assumptions to DCG models. One of the challenges we address is dealing with the distinction between real and virtual edges in DCGs. We show that unlike for DAG models, the identifiable SMR assumption is not necessarily a weaker assumption than the CFC. Consequently, we introduce a new principle which is the maximum d-separation rule (MDR) principle which chooses the directed Markov graph with the greatest number of d-separation rules. We show that our MDR principle is strictly weaker than the CFC and stronger than the P-minimality assumption, while also guaranteeing model identifiability for DCG models. Our simulation results complement our theoretical results, showing that the MDR principle is more successful than the CFC in terms of recovering the true MEC for DCG models.

6.3 Sparsity and SMR for DCG models

In this section, we extend notions of sparsity and the SMR assumptions to DCG models. As mentioned earlier, in contrast to DAGs, DCGs can have two different types of edges which are real and virtual edges. In this paper, we define the *sparsest* DCG as the graph with the fewest *total edges* which are virtual edges plus real edges. The main reason we choose total edges rather than just real edges is that all DCGs in the same Markov equivalence class (MEC) have the same number of total edges [58]. However, the number of real edges may not be the same among the graphs even in the same MEC. For example in Figure 6.2, there are two different MECs and each MEC has two graphs: $G_1, G_2 \in \mathcal{M}(G_1)$ and $G_3, G_4 \in \mathcal{M}(G_3)$. G_1 and G_2 have 9 total edges but G_3 and G_4 has 7 total edges. On the other hand, G_1 has 6 real edges, G_2 has 9 real edges, G_3 has 5 real edges, and G_4 has 7 real edges (a bi-directed edge is counted as 1 total edge). For a DCG G, let $S(G)$ denote the *skeleton* of G where $(j, k) \in S(G)$ is a real or virtual edge.

Using this definition of the skeleton $S(G)$ for a DCG G, the definitions of the identifiable and weak SMR assumptions carry over from DAG to DCG models. For

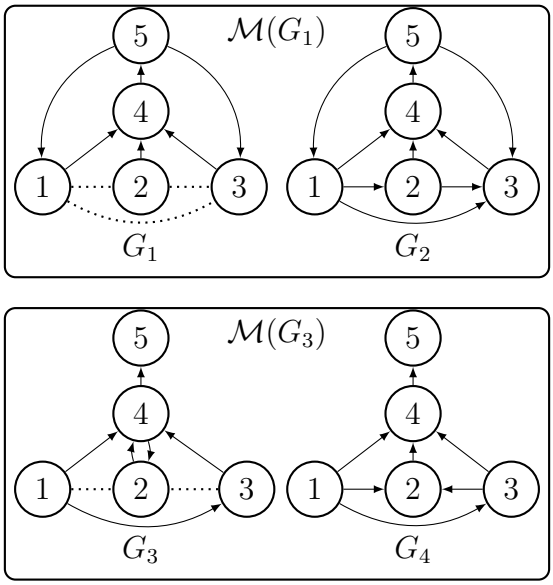


Figure 6.2:: 5-node examples with different numbers of real and total edges

completeness, we re-state the definitions here.

Definition 6.5 (Identifiable SMR for DCG models)**.** *A DCG model* $(G^*, \mathbb{P})$ *satisfies the identifiable SMR assumption if* $(G^*, \mathbb{P})$ *satisfies the CMC and* $|S(G^*)| < |S(G)|$ *for every DCG* G *such that* $(G, \mathbb{P})$ *satisfies the CMC and* $G \notin \mathcal{M}(G^*)$*.*

Definition 6.6 (Weak SMR for DCG models)**.** *A DCG model* $(G^*, \mathbb{P})$ *satisfies the weak SMR assumption if* $(G^*, \mathbb{P})$ *satisfies the CMC and* $|S(G^*)| \leq |S(G)|$ *for every DCG* G *such that* $(G, \mathbb{P})$ *satisfies the CMC and* $G \notin \mathcal{M}(G^*)$*.*

Both the SMR and SGS minimality assumptions prefer graphs with the fewest total edges. The main difference between the SGS-minimality assumption and the SMR assumptions is that the SGS-minimality assumption requires that there is no DCGs with a *strict subset* of edges whereas the SMR assumptions simply require that there are no DCGs with *fewer* edges.

Unfortunately as we observe later unlike for DAG models, the identifiable SMR assumption is not weaker than the CFC for DCG models. Therefore, the identifiable SMR assumption does not guarantee identifiability of MECs for DCG models. On the

other hand, while the weak SMR assumption may not guarantee uniqueness, we prove it is a strictly weaker assumption than the CFC. We explore the relationships between the CFC, identifiable and weak SMR, and minimality assumptions in the next section.

6.3.1 Comparison of SMR, CFC and minimality assumptions for DCG models

Before presenting our main result in this section, we provide a lemma which highlights the important difference between the SMR assumptions for graphical models with cycles compared to DAG models. Recall that the SMR assumptions involve counting the number of edges, whereas the CFC and P-minimality assumption involve d-separation rules. First, we provide a fundamental link between the presence of an edge in $S(G)$ and d-separation/connection rules.

Lemma 6.1. *For a DCG G, $(j,k) \in S(G)$ if and only if j is d-connected to k given S for all $S \subset V \setminus \{j,k\}$.*

Proof. First, we show that if $(j,k) \in S(G)$ then j is d-connected to k given S for all $S \subset V \setminus \{j,k\}$. By the definition of d-connection/separation, there is no subset $S \subset V \setminus \{j,k\}$ such that j is d-separated from k given S. Second, we prove that if $(j,k) \notin S(G)$ then there exists $S \subset V \setminus \{j,k\}$ such that j is d-separated from k given S. Let $S = \text{an}(j) \cup \text{an}(k)$. Then S has no common children or descendants, otherwise (j,k) are virtually adjacent. Then there is no undirected path between j and k conditioned on the union of ancestors of j and k, and therefore j is d-separated from k given S. This completes the proof. □

Note that the above statement is true for real or virtual edges and not real edges alone. We now state an important lemma which shows the key difference in comparing the SMR assumptions to other identifiability assumptions (CFC, P-minimality, SGS-minimality) for graphical models with cycles, which does not arise for DAG models.

Lemma 6.2. *(a) For any two DCGs G_1 and G_2, $D_{sep}(G_1) \subseteq D_{sep}(G_2)$ implies $S(G_2) \subseteq S(G_1)$.*

(b) There exist two DCGs G_1 and G_2 such that $S(G_1) = S(G_2)$, but $D_{sep}(G_1) \neq D_{sep}(G_2)$ and $D_{sep}(G_1) \subset D_{sep}(G_2)$. For DAGs, no two such graphs exist.

Proof. We begin with the proof for (a). Suppose that $S(G_1)$ is not a sub-skeleton of $S(G_2)$, meaning that there exists a pair $(j, k) \in S(G_1)$ and $(j, k) \notin S(G_2)$. By Lemma 6.1, j is d-connected to k given S for all $S \subset V \setminus \{j, k\}$ in G_1 while there exists $S \subset V \setminus \{j, k\}$ such that j is d-separated from k given S entailed by G_2. Hence it is contradictory that $D_{sep}(G_1) \subset D_{sep}(G_2)$. For (b), we refer to the example in Figure 6.3. In Figure 6.3, the unshielded triple $(1, 4, 2)$ is a conductor in G_1 and an imperfect non-conductor in G_2 because of a reversed directed edge between 4 and 5. By the property of a conductor, 1 is not d-separated from 4 given the empty set for G_1. In contrast for G_2, 1 is d-separated from 4 given the empty set. Other d-separation rules are the same for both G_1 and G_2.

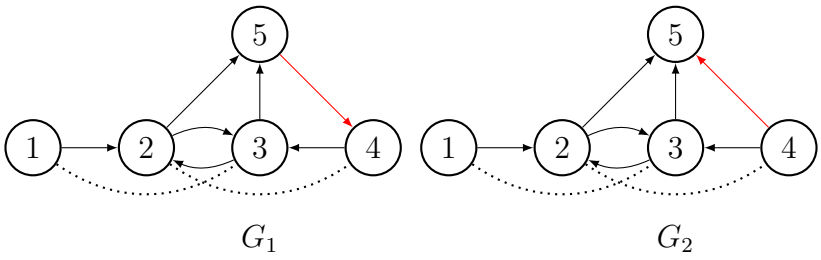


Figure 6.3:: 5-node examples for Lemma 6.2 and Theorem 6.3

□

Lemma 6.2 (a) holds for both DAGs and DCGs, and allows us to conclude a subset-superset relation between edges in the skeleton and d-separation rules in a graph G. Part (b) is where there is a key difference DAGs and directed graphs with cycles. Part (b) asserts that there are examples in which the edge set in the skeleton may be totally equivalent, yet one graph entails a strict superset of d-separation rules.

Now we present the main result of this section which compares the identifiable and weak SMR assumptions with the CFC and P-minimality assumption.

Theorem 6.3. *For DCG models,*

(a) the weak SMR assumption is weaker than the CFC.

(b) there exists a DCG model $(G, \mathbb{P})$ satisfying the CFC that does not satisfy the identifiable SMR assumption.

(c) the identifiable SMR assumption is stronger than the P-minimality assumption.

(d) there exists a DCG model $(G, \mathbb{P})$ satisfying the weak SMR assumption that does not satisfy the P-minimality assumption.

Proof. (a) The proof for (a) follows from Lemma 6.2 (a). If a DCG model $(G^*, \mathbb{P})$ satisfies the CFC, then for any graph G such that $(G, \mathbb{P})$ satisfies the CMC, $D_{sep}(G) \subseteq D_{sep}(G^*)$. Hence based on Lemma 6.2 (a), $S(G^*) \subseteq S(G)$ and $(G^*, \mathbb{P})$ satisfies the weak SMR assumption.

(b) We refer to the example in Figure 6.3 where $(G_2, \mathbb{P})$ satisfies the CFC and fails to satisfy the identifiable SMR assumption because G_1 has fewer edges than G_2 and $(G_1, \mathbb{P})$ satisfies the CMC.

(c) The proof for (c) again follows from Lemma 6.2 (a). Suppose that a DCG model $(G^*, \mathbb{P})$ fails to satisfy the P-minimality assumption. This implies that there exists a DCG G such that $(G, \mathbb{P})$ satisfies the CMC, $G \notin \mathcal{M}(G^*)$ and $D_{sep}(G^*) \subset D_{sep}(G)$. Lemma 6.2 (a) implies $S(G) \subseteq S(G^*)$. Hence G^* cannot have the fewest edges uniquely, therefore $(G^*, \mathbb{P})$ fails to satisfy the identifiable SMR assumption.

(d) We refer to the example in Figure 6.3 where $(G_1, \mathbb{P})$ satisfies the weak SMR assumption and fails to satisfy the P-minimality assumption. Further explanation is given in Figure D.1 in the appendix.

□

Theorem 6.3 shows that if a DCG model $(G, \mathbb{P})$ satisfies the CFC, the weak SMR assumption is satisfied whereas the identifiable SMR assumption is not necessarily satisfied. For DAG models, the identifiable SMR assumption is strictly weaker than the CFC and the identifiable SMR assumption guarantees identifiability of the true MEC. However, Theorem 6.3 (b) implies that the identifiable SMR assumption is not strictly weaker than the CFC for DCG models. On the other hand, unlike for DAG models, the weak SMR assumption does not imply the P-minimality assumption for DCG models, according to (d). In Section 6.5, we implement an algorithm that uses the identifiable SMR assumption and the results seem to suggest that on average for DCG models, the identifiable SMR assumption is weaker than the CFC.

6.4 New principle: Maximum d-separation rules (MDR)

In light of the fact that the identifiable SMR assumption does not lead to a strictly weaker assumption than the CFC, we introduce the maximum d-separation rules (MDR) assumption. The MDR assumption asserts that G^* entails more d-separation rules than any other graph satisfying the CMC according to the given distribution $\mathbb{P}$. We use $CI(\mathbb{P})$ to denote the conditional independence (CI) statements corresponding to the distribution $\mathbb{P}$.

Definition 6.7 (Maximum d-separation rules (MDR))**.** *A DCG model* $(G^*, \mathbb{P})$ *satisfies the maximum* d-separation *rules (MDR) assumption if* $(G^*, \mathbb{P})$ *satisfies the CMC and* $|D_{sep}(G)| < |D_{sep}(G^*)|$ *for every DCG* G *such that* $(G, \mathbb{P})$ *satisfies the CMC and* $G \notin \mathcal{M}(G^*)$.

There is a natural and intuitive connection between the MDR assumption and the P-minimality assumption. Both assumptions encourage DCGs to entail more d-separation rules. The key difference between the P-minimality assumption and the MDR assumption is that the P-minimality assumption requires that there is no DCGs that entail a *strict superset* of d-separation rules whereas the MDR assumption simply

requires that there are no DCGs that entail a *greater number* of d-separation rules.

6.4.1 Comparison of MDR to CFC and minimality assumptions for DCGs

In this section, we provide a comparison of the MDR assumption to the CFC and P-minimality assumption. For ease of notation, let $\mathcal{G}_M(\mathbb{P})$ and $\mathcal{G}_F(\mathbb{P})$ denote the set of Markovian DCG models satisfying the MDR assumption and CFC, respectively. In addition, let $\mathcal{G}_P(\mathbb{P})$ denote the set of DCG models satisfying the P-minimality assumption.

Theorem 6.4. *Consider a DCG model* $(G^*, \mathbb{P})$.

(a) If $\mathcal{G}_F(\mathbb{P}) \neq \emptyset$, *then* $\mathcal{G}_F(\mathbb{P}) = \mathcal{G}_M(\mathbb{P})$. *Consequently if* $(G^*, \mathbb{P})$ *satisfies the CFC, then* $\mathcal{G}_F(\mathbb{P}) = \mathcal{G}_M(\mathbb{P}) = \mathcal{M}(G^*)$.

(b) There exists a distribution $\mathbb{P}$ *for which* $\mathcal{G}_F(\mathbb{P}) = \emptyset$ *while* $(G^*, \mathbb{P})$ *satisfies the MDR assumption and* $\mathcal{G}_M(\mathbb{P}) = \mathcal{M}(G^*)$.

(c) $\mathcal{G}_M(\mathbb{P}) \subseteq \mathcal{G}_P(\mathbb{P})$.

(d) There exists a distribution $\mathbb{P}$ *for which* $\mathcal{G}_M(\mathbb{P}) = \emptyset$ *while* $(G^*, \mathbb{P})$ *satisfies the P-minimality assumption and* $\mathcal{G}_P(\mathbb{P}) \supseteq \mathcal{M}(G^*)$.

Proof. (a) Suppose that $(G^*, \mathbb{P})$ satisfies the CFC. Then $CI(\mathbb{P})$ corresponds to the set of d-separation rules entailed by G^*. Note that if $(G, \mathbb{P})$ satisfies the CMC and $G \notin \mathcal{M}(G^*)$, then $CI(\mathbb{P})$ is a superset of the set of d-separation rules entailed by G and therefore $D_{sep}(G) \subset D_{sep}(G^*)$ and $D_{sep}(G) \neq D_{sep}(G^*)$. This allows us to conclude that graphs belonging to $\mathcal{M}(G^*)$ should entail the maximum number of d-separation rules among graphs satisfying the CMC. Furthermore, based on the CFC $\mathcal{G}_F(\mathbb{P}) = \mathcal{M}(G^*)$ which completes the proof.

(c) Suppose that $(G^*, \mathbb{P})$ fails to satisfy the P-minimality assumption. By the definition of the P-minimality assumption, there exists $(G, \mathbb{P})$ satisfying the CMC such

that $D_{sep}(G^*) \subset D_{sep}(G)$ and $D_{sep}(G^*) \neq D_{sep}(G)$. Hence, G^* entails strictly less d-separation rules than G, and therefore $(G^*, \mathbb{P})$ violates the MDR assumption.

(b) For (b) and (d), we refer to the example in Figure 6.4. Suppose that X_1, X_2, X_3, X_4 are random variables with distribution $\mathbb{P}$ with the following CI statements:

$$CI(\mathbb{P}) = \{X_1 \perp\!\!\!\perp X_3 \mid X_2;\ X_2 \perp\!\!\!\perp X_4 \mid X_1, X_3;\ X_1 \perp\!\!\!\perp X_2 \mid X_4\}. \tag{6.1}$$

We show that $(G_1, \mathbb{P})$ satisfies the MDR assumption but not the CFC, whereas $(G_2, \mathbb{P})$ satisfies the P-minimality assumption but not the MDR assumption. Any graph satisfying the CMC with respect to $\mathbb{P}$ must only entail a subset of the three d-separation rules: $\{X_1 \text{ d-sep } X_3 \mid X_2; X_2 \text{ d-sep } X_4 \mid X_1, X_3;\ X_1 \text{ d-sep } X_2 \mid X_4\}$. Clearly $D_{sep}(G_1) = \{X_1 \text{ d-sep } X_3 \mid X_2;\ X_2 \text{ d-sep } X_4 \mid X_1, X_3\}$, therefore $(G_1, \mathbb{P})$ satisfies the CMC. It can be shown that no graph entails any subset containing two or three of these d-separation rules other than G_1. Hence no graph follows the CFC with respect to $\mathbb{P}$ since there is no graph that entails all three d-separation rules and $(G_1, \mathbb{P})$ satisfies the MDR assumption because no graph entails more or as many d-separation rules as G_1 entails, and satisfies the CMC with respect to $\mathbb{P}$.

(d) Note that G_2 entails the sole d-separation rule, $D_{sep}(G_2) = \{X_1 \text{ d-sep } X_2 \mid X_4\}$ and it is clear that $(G_2, \mathbb{P})$ satisfies the CMC. If $(G_2, \mathbb{P})$ does not satisfy the P-minimality assumption, there exists a graph G such that $(G, \mathbb{P})$ satisfies the CMC and $D_{sep}(G_2) \subset D_{sep}(G)$ and $D_{sep}(G_2) \neq D_{sep}(G)$. It can be shown that no such graph exists. Therefore, $(G_2, \mathbb{P})$ satisfies the P-minimality assumption. Clearly, $(G_2, \mathbb{P})$ fails to satisfy the MDR assumption because G_1 entails more d-separation rules.

□

Theorem 6.4 (a) asserts that whenever the set of DCG models satisfying the CFC is not empty, it is equivalent to the set of DCG models satisfying the MDR assumption.

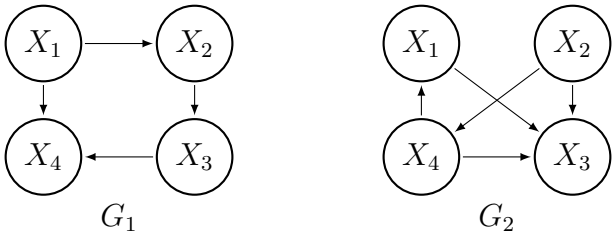


Figure 6.4:: 4-node examples for Theorem 6.4

Part (b) claims that there exists a distribution in which no DCG model satisfies the CFC, while the set of DCG models satisfying the MDR assumption consists of its MEC. Hence, (a) and (b) show that the MDR assumption is strictly superior to the CFC in terms of recovering the true MEC. Theorem 6.4 (c) claims that any DCG models satisfying the MDR assumption should lie in the set of DCG models satisfying the P-minimality assumption. (d) asserts that there exist DCG models satisfying the P-minimality assumption but violating the MDR assumption. Therefore, (c) and (d) prove that the MDR assumption is strictly stronger than the P-minimality assumption.

6.4.2 Comparison between the MDR and SMR assumptions

Now we show that the MDR assumption is neither weaker nor stronger than the SMR assumptions for both DAG and DCG models.

Lemma 6.3. *(a) There exists a DAG model satisfying the identifiable SMR assumption that does not satisfy the MDR assumption. Further, there exists a DAG model satisfying the MDR assumption that does not satisfy the weak SMR assumption.*

(b) There exists a DCG model that is not a DAG that satisfies the same conclusion as (a).

Proof. Our proof for Lemma 6.3 involves us constructing two sets of examples, one for DAGs corresponding to (a) and one for cyclic graphs corresponding to (b). For (a), Figure 6.5 displays two DAGs, G_1 and G_2 which are clearly not in the same

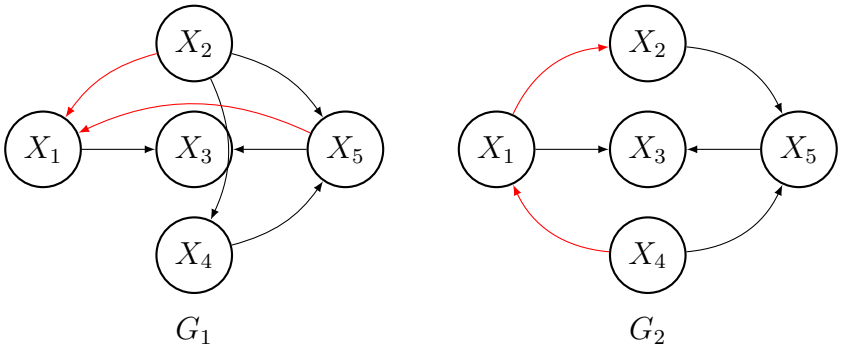


Figure 6.5:: 5-node examples for Lemma 6.3.(a)

MEC. For clarity, we use red arrows to represent the edges/directions that are different between the graphs. We associate the same distribution $\mathbb{P}$ to each DAG where $CI(\mathbb{P})$ is provided in Appendix D.1.2. With this $CI(\mathbb{P})$, both $(G_1, \mathbb{P})$ and $(G_2, \mathbb{P})$ satisfy the CMC (explained in Appendix D.1.2). The main point of this example is that $(G_2, \mathbb{P})$ satisfies the identifiable and weak SMR assumptions whereas $(G_1, \mathbb{P})$ satisfies the MDR assumption, and therefore two different graphs are determined depending on the given identifiability assumption with respect to the same $\mathbb{P}$. A more detailed proof that $(G_1, \mathbb{P})$ satisfies the MDR assumption whereas $(G_2, \mathbb{P})$ satisfies the SMR assumption is provided in Appendix D.1.2.

For (b), Figure 6.6 displays two DCGs G_1 and G_2 which do not belong to the same MEC. Once again red arrows are used to denote the edges (both real and virtual) that are different between the graphs. We associate the same distribution $\mathbb{P}$ with conditional independent statements $CI(\mathbb{P})$ (provided in Appendix D.1.3) to each graph such that both $(G_1, \mathbb{P})$ and $(G_2, \mathbb{P})$ satisfy the CMC (explained in Appendix D.1.3). Again, the main idea of this example is that $(G_1, \mathbb{P})$ satisfies the MDR assumption whereas $(G_2, \mathbb{P})$ satisfies the identifiable SMR assumption. A detailed proof that $(G_1, \mathbb{P})$ satisfies the MDR assumption whereas $(G_2, \mathbb{P})$ satisfies the identifiable SMR assumption can be found in Appendix D.1.3.

□

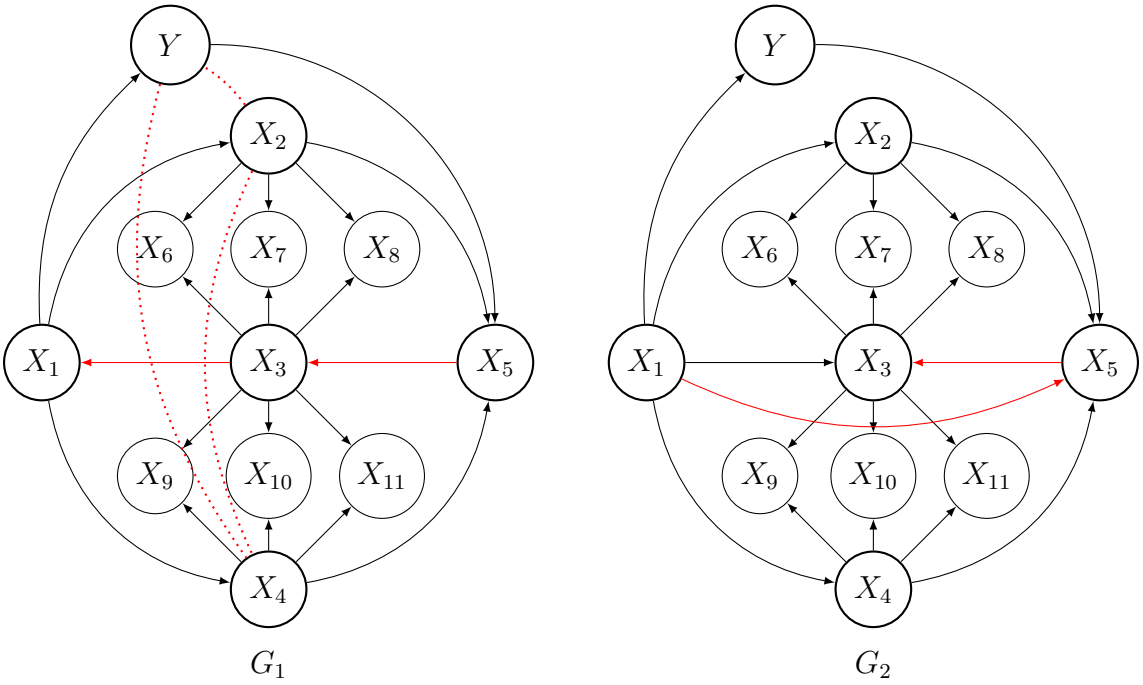


Figure 6.6:: 12-node examples for Lemma 6.3.(b)

Intuitively, the reason why fewer edges does not necessarily translate to entailing more d-separation rules is that the placement of edges relative to the rest of the graph and what additional paths they allow affects the total number of d-separation rules entailed by the graph.

In summary, the flow chart in Figure 6.7 shows how the CFC, SMR, MDR and minimality assumptions are related for both DAG and DCG models:

6.5 Simulation results

In Sections 6.3 and 6.4, we proved that the MDR assumption is strictly weaker than the CFC and stronger than the P-minimality assumption for both DAG and DCG models, and the identifiable SMR assumption is stronger than the P-minimality assumption for DCG models. In this section, we support our theoretical results with numerical experiments on small-scale Gaussian linear DCG models (see e.g., [66]) using the generic Algorithm 6.1. We also provide a comparison of Algorithm 6.1 to state-

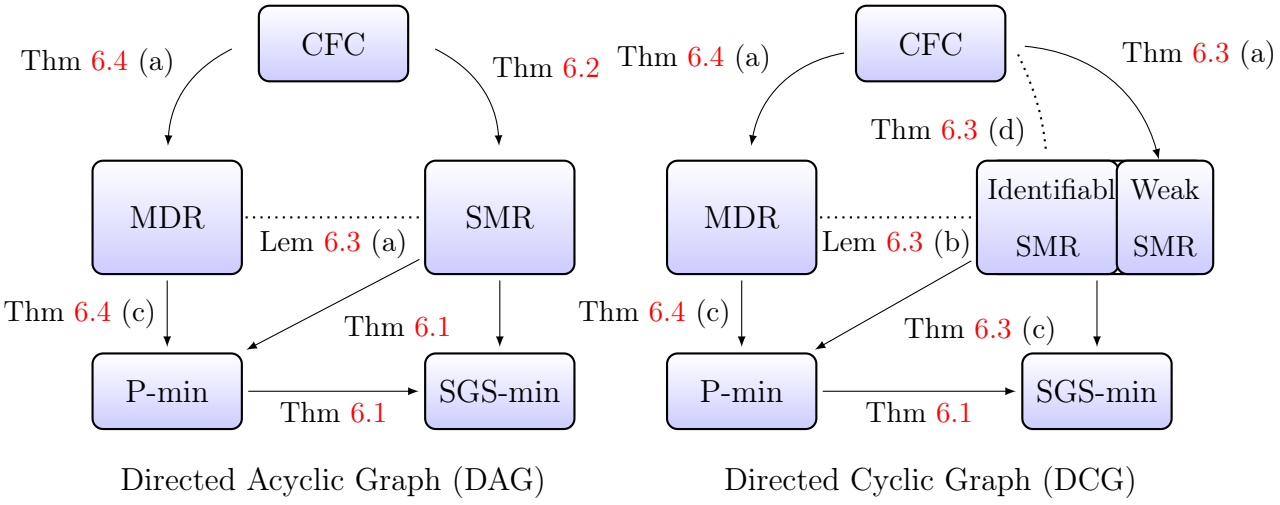


Figure 6.7:: Summary of relationships between assumptions

Algorithm 6.1 Directed Graph Learning Algorithm

Input: iid n samples from the DCG model $(G, \mathbb{P})$
Output: MEC $\widehat{\mathcal{M}}(G)$ and skeleton $\widehat{S}(G)$
Step 1: Find all conditional independence statements $\widehat{CI}(\mathbb{P})$ using a conditional independence test
Find the set of graphs $\widehat{\mathcal{G}}$ satisfying the given identifiability assumption
$\widehat{\mathcal{M}}(G) \leftarrow \emptyset$
$\widehat{S}(G) \leftarrow \emptyset$
if All graphs of $\widehat{\mathcal{G}}$ belong to the same MEC $\mathcal{M}(\widehat{\mathcal{G}})$ **then**
    $\widehat{\mathcal{M}}(G) \leftarrow \mathcal{M}(\widehat{\mathcal{G}})$
end if
if All graphs of $\widehat{\mathcal{G}}$ have the same skeleton $S(\widehat{\mathcal{G}})$ **then**
    $\widehat{S}(G) \leftarrow S(\widehat{\mathcal{G}})$
end if
Return: $\widehat{\mathcal{M}}(G)$ and $\widehat{S}(G)$

of-the-art algorithms for small-scale DCG models in terms of recovering the skeleton of a DCG model.

6.5.1 DCG model and simulation setup

Our simulation study involves simulating DCG models from p-node random Gaussian linear DCG models where the distribution $\mathbb{P}$ is defined by the following linear structural equations:

$$(X_1, X_2, \cdots, X_p)^T = B^T(X_1, X_2, \cdots, X_p)^T + \epsilon \tag{6.2}$$

where $B \in \mathbb{R}^{p\times p}$ is an edge weight matrix with $B_{jk} = \beta_{jk}$ and β_{jk} is a weight of an edge from X_j to X_k. Furthermore, $\epsilon \sim \mathcal{N}(\mathbf{0}_p, I_p)$ where $\mathbf{0}_p = (0, 0, \cdots, 0)^T \in \mathbb{R}^p$ and $I_p \in \mathbb{R}^{p\times p}$ is the identity matrix.

The matrix B encodes the DCG structure since if β_{jk} is non-zero, $X_j \to X_k$ and the pair (X_j, X_k) is *really adjacent*, otherwise there is no directed edge from X_j to X_k. In addition if there is a set of nodes $S = (s_1, s_2, \cdots, s_t)$ such that the product of $\beta_{js_1}, \beta_{ks_1}, \beta_{s_1s_2}, \cdots, \beta_{s_tj}$ is non-zero, the pair (X_j, X_k) is *virtually adjacent*. Note that if the graph is a DAG, we would need to impose the constraint that B is upper triangular; however for DCGs we impose no such constraints.

We present simulation results for two sets of models, DCG models where edges and directions are determined randomly, and DCG models whose edges have a specific graph structure. For the set of random DCG models, the simulation was conducted using 100 realizations of 5-node random Gaussian linear DCG models (6.2) where we impose sparsity by assigning a probability that each entry of the matrix B is non-zero and we set the expected neighborhood size range from 1 (sparse graph) to 4 (fully connected graph) depending on the non-zero edge weight probability. Furthermore the non-zero edge weight parameters were chosen uniformly at random from the range $\beta_{jk} \in [-1, -0.25] \cup [0.25, 1]$ which ensures the edge weights are bounded away from 0.

We also ran simulations using 100 realizations of a 5-node Gaussian linear DCG models (6.2) with specific graph structures, namely trees, bipartite graphs, and cycles. Figure 6.8 shows examples of skeletons of these special graphs. We generate these graphs as follows: First, we set the skeleton for our desired graph based on Figure. 6.8

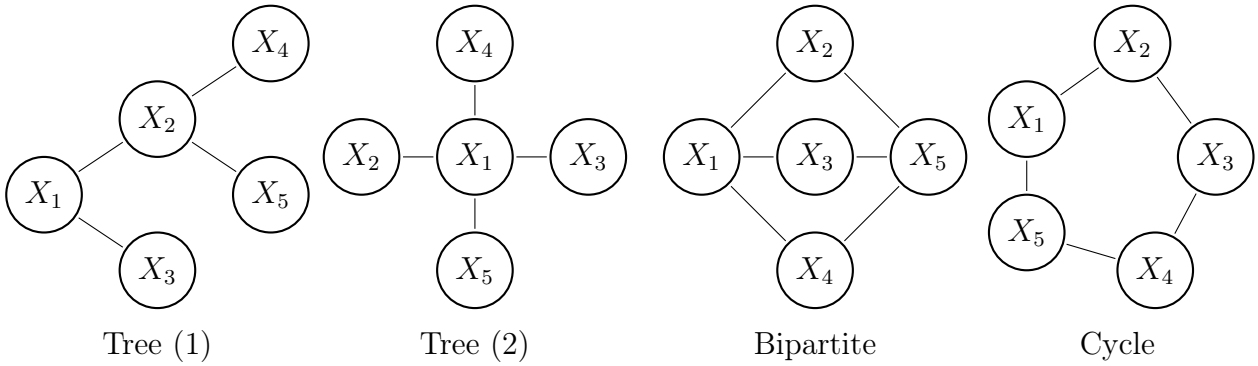


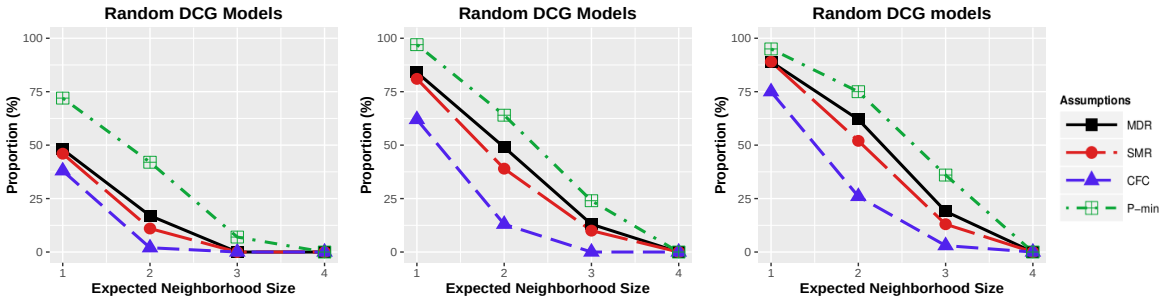
Figure 6.8:: Skeletons of tree, bipartite, and cycle graphs

and then determine the non-zero edge weights which are chosen uniformly at random from the range $\beta_{jk} \in [-1, -0.25] \cup [0.25, 1]$. Second, we repeatedly assign a randomly chosen direction to each edge until every graph has at least one possible directed cycle. Therefore, the bipartite graphs always have at least one directed cycle. However, tree graphs have no cycles because they have no cycles in the skeleton. For cycle graphs, we fix the directions of edges to have a directed cycle $X_1 \to X_2 \to \cdots \to X_5 \to X_1$.

6.5.2 Comparison of assumptions

In this section we provide a simulation comparison between the SMR, MDR, CFC and minimality assumptions. The CI statements were estimated based on n independent samples drawn from $\mathbb{P}$ using Fisher's conditional correlation test with significance level $\alpha = 0.001$. We detected all directed graphs satisfying the CMC and we measured what proportion of graphs in the simulation satisfy each assumption (CFC, MDR, identifiable SMR, P-minimality).

In Figures 6.9, 6.10 and 6.11, we simulated how restrictive each identifiability assumption (CFC, MDR, identifiable SMR, P-minimality) is for random DCG models and specific graph structures with sample sizes $n \in \{100, 200, 500, 1000\}$ and expected neighborhood sizes from 1 (sparse graph) to 4 (fully connected graph). As shown in Figures 6.10 and 6.11, the proportion of graphs satisfying each assumption increases as sample size increases because of fewer errors in CI tests. Furthermore,



(a) $n = 100$ (b) $n = 500$ (c) $n = 1000$

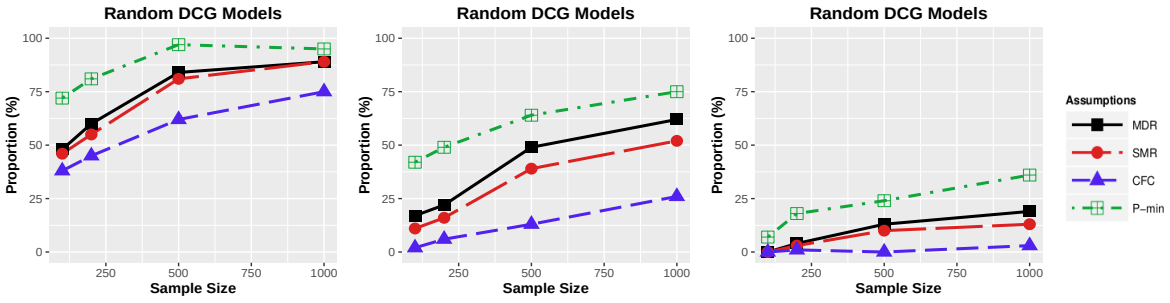
Figure 6.9:: Proportions of 5-node random DCG models satisfying the CFC, MDR, identifiable SMR and P-minimality assumptions with different sample sizes, varying expected neighborhood size

there are more DCG models satisfying the MDR assumption than the CFC and less DCG models satisfying the MDR assumption than the P-minimality assumption for all sample sizes and different expected neighborhood sizes. We can also see similar relationships between the CFC, identifiable SMR and P-minimality assumptions. The simulation study supports our theoretical result that the MDR assumption is weaker than the CFC but stronger than the P-minimality assumption, and the identifiable SMR assumption is stronger than the P-minimality assumption. Although there are no theoretical guarantees that the identifiable SMR assumption is stronger than the MDR assumption and weaker than the CFC, Figures 6.9 and 6.10 represent that the identifiable SMR assumption is substantially stronger than the MDR assumption and weaker than the CFC on average.

6.5.3 Comparison to state-of-the-art algorithms

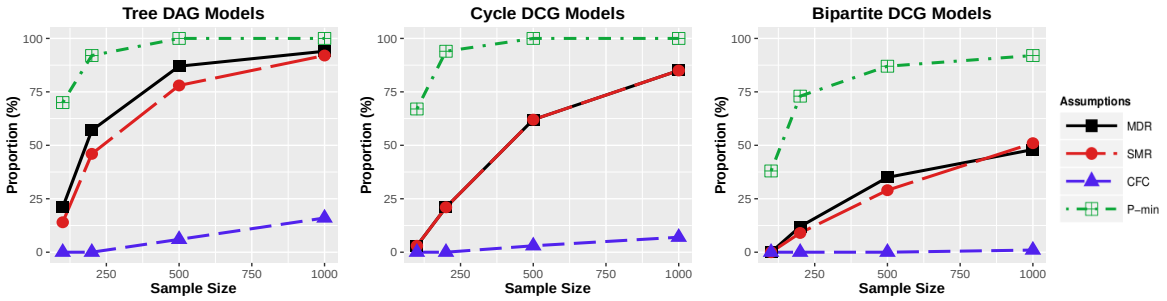
In this section, we compare Algorithm 6.1 to state-of-the-art algorithms for small-scale DCG models in terms of recovering the skeleton $S(G)$ for the graph. This addresses the issue of how likely Algorithm 6.1 based on each assumption is to recover the skeleton of a graph compared to state-of-the-art algorithms.

Once again we used Fisher's conditional correlation test with significance level



(a) Neighborhood sizes: 1 (b) Neighborhood sizes: 2 (c) Neighborhood sizes: 3

Figure 6.10:: Proportions of 5-node random DCG models satisfying the CFC, MDR, identifiable SMR and P-minimality assumptions with different expected neighborhood sizes, varying sample size

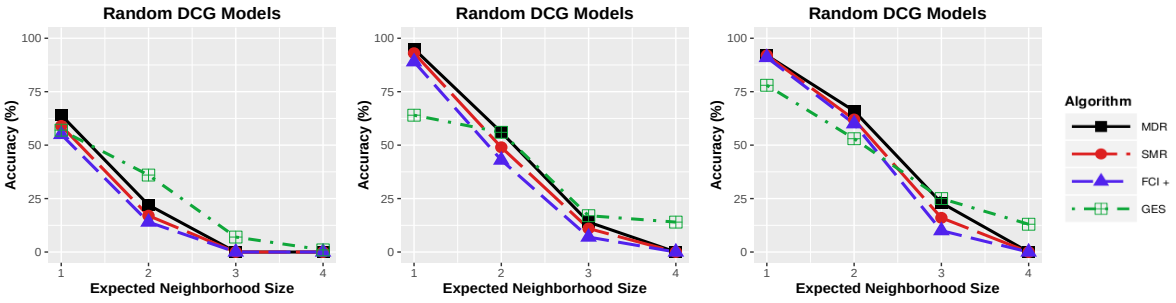


(a) Tree (b) Cycle (c) Bipartite

Figure 6.11:: Proportions of special types of 5-node DAG and DCG models satisfying the CFC, MDR, identifiable SMR, and P-minimality assumptions, varying sample size

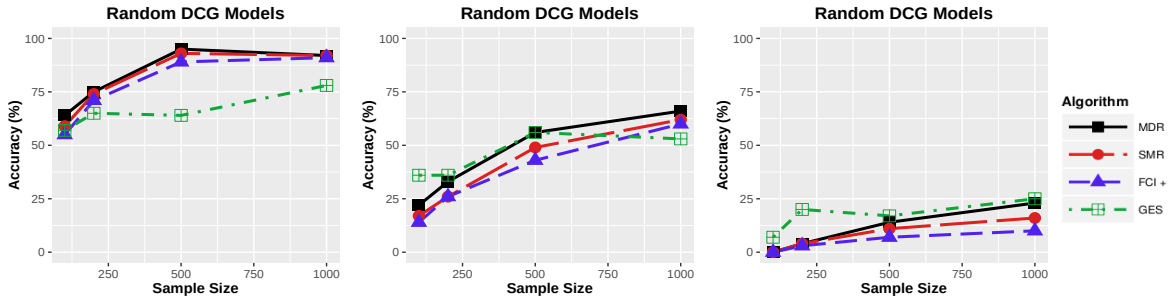
$\alpha = 0.001$ for Step 1) of Algorithm 6.1, and we used the MDR and identifiable SMR assumptions for Step 2). For comparison algorithms, we used the state-of-the-art GES algorithm [11] and the FCI+ algorithms [12] for small-scale DCG models. We used the R package 'pcalg' [**?**] for the FCI+ algorithm, and 'bnlearn' [62] for the GES algorithm.

Figures 6.12 and 6.13 show recovery rates of skeletons for random DCG models with sample sizes $n \in \{100, 200, 500, 1000\}$ and expected neighborhood sizes from 1 (sparse graph) to 4 (fully connected graph). Our simulation results show that the



(a) $n = 100$ (b) $n = 500$ (c) $n = 1000$

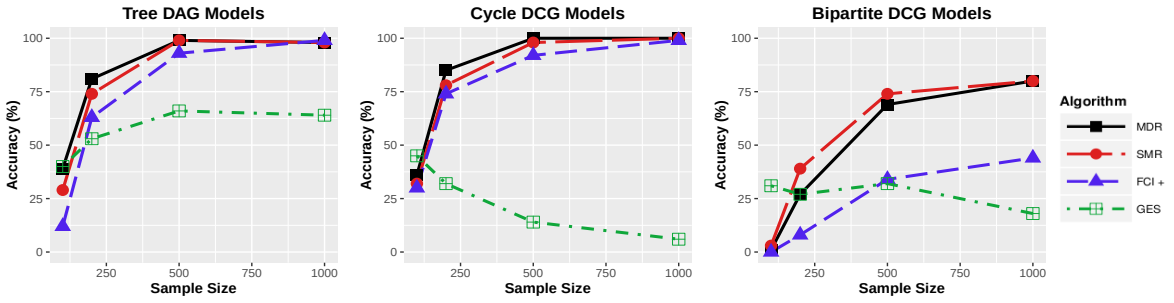
Figure 6.12:: Accuracy rates of recovering skeletons of 5-node random DCG models using the MDR and identifiable SMR assumptions, the GES algorithm, and the FCI+ algorithm with different sample sizes, varying expected neighborhood size



(a) Neighborhood sizes: 1 (b) Neighborhood sizes: 2 (c) Neighborhood sizes: 3

Figure 6.13:: Accuracy rates of recovering skeletons of 5-node random DCG models using the MDR and identifiable SMR assumptions, the GES algorithm, and FCI+ algorithm with different expected neighborhood sizes, varying sample size

accuracy increases as sample size increases because of fewer errors in CI tests. Algorithms 6.1 based on the MDR and identifiable SMR assumptions outperforms the FCI+ algorithm on average. For dense graphs, we see that the GES algorithm outperforms other algorithms because the GES algorithm often prefers dense graphs. However, the GES algorithm is not theoretically consistent and cannot recover directed graphs with cycles while other algorithms are designed for recovering DCG models (see e.g., Figure 6.14).



(a) Tree (b) Cycle (c) Bipartite

Figure 6.14:: Accuracy rates of recovering skeletons of special types of 5-node random DAG and DCG models using the MDR and identifiable SMR assumptions, the GES algorithm, and the FCI+ algorithm, varying sample size

Figure 6.14 shows the accuracy for each type of graph (Tree, Cycle, Bipartite) using Algorithms 6.1 based on the MDR and identifiable SMR assumptions and the GES and the FCI+ algorithms. Simulation results show that Algorithms 6.1 based on the MDR and identifiable SMR assumptions are favorable in comparison to the FCI+ and GES algorithms for small-scale DCG models.

Appendix A

Proofs for Chapter 3

A.1 Proof for Theorem 3.1

Proof. We prove it by induction that requires p steps to find a causal ordering that is consistent with the DAG. Without loss of generality, assume that one of the true causal ordering π^* is $\{1, 2, ...p\}$. For ease of notation, let $\mathcal{F}_s = \{X_1, X_2, \cdots, X_s\}$. Let $k = 1$ be the first step:

$$\text{Var}(X_j) = \mathbb{E}(\text{Var}[X_j|\mathcal{F}_{j-1}]) + \text{Var}(\mathbb{E}[X_j|\mathcal{F}_{j-1}]),$$

where the outer expectation and variance is taken over $X_1, X_2, ..., X_{j-1}$. Since the conditional distribution $X_j|\mathcal{F}_{j-1} \sim \text{Poisson}(g_j(X_{\text{pa}(j)}))$, we have $\text{Var}[X_j|\mathcal{F}_{j-1}] = \mathbb{E}[X_j|\mathcal{F}_{j-1}] = g_j(X_{\text{pa}(j)})$. Hence,

$$\begin{aligned}\text{Var}(X_j) &= \mathbb{E}(\mathbb{E}[X_j|\mathcal{F}_{j-1}]) + \text{Var}(g_j(X_{\text{pa}(j)})) \\ &= \mathbb{E}(X_j) + \text{Var}(g_j(X_{\text{pa}(j)})),\end{aligned}$$

yielding that

$$\text{Var}(X_j) - \mathbb{E}(X_j) = \text{Var}(g_j(X_{\text{pa}(j)})).$$

Clearly, if $\text{pa}(j)$ is empty, meaning the node is the first component of the causal ordering, $\text{Var}(g_j(X_{\text{pa}(j)})) = 0$. Otherwise, $\text{Var}(g_j(X_{\text{pa}(j)})) > 0$ by the assumption.

Hence for any node that can not be the first in the ordering, $\text{Var}(X_j) - \mathbb{E}(X_j) > 0$. Hence we pick any node X_k such that $\text{Var}(X_k) - \mathbb{E}(X_k) = 0$ as being the first element of the causal ordering and X_1 satisfies the above equation.

For $k = m$, assume $X_1, X_2, ..., X_m$ is a valid causal ordering for the first m nodes. Now we consider

$$\text{Var}(X_j|\mathcal{F}_m) = \mathbb{E}(\text{Var}[X_j|\mathcal{F}_{j-1}]|\mathcal{F}_m) + \text{Var}(\mathbb{E}[X_j|\mathcal{F}_{j-1}]|\mathcal{F}_m),$$

for $j = m+1, m+2, ..., p$, where the expectation and variance are taken over the variables $X_1, X_2, ..., X_m$. Again, for any $j = m+1, m+2, ..., p$, we have $\text{Var}[X_j|\mathcal{F}_{j-1}] = \mathbb{E}[X_j|\mathcal{F}_{j-1}] = g_j(X_{\text{pa}(j)})$. Further, since $X_1, X_2, ..., X_m$ is a valid causal ordering for the first m nodes,

$$\begin{aligned}\text{Var}(X_j|\mathcal{F}_m) &= \mathbb{E}(\mathbb{E}[X_j|\mathcal{F}_{j-1}]|\mathcal{F}_m) + \text{Var}(\mathbb{E}(X_j|\mathcal{F}_{j-1})|\mathcal{F}_m) \\ &= \mathbb{E}(X_j|\mathcal{F}_m) + \text{Var}(g_j(X_{\text{pa}(j)})|\mathcal{F}_m).\end{aligned}$$

Hence, following on similar lines,

$$\text{Var}(X_j|\mathcal{F}_m) - \mathbb{E}(X_j|\mathcal{F}_m) = \text{Var}[g_j(X_{\text{pa}(j)})|\mathcal{F}_m].$$

Hence if $\text{pa}(j) \setminus \{1, 2, ..., m\}$ is empty, $\text{Var}(g_j(X_{\text{pa}(j)})|\mathcal{F}_m) = 0$ and $\text{Var}(X_j|\mathcal{F}_m) - \mathbb{E}(X_j|\mathcal{F}_m) = 0$. Any such node can be next on the causal ordering and X_m holds the above property. On the other hand, for any node in which $\text{pa}(j) \setminus \{1, 2, ..., m\}$ is non-empty $\text{Var}(X_j|\mathcal{F}_m) - \mathbb{E}(X_j|\mathcal{F}_m) > 0$ which excludes it from being next in the causal ordering. Hence $X_1, X_2, ..., X_{m+1}$ is a valid causal ordering for the first $m+1$ nodes. This completes the proof by induction. □

A.2 Proof for Theorem 3.3

Proof. Let $X^{(i)} = (X_1^{(i)}, ..., X_p^{(i)})$ be the i.i.d n samples from the given DAG model. Let π^* be a true causal ordering and $\hat{\pi}$ be the estimated causal ordering. Without loss of generality, assume that the true causal ordering π^* is $\{1, 2, ...p\}$. For an arbitrary permutation or causal ordering π, let π_j represent its j^{th} element.

Let E_u denote the set of undirected edges corresponding to the *moralized* graph (i.e. the directed edges without directions and edges between nodes with common children). Recall the definitions $\mathcal{N}(j) := \{k \in \{1, 2, ..., p\} \mid (j, k) \in E_u\}$ denote the neighborhood set of j in the moralized graph and $K(j) = \{k | k \in \mathcal{N}(j-1) \cap \{j, ..., p\}\}$ denote a candidate set for π_j and $C_{jk} = \mathcal{N}(k) \cap \{\pi_1, \pi_2, ..., \pi_{j-1}\}$ which is the intersection of the neighbors of k with $\{1, 2, ..., j-1\}$.

Recall that for ease of notation for any $j \in \{1, 2, ...p\}$, and $S \subset \{1, 2, ..., p\}$ let $\mu_{j|S}$ and represent $\mathbb{E}[X_j|X_S]$ and $\sigma^2_{j|S} = \mathrm{Var}(X_j|X_S)$. Also, denote $\mu_{j|S}(x_S)$ and represent $\mathbb{E}[X_j|X_S = x_S]$ and $\sigma^2_{j|S}(x_S) = \mathrm{Var}(X_j|X_S = x_S)$. Let $n_S(x_S) = \sum_{i=1}^n \mathbf{1}(X_S^{(i)} = x_S)$ and $n_S = \sum_{x_S} n(x_S)\mathbf{1}(n(x_S) \geq c_0.n)$ for an arbitrary $c_0 \in (0, 1)$.

The overdispersion score of $k \in K(j)$ for the j^{th} component of the causal ordering, defined in the second step of our ODS algorithm only considers elements of $\mathcal{X}(\widehat{C}_{jk}) = \{x \in \{X^{(1)}_{\widehat{C}_{jk}}, X^{(2)}_{\widehat{C}_{jk}}, ..., X^{(n)}_{\widehat{C}_{jk}}\} \mid n(x) \geq c_0.n\}$ so we only count up elements that occur sufficiently frequently.

According to the ODS algorithm, the truncated sample conditional expectation and variance of X_j given $X_S = x$ for $j \in \{1, 2, ...p\}$ and any subset $S \subset \{1, 2, ...p\} \setminus \{j\}$ be following: for $x \in \mathcal{X}(S)$,

$$\widehat{\mu}_{j|S}(x) = \frac{1}{n_S(x)} \sum_{i=1}^n X_j^{(i)} \mathbf{1}(X_S^{(i)} = x)$$

$$\widehat{\sigma}^2_{j|S}(x) = \frac{1}{n_S(x) - 1} \sum_{i=1}^n (X_j^{(i)} - \widehat{\mu}_{j|S}(x))^2 \mathbf{1}(X_S^{(i)} = x)$$

The overdispersion score of $k \in K(j)$ for the j^{th} element of the causal ordering is for $x \in \mathcal{X}(C_{jk})$,

$$\widehat{s}_{jk}(x) = \widehat{\sigma}^2_{k|\widehat{C}_{jk}}(x) - \widehat{\mu}_{k|\widehat{C}_{jk}}(x)$$

$$\widehat{s}_{jk} = \widehat{\mathbb{E}}_{\widehat{C}_{jk}}(\widehat{s}_{jk}(x)) = \sum_{x \in \mathcal{X}(jk)} \frac{n_{\widehat{C}_{jk}}(x)}{n_{\widehat{C}_{jk}}} \widehat{s}_{jk}(x).$$

And the correct overdispersion score is

$$s^*_{jk} = \mathbb{E}_{C_{jk}}[\sigma^2_{k|C_{jk}} - \mu_{k|C_{jk}}] = \mathbb{E}_{C_{jk}}[\mathrm{Var}(g_k(\mathrm{pa}(k))|C_{jk})].$$

Let us define some events for the proof and d denote the maximum degree of the moralized graph. For any $j \in \{1, 2, ..., p\}$ and $k \in K(j)$,

$$\xi_1 = \{\max_{j,k} |\widehat{s}_{jk} - s^*_{jk}| < m/2\}$$

$$\xi_2 = \{\max_{k} \max_{i=1,...,n} X_k^{(i)} < n^{\frac{1}{5+d}}\}$$

We prove it by induction that requires p steps to recover a causal ordering that is consistent with the Poisson DAG. Without loss of generality, assume that the true causal ordering π^* is $\{1, 2, ...p\}$. For the first step $j = 1$, a set of candidate element of π_1 is $K(1) = \{1, 2, ..., p\}$ and a candidate parent set of each node $C_{1k} = \emptyset$ for all $k \in K(1)$.

$$\begin{aligned} P(\widehat{\pi}_1 \neq \pi^*_1) &= P\big(\text{exists at least one } k \in K(1) \setminus \{1\} \text{ s.t. } \widehat{s}_{11} > \widehat{s}_{1k}\big) \\ &\leq |K(1)| \max_{k \in K(1)\setminus\{1\}} \big\{P\big(s^*_{11} + \frac{m}{2} > s^*_{1k} - \frac{m}{2}|\xi_1\big) + P(\xi_1^c|\xi_2) + P(\xi_2^c)\big\} \\ &\leq p \max_{k \in K(1)\setminus\{1\}} \big\{P\big(m > s^*_{1k}|\xi_1\big) + P(\xi_1^c|\xi_2) + P(\xi_2^c)\big\} \end{aligned}$$

By Assumption (A1), $s^*_{1k} > m$ and we will represent some Propositions that respectively control $P(\xi_1^c|\xi_2)$ and $P(\xi_2^c)$.

For the $j-1$ step, assume $(\widehat{\pi}_1, \widehat{\pi}_2, ..., \widehat{\pi}_{j-1})$ is a valid ordering for the first $j-1$ nodes. Note that with the correct $\mathcal{N}(j)$, $\widehat{C}_{jk} = C_{jk}$. Now, we consider π^*_j. The probability of a false recovery of π^*_j given the true undirected edges of the moralized graph and the true causal ordering before j is following:

$$\begin{aligned} &\mathrm{P}(\widehat{\pi}_j \neq \pi^*_j | \widehat{\pi}_1 = \pi^*_1, ..., \widehat{\pi}_{j-1} = \pi^*_{j-1}) \\ &= P\big(\text{exists at least one } k \in K(j) \setminus \{j\} \text{ s.t. } \widehat{s}_{jj} > \widehat{s}_{jk}\big) \\ &\leq |K(j)| \max_{k \in K(j)\setminus\{j\}} \big\{P\big(\widehat{s}_{jj} + m/2 > s^*_{jk} - m/2|\xi_1\big) + P(\xi_1^c|\xi_2) + P(\xi_2^c)\big\} \\ &\leq |K(j)| \max_{k \in K(j)\setminus\{j\}} \big\{P\big(m > s^*_{jk}|\xi_1\big) + P(\xi_1^c|\xi_2) + P(\xi_2^c)\big\} \end{aligned}$$

By Assumption (A1), $s^*_{jk} > m$ and we represent some Propositions that respectively control $P(\xi_1^c|\xi_2)$ and $P(\xi_2^c)$. Furthermore we also show a condition on c_0.

Proposition A.1. *For all $j \in \{1, 2, ..., p\}, k \in K(j)$, $c_0 \leq n^{-\frac{d}{5+d}}$ given ξ_2 is a sufficient that a candidate parents set $\mathcal{X}(C_{jk})$ is not empty*

Proposition A.2.

$$P(\xi_1^c|\xi_2) \leq 2p^2 n^{\frac{d}{5+d}} \Big\{ exp\big(-\frac{m^2 n^{1/(5+d)}}{18}\big) + exp\big(-\frac{m^2 n^{1/(5+d)}}{9}\big) + exp\big(-\frac{m^2 n^{3/(5+d)}}{9}\big)\Big\},$$

where m is the constant in Assumption (A1).

Proposition A.3.

$$P(\xi_2^c) \leq npM\, exp\big(-n^{1/(5+d)} \log 2\big)$$

where M is the constant in Assumption (A2).

Hence for any $j \in \{1, 2, ...p\}$ with $c_0 = n^{-\frac{d}{5+d}}$,

$$\begin{aligned}
&\mathrm{P}(\widehat{\pi}_j \neq \pi_j^* | \widehat{\pi}_1 = \pi_1^*, ..., \widehat{\pi}_{j-1} = \pi_{j-1}^*) \\
&\leq p \max_{k \in K(j)\setminus\{j\}} \big\{ P\big(m > s_{jk}^* | \xi_1\big) + P(\xi_1^c|\xi_2) + P(\xi_2^c)\big\} \\
&\leq 2p^3 n^{\frac{d}{5+d}} \big\{ \exp\big(-\frac{m^2 n^{1/(5+d)}}{18}\big) + \exp\big(-\frac{m^2 n^{1/(5+d)}}{9}\big) + \exp\big(-\frac{m^2 n^{3/(5+d)}}{9}\big)\big\} \\
&+ np^2 M \exp\big(-n^{1/(5+d)} \log 2\big) \qquad \text{(A.1)}
\end{aligned}$$

By using the above probability bound (A.1),

$$\begin{aligned}
P(\widehat{\pi} \neq \pi^*) &\overset{(E_1)}{\leq} P(\widehat{\pi}_1 \neq \pi_1^*) + ... + P(\widehat{\pi}_{p-1} \neq \pi_{p-1}^* | \widehat{\pi}_1 = \pi_1^*, ..., \widehat{\pi}_{p-2} = \pi_{p-2}^*) \\
&\overset{(E_2)}{\leq} 2p^4 n^{\frac{d}{5+d}} \big\{ \exp\big(-\frac{m^2 n^{1/(5+d)}}{18}\big) + \exp\big(-\frac{m^2 n^{1/(5+d)}}{9}\big) + \exp\big(-\frac{m^2 n^{3/(5+d)}}{9}\big)\big\} \\
&+ np^3 M \exp\big(-n^{1/(5+d)} \log 2\big)
\end{aligned}$$

The first inequality (E_1) is followed from $P(A \cup B) = P(A) + P(B \cap A^c) = P(A) + P(B \mid A^c)P(A^c) \leq P(A) + P(B \mid A^c)$ for some events A, B. And (E_2) is directly from (A.1).

Hence, there exists some positive constants $C_1, C_2, C_3 > 0$ such that

$$P(\hat{\pi} \neq \pi^*) \leq C_1 \exp\big(-C_2 n^{1/(5+d)} + C_3 \log \max\{p, n\}\big)$$

□

A.2.1 Proof for Proposition A.1

Proof. Let $|X_S|$ denote the cardinality of a set $\{X_S^{(1)}, X_S^{(2)}, ..., X_S^{(n)}\}$ and $|\mathcal{X}(S)|$ denote the cardinality of a set $\mathcal{X}(S)$. In worst case where $|\mathcal{X}(S)| = 1$, for all $x \in \{X_S^{(1)}, X_S^{(2)}, ..., X_S^{(n)}\}$, $n_S(x) = c_0.n - 1$ except for only one component $y \in \mathcal{X}(S)$. In this case, the sample size $n = n_S(y) + (|X_S| - 1)(c_0.n - 1)$. A simple calculation yields that

$$n_S(y) = n - (|X_S| - 1)(c_0.n - 1) = n - c_0.n|X_S| + c_0.n + |X_S| - 1.$$

Hence $c_0.n \leq n_S(y)$ is equivalent to $c_0 \leq \frac{n+|X_S|-1}{n.|X_S|}$. Since $\frac{1}{|X_S|} \leq \frac{n+|X_S|-1}{n|X_S|}$, if $c_0 \leq \frac{1}{|X_S|}$ there exists at least one component $y \in \mathcal{X}(S)$. In addition under the event ξ_2, $|X_S| \leq n^{\frac{d}{5+d}}$ which is all possible combinations. Hence if $c_0 \leq n^{-\frac{d}{5+d}}$, $|\mathcal{X}(S)| \neq 0$. □

A.2.2 Proof for Proposition A.2

Proof. This problem is reduced to the consistency rate of a sample conditional mean and conditional variance. For ease of notation, let $n_{jk} = n_{C_{jk}}$ and $n_{jk}(x) = n_{C_{jk}}(x)$. Suppose that $c_0 = n^{-\frac{d}{5+d}}$. Then for any $j \in \{1, 2, ..., p\}$ and $k \in K(j)$,

$$
\begin{aligned}
&P(\xi_1^c, \xi_2) \leq p^2 \max_{j,k} P(|\widehat{s}_{jk} - s^*_{jk}| > \frac{m}{2}, \xi_2) \\
&\leq p^2 \max_{j,k} P(\sum_{x \in \mathcal{X}(C_{jk})} \frac{n_{jk}(x)}{n_{jk}} |\widehat{s}_{jk}(x) - s^*_{jk}(x)| > \frac{m}{2}, \xi_2) \\
&\overset{(a)}{\leq} p^2 \max_{j,k} \sum_{x \in \mathcal{X}(C_{jk})} P(|\widehat{s}_{jk}(x) - s^*_{jk}(x)| > \frac{m}{2} \frac{n_{jk}}{n_{jk}(x)}, \xi_2) \\
&\overset{(b)}{\leq} p^2 \max_{j,k} |\mathcal{X}(C_{jk})| \max_{x \in \mathcal{X}(C_{jk})} P(|\widehat{s}_{jk}(x) - s^*_{jk}(x)| > \frac{m}{2}, \xi_2) \\
&\overset{(c)}{\leq} p^2 n^{\frac{d}{5+d}} \max_{j,k,x} P(|(\widehat{\sigma}^2_{k|C_{jk}}(x) - \widehat{\mu}_{k|C_{jk}}(x)) - (\sigma^2_{k|C_{jk}}(x) - \mu_{k|C_{jk}}(x))| > \frac{m}{2}, \xi_2) \\
&\leq p^2 n^{\frac{d}{5+d}} \max_{j,k,x} \big\{ P\big(|\widehat{\sigma}^2_{k|C_{jk}}(x) - \sigma^2_{j|C_{jk}}(x)| > \frac{m}{3}, \xi_2\big) \\
&\qquad\qquad + P\big(|\widehat{\mu}_{k|C_{jk}}(x) - \mu_{k|C_{jk}}(x)| > \frac{m}{6}, \xi_2\big) \big\} \\
&\overset{(d)}{\leq} 2p^2 n^{\frac{d}{5+d}} \max_{j,k,x} \big\{ \exp\big(- \frac{m^2 n_{jk}(x)}{18 n^{4/(5+d)}}\big) + \exp\big(- \frac{m^2 n_{jk}(x)}{9 n^{4/(5+d)}}\big) + \exp\big(- \frac{m^2 n_{jk}(x)}{9 n^{2/(5+d)}}\big) \big\} \\
&\overset{(e)}{\leq} 2p^2 n^{\frac{d}{5+d}} \max_{j,k,x} \big\{ \exp\big(- \frac{m^2 n^{1/(5+d)}}{18}\big) + \exp\big(- \frac{m^2 n^{1/(5+d)}}{9}\big) + \exp\big(- \frac{m^2 n^{3/(5+d)}}{9}\big) \big\} \\
&= 2p^2 n^{\frac{d}{5+d}} \big\{ \exp\big(- \frac{m^2 n^{1/(5+d)}}{18}\big) + \exp\big(- \frac{m^2 n^{1/(5+d)}}{9}\big) + \exp\big(- \frac{m^2 n^{3/(5+d)}}{9}\big) \big\}.
\end{aligned}
$$

(a) is followed from that $P(\sum_i \omega_i X_i > \delta) \leq \sum_i P(X_i > \delta/\omega_i)$, and (b) is from $\frac{n_{jk}(x)}{n_{jk}} < 1$. Since $n_{jk}(x) \geq c_0.n$ for all $x \in \mathcal{X}(C_{jk})$, $|\mathcal{X}(C_{jk})| \leq 1/c_0$ hence (c) and (e) hold. Moreover, (d) is followed from the Hoeffding's inequality (Theorem 2 [30]) since samples are independent and bounded above $n^{1/(5+d)}$ given ξ_2. □

A.2.3 Proof for Proposition A.3

Proof. For any $j \in \{1, 2, ..., p\}$, the conditional distribution of X_j given $X_{\mathrm{pa}(j)}$ is Poisson with rate parameter $g_j(\mathrm{pa}(j))$. Hence for $k \in K(j)$,

$$
\begin{aligned}
P(\xi_2^c) &= P(\max_{k\in K(j)} \max_{i=1,\dots,n} X_k^{(i)} > n^{1/(5+d)}) \\
&\overset{(a)}{\leq} np \max_{k\in K(j)} \max_{i=1,\dots,n} P(|X_k^{(i)}| > n^{1/(5+d)}) \\
&\overset{(b)}{\leq} np \max_{k\in K(j)} \max_{i=1,\dots,n} \mathbb{E}_{\mathrm{pa}(k)}\big[\exp\big(-n^{1/(5+d)}\log 2 + g_k(\mathrm{pa}(k))\big)\big] \\
&\overset{(c)}{\leq} np \max_{k\in K(j)} \max_{i=1,\dots,n} M\exp(-n^{1/(5+d)}\log 2) \\
&= npM\exp\big(-n^{1/(5+d)}\log 2\big).
\end{aligned}
$$

(a) follows from the union bound and $|K(j)| < p$, and (b) follows from the moment generating function of Poisson distribution with $t = \log 2$. Furthermore, (c) is from Assumption 3.2 (A2). □

Appendix B

Proofs for Chapter 4

B.1 Appendix

B.1.1 Proof for Theorem 4.1

Proof. Without loss of generality, we assume the causal ordering is $\pi^* = (1, 2, \cdots, p)$. For notational convenience, we define $X_{1:j} = \{X_1, X_2, \cdots, X_j\}$ and $X_{1:0} = \emptyset$. for $m \in V$ and $j \in \{m, m+1, \cdots, p\}$, let $c_{jm} = (\beta_0 + \beta_1 \mathbb{E}(X_j \mid X_{1:m-1}))^{-1}$ and $c_{j1} = (\beta_0 + \beta_1 \mathbb{E}(X_j))^{-1}$. Then, the overdispersion score is as follows:

$$\mathcal{S}(j, m) = c_{jm}^2 \mathrm{Var}(X_j \mid X_{1:m-1}) - c_{jm} \mathbb{E}(X_j \mid X_{1:m-1}).$$

We now prove the identifiability of our class of DAG models by induction. For the first step, and $j \in \{1, 2, \cdots, p\}$,

$$\begin{aligned}
&\mathcal{S}(j, 1) = c_{j1}^2 \mathrm{Var}(X_j) - c_{j1} \mathbb{E}(X_j) \\
&\overset{(a)}{=} c_{j1}^2 \big\{ \mathrm{Var}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)})) + \mathbb{E}(\mathrm{Var}(X_j \mid X_{\mathrm{pa}(j)})) - c_{j1}^{-1} \mathbb{E}(X_j) \big\} \\
&\overset{(b)}{=} c_{j1}^2 \big\{ \mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)})) + \mathbb{E}(\beta_0 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) + \beta_1 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)})^2) - (\beta_0 + \beta_1 \mathbb{E}(X_j)) \mathbb{E}(X_j) \big\} \\
&= c_{j1}^2 \big\{ \mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)})) + \beta_1 \mathbb{E}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)})^2) - \beta_1 \mathbb{E}(X_j)^2 \big\} \\
&= c_{j1}^2 (1 + \beta_1) \mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)})).
\end{aligned}$$

(a) follows from the variance decomposition formula $\mathrm{Var}[Y] = \mathbb{E}(\mathrm{Var}[Y \mid X]) + \mathrm{Var}(\mathbb{E}[Y \mid X])$ for some random variables X and Y. In addition (b) follows from

the quadratic variance property (4.1) of our class of distributions and the definition of c_{j1}. Note that the score of the true first element of the causal ordering is $\mathcal{S}(1,1) = 0$ because $E(X_1 \mid X_{\mathrm{pa}(1)})$ is a constant and other scores are strictly positive $\mathcal{S}(j,1) > 0$ by the identifiability assumption in Theorem 4.8. Therefore we can choose the first element of the causal ordering.

For $(m-1)^{th}$ step, assume that first $m-1$ elements of the causal ordering are correctly estimated. Now, we consider m^{th} step. Then, for $j \in \{m, m+1, \cdots, p\}$,

$$
\begin{aligned}
&\mathcal{S}(j,m) = c_{jm}^2 \mathrm{Var}(X_j \mid X_{1:m-1}) - c_{jm}\mathbb{E}(X_j \mid X_{1:m-1}) \\
&\overset{(a)}{=} c_{jm}^2 \big\{ \mathrm{Var}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}) \mid X_{1:m-1}) + \mathbb{E}(\mathrm{Var}(X_j \mid X_{\mathrm{pa}(j)}) \mid X_{1:m-1}) - c_{jm}^{-1}\mathbb{E}(X_j \mid X_{1:m-1}) \big\} \\
&\overset{(b)}{=} c_{jm}^2 \big\{ \mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_{1:m-1}) + \mathbb{E}(\beta_0 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)} \mid X_{1:m-1}) + \beta_1 \mathbb{E}(X_j \mid X_{\mathrm{pa}(j)} \mid X_{1:m-1})^2) \\
&\qquad\qquad - (\beta_0 + \beta_1 \mathbb{E}(X_j \mid X_{1:m-1}))\mathbb{E}(X_j \mid X_{1:m-1}) \big\} \\
&= c_{jm}^2 \big\{ \mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_{1:m-1}) + \beta_1 \mathbb{E}(\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)})^2 \mid X_{1:m-1}) - \beta_1 \mathbb{E}(X_j \mid X_{1:m-1})^2 \big\} \\
&= c_{jm}^2 (1+\beta_1) \mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_{1:m-1}).
\end{aligned}
$$

Again (a) follows from the variance decomposition formula and (b) follows from the quadratic variance property (4.1) of our class of distributions and the definition of c_{jm}.

If $\mathrm{pa}(j) \setminus \{1, 2, \cdots, m-1\}$ is empty, $\mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_{1:m-1}) = 0$, and hence $\mathcal{S}(m,m) = 0$. On the other hand, for any node j in which $\mathrm{pa}(j) \setminus \{1, 2, \cdots, m-1\}$ is non-empty, $\mathcal{S}(j,m) > 0$ by the identifiability assumption in Theorem 4.8, which excludes it from being next in the causal ordering. Therefore, we can estimate a valid m^{th} component of the causal ordering, $\widehat{\pi}_m = m$. This completes the proof by induction. □

B.1.2 Proof for Lemma 4.1

Proof. For any $k \notin \mathrm{pa}(j)$ $[\theta_D^*]_k = 0$ by the construction of θ_D^*. Secondly, we show that for any $k \in \mathrm{pa}(j)$, $[\theta_D^*]_k \neq 0$. Assume for the sake of contradiction that $[\theta_D^*]_k = 0$.

By the first order optimality condition, we have

$$\begin{aligned} \mathbb{E}(X_j) &= \mathbb{E}(D'(\langle \theta_D^*, X_{\mathrm{pa}(j)} \rangle)) \\ \mathbb{E}(X_j X_k) &= \mathbb{E}(D'(\langle \theta_D^*, X_{\mathrm{pa}(j)} \rangle) X_k). \end{aligned} \tag{B.1}$$

By the definition of the covariance, we obtain

$$\mathbb{E}(X_j X_k) = \mathrm{Cov}(D'(\langle \theta_D^*, X_{1:j-1} \rangle), X_k) - \mathbb{E}(D'(\langle \theta_D^*, X_{1:j-1} \rangle))\mathbb{E}(X_k).$$

Equation (B.1) implies that

$$\mathbb{E}(X_j X_k) = \mathrm{Cov}(D'(\langle \theta_D^*, X_{1:j-1} \rangle), X_k) - \mathbb{E}(X_j)\mathbb{E}(X_k).$$

Hence, we have

$$\mathrm{Cov}(X_j, X_k) = \mathrm{Cov}(X_k, D'(\langle \theta_D^*, X_{\mathrm{pa}(j)} \rangle)).$$

From the assumption that $[\theta_D^*]_k = 0$, we obtain

$$\mathrm{Cov}(X_j, X_k) = \mathrm{Cov}(X_k, D'(\langle [\theta_D^*]_{\mathrm{pa}(j)\setminus k}, X_{\mathrm{pa}(j)\setminus j} \rangle)).$$

However it is contradictory to the assumption $\mathrm{Cov}(X_j, X_k) \neq \mathrm{Cov}(X_k, D'(\langle [\theta_D^*]_{\mathrm{pa}(j)\setminus k}, X_{\mathrm{pa}(j)\setminus j} \rangle))$. Therefore $[\theta_D^*]_k \neq 0$. Furthermore since $k \in \mathrm{pa}(j)$ is arbitrary, the proof is complete. □

B.1.3 Proof for Theorem 4.6

Proof. Assume that there are n iid samples $x = \{X^{(1)}, X^{(2)}, \cdots, X^{(n)}\}$ and $X^{(i)} = \{X_1^{(i)}, X_2^{(i)} \cdots, X_p^{(i)}\}$ for $i \in \{1, 2, \cdots, n\}$ from a given DAG model $(G, \mathbb{P})$. For ease of notation, let $\S = \mathcal{N}(j)$ for a node $j \in V$ and recall that $\langle \cdot, \cdot \rangle$ represents the inner product and $[\cdot]_k$ is an element of a vector corresponding to a variable X_k. Then, the negative surrogate conditional log-likelihood of GLM (4.8) is as follows:

$$\ell(\theta; x) := \frac{1}{n} \sum_{i=1}^{n} \left(-X_j^{(i)} \langle \theta, X_{V\setminus j}^{(i)} \rangle + D(\langle \theta, X_{V\setminus j}^{(i)} \rangle) \right)$$

where $D(\cdot)$ is the log-normalization constant determined by the choice of GLM and $\theta \in \mathbb{R}^{p-1}$.

The main goal of the proof is to find the unique minimizer of the following convex problem:

$$\widehat{\theta}_M := \arg\min_{\theta \in \mathbb{R}^{p-1}} \mathcal{L}(\theta, \lambda_n) = \arg\min_{\theta \in \mathbb{R}^{p-1}} \{\ell(\theta; x) + \lambda_n \|\theta\|_1\}. \tag{B.2}$$

By the *sub-differential* method, $\widehat{\theta}_M$ must hold the following condition:

$$\bigtriangledown_\theta \mathcal{L}(\widehat{\theta}_M, \lambda_n) = \bigtriangledown_\theta \ell(\widehat{\theta}_M; x) + \lambda_n \widehat{z} = 0 \tag{B.3}$$

where $\widehat{z} \in \mathbb{R}^{p-1}$ and an element of $\widehat{z}$ corresponding to a parameter $[\widehat{\theta}_M]_t$ is $\widehat{z}_t = \text{sign}([\widehat{\theta}_M]_t)$ if a node $t \in \S$ otherwise $|\widehat{z}_t| < 1$.

Main idea of the proof is *primal-dual-witness* method which asserts that there is a dual problem $\widetilde{\theta}_M = \widehat{\theta}_M$ if the following conditions are satisfied:

(a) We determine the vector $\widetilde{\theta}_M \in \Theta$ where $\Theta = \{\theta \in \mathbb{R}^{p-1} : \theta_{\S^c} = 0\}$ by solving the following restricted objective problem.

$$\widetilde{\theta}_M := \arg\min_{\theta \in \Theta} \mathcal{L}(\theta, \lambda_n) = \arg\min_{\theta \in \Theta} \{\ell(\theta; x) + \lambda_n \|\theta\|_1\}. \tag{B.4}$$

(b) We choose $\widetilde{z}$ as a member of the sub-differential of regularizer $\|\cdot\|_1$ evaluated by $\widetilde{\theta}_M$.

(c) For any $t \in \S$, $\widetilde{z}_t = \text{sign}([\widetilde{\theta}_M]_t)$.

(d) For any $t \notin \S$, $|\widetilde{z}_t| < 1$.

If all conditions (a), (b), (c), and (d) are satisfied, $\widehat{\theta}_M = \widetilde{\theta}_M$, meaning that the solution of the unrestricted problem (B.2) is the same as the solution of the restricted problem (B.4). The conditions (a), (b) and (c) suffice to obtain a pair $(\widetilde{\theta}_M, \widetilde{z})$ that satisfies the optimality condition (B.3), but do not guarantee that $\widetilde{z}$ is an element of the sub-differential $\|\widetilde{\theta}_M\|_1$. Therefore, the remainder of the proof is to show $|\widetilde{z}_t| < 1$ for all $t \notin \S$.

Equation (B.3) with the dual solution $(\widetilde{\theta}_M, \widetilde{z})$ can be represented as $\bigtriangledown^2 \ell(\theta^*_M; x)(\widetilde{\theta}_M - \theta^*_M) = -\lambda_n \widetilde{z} - W^n + R^n$ by using mean value theorem where

(a) W^n is the sample score function.

$$W^n := - \bigtriangledown \ell(\theta^*_M; x) \tag{B.5}$$

(b) $R^n = (R^n_1, R^n_2, \cdots, R^n_{p-1})$ and R^n_k is the remainder term by applying coordinate-wise mean value theorem.

$$R^n_k := [\bigtriangledown^2 \ell(\theta^*_M; x) - \bigtriangledown^2 \ell(\bar{\theta}^{(k)}; x)]^T_k (\widetilde{\theta}^{(k)} - \theta^*_M) \tag{B.6}$$

where $\bar{\theta}^{(k)}$ is a vector on the line between $\widetilde{\theta}$ and θ^*_M and $[\cdot]^T_k$ is the k^{th} row of a matrix.

Recall that $Q = \bigtriangledown^2 \ell(\theta^*_M; x)$ be the Hessian of the negative conditional log-likelihood of a GLM and $Q_{\S\S}$ be a sub-matrix corresponding to variables $X_\S$. In addition we use $\widetilde{\theta}_\S = [\widetilde{\theta}_M]_\S$ and $\widetilde{\theta}_{\S^c} = [\widetilde{\theta}_M]_{\S^c}$. Since the set $[\widetilde{\theta}_M]_{\S^c} = 0$ in our primal-dual construction, we can re-state the condition of (B.3) in a block form as follows:

$$\begin{aligned} Q_{\S^c\S}[\widetilde{\theta}_\S - \theta^*_\S] &= W^n_{\S^c} - \lambda_n \widetilde{z}_{\S^c} + R^n_{\S^c}. \\ Q_{\S\S}[\widetilde{\theta}_S - \theta^*_S] &= W^n_\S - \lambda_n \widetilde{z}_\S + R^n_\S. \end{aligned}$$

Since the matrix $Q_{\S\S}$ is invertible, the above equations can be rewritten as

$$Q_{\S^c\S} Q^{-1}_{\S\S} [W^n_\S - \lambda_n \widetilde{z}_\S - R^n_\S] = W^n_{\S^c} - \lambda_n \widetilde{z}_{\S^c} - R^n_{\S^c}.$$

It implies that

$$[W^n_{\S^c} - R^n_{\S^c}] - Q_{\S^c\S} Q^{-1}_{\S\S} [W^n_\S - R^n_\S] + \lambda_n Q_{\S^c\S} Q^{-1}_{\S\S} \widetilde{z}_\S = \lambda_n \widetilde{z}_{\S^c}.$$

Taking the ℓ_∞ norm of both sides yields

$$\|\widetilde{z}_{\S^c}\|_\infty \leq |||Q_{\S^c\S} Q^{-1}_{\S\S}|||_\infty \left[\frac{\|W^n_\S\|_\infty}{\lambda_n} + \frac{\|R^n_\S\|_\infty}{\lambda_n} + 1 \right] + \frac{\|W^n_{\S^c}\|_\infty}{\lambda_n} + \frac{\|R^n_{\S^c}\|_\infty}{\lambda_n}.$$

Recalling Assumptions (4.3), we obtain $|||Q_{\S^c\S}Q_{\S\S}^{-1}|||_\infty \leq (1-\alpha)$, hence we have

$$\begin{aligned}\|\widetilde{z}_{\S^c}\|_\infty &\leq (1-\alpha)\left[\frac{\|W^n_\S\|_\infty}{\lambda_n}+\frac{\|R^n_\S\|_\infty}{\lambda_n}+1\right]+\frac{\|W^n_{\S^c}\|_\infty}{\lambda_n}+\frac{\|R^n_{\S^c}\|_\infty}{\lambda_n}\\ &\leq (1-\alpha)+(2-\alpha)\left[\frac{\|W^n\|_\infty}{\lambda_n}+\frac{\|R^n\|_\infty}{\lambda_n}\right].\end{aligned}$$

We need the following three lemmas to show $\|\widetilde{z}_{\S^c}\|_\infty < 1$. For ease of notation, let $\eta = \max\{n,p\}$. Suppose that Assumptions 4.2, 4.3, 4.4, and 4.5 are satisfied.

Lemma B.1. *Suppose that* $\lambda_n \geq \frac{n^{\kappa_2}\log(\eta)}{n^a}$. *Then, for any* $a \in [0,1/2)$ *there exists a positive constant* C_0 *such that*

$$P(\frac{\|W^n\|_\infty}{\lambda_n} \leq \frac{\alpha}{4(2-\alpha)}) \geq 1-2d\cdot exp(-C_0\cdot n^{1-2a}) - M.\eta^{-2}. \tag{B.7}$$

Lemma B.2. *Suppose that* $\|W^n\|_\infty \leq \frac{\lambda_n}{4}$. *For* $\lambda_n \leq \frac{\lambda^2_{\min}}{30n^{\kappa_2}\log(\eta)d\lambda_{\max}}$,

$$P\left(\|\widetilde{\theta}_S - \theta^*_S\|_2 \leq \frac{5}{\lambda_{\min}}\sqrt{d}\lambda_n\right) \geq 1-2M\cdot\eta^{-2}. \tag{B.8}$$

Lemma B.3. *Suppose that* $\|W^n\|_\infty \leq \frac{\lambda_n}{4}$. *For* $\lambda_n \leq \frac{\alpha}{300(2-\alpha)}\frac{\lambda^2_{\min}}{n^{\kappa_2}\log(\eta)d\lambda_{\max}}$,

$$P\left(\|R^n\|_\infty \leq \frac{\alpha\lambda_n}{4(2-\alpha)}\right) \geq 1-2M\cdot\eta^{-2}. \tag{B.9}$$

The rest of the proof is straightforward from Lemmas B.1, B.2, and B.3. Consider the choice of regularization parameter $\lambda_n = \frac{n^{\kappa_2}\log(\eta)}{n^a}$ for some constants $a \in (2\kappa_2, 1/2)$ where κ_2 is distribution depending constant in Assumption 4.5. Then, the condition for Lemma B.1 is satisfied, and therefore $\|W_n\|_\infty \leq \frac{\lambda_n}{4}$. Moreover, the conditions for Lemmas B.2 and B.3 are satisfied for a sufficiently large sample size $n \geq C'(d\log(\eta)^2)^{\frac{1}{a-2\kappa_2}}$ for some positive constants C'. Therefore, there exist some positive constants C_1, C_2 and C_3 such that

$$\|\widetilde{z}_{\S^c}\|_\infty \leq (1-\alpha)+(2-\alpha)\left[\frac{\|W^n\|_\infty}{\lambda_n}+\frac{\|R^n\|_\infty}{\lambda_n}\right] \leq (1-\alpha)+\frac{\alpha}{4}+\frac{\alpha}{4} < 1, \tag{B.10}$$

with probability of at least $1 - C_1 d\exp(-C_2 n^{1-2a}) - C_3\eta^{-2}$.

For the sign recovery, it is sufficient to show that $\|\widehat{\theta}_M - \theta^*_M\|_\infty \leq \frac{\|\theta^*_M\|_{\min}}{2}$. By Lemma B.2, we have $\|\widehat{\theta}_M - \theta^*_M\|_\infty \leq \|\widehat{\theta}_M - \theta^*_M\|_2 \leq \frac{5}{\lambda_{\min}}\sqrt{d}\ \lambda_n \leq \frac{\|\theta^*_M\|_{\min}}{2}$ as long as $\|\theta^*_M\|_{\min} \geq \frac{10}{\lambda_{\min}}\sqrt{d}\ \lambda_n$.

Furthermore the assumption $\|\theta^*_M\|_{\min} \geq 0$ guarantees that surrogate GLMLasso recovers the true neighborhood of each node with high probability since the solution of GLMLasso is sufficiently close to the solution of GLM.

Furthermore, since we have p regression problems if a sample size $n \geq C'(d\log(\eta)^2)^{\frac{1}{a-2\kappa_2}})$, the moralized graph can be recovered with high probability:

$$P(\widehat{G^m} = G^m) \geq 1 - C_1 d \cdot p \cdot \exp(-C_2 n^{1-2a}) - C_3\eta^{-1}. \tag{B.11}$$

□

B.1.3.1 Proposition B.1

Here we provide a proposition for the proof for Lemmas B.1, B.2 and B.3.

Proposition B.1. *Suppose that X is a random vector with a distribution $\mathbb{P}$ according to a given DAG G. Let*

$$\xi_1 \ := \ \{\max_{i\in\{1,\cdots,n\}} |X_j^{(i)}| < 3\log(\eta)\}.$$

Then, the following statement holds.

$$P(\xi_1^c) \leq M \cdot \eta^{-2}. \tag{B.12}$$

Proof. We now show the $P(\xi_1^c)$ is bounded. Applying the union bound and the Chernoff bound for any $i \in \{1, 2, \cdots, n\}$ and $j \in V$,

$$P(\xi_1^c) \leq n. \max_{i\in\{1,\cdots,n\}} P\left(|X_j^{(i)}| > 3\log(\eta)\right) \leq n \cdot \max_i \eta^{-3}\mathbb{E}[\exp(|X_j^{(i)}|)].$$

We obtain $\max_i \mathbb{E}(\exp(|X_j|^{(i)})) < M$ by Assumption 4.4 and hence Therefore we compete the proof. □

B.1.3.2 Proposition B.2

Here we provide a proposition for the proof for Lemma B.2.

Proposition B.2. *Suppose that X is a random vector with a distribution $\mathbb{P}$ according to a given DAG G and M is a positive concentration bound constant in Assumption 4.4. Then, for any vector $u \in \mathbb{R}^p$ such that $\|u\|_1 \leq c'$, and for any positive constant δ, the following statement holds.*

$$P(|\langle u, X\rangle)| \geq \delta \log \eta) \leq M \cdot p \cdot \eta^{-\delta/c'}. \tag{B.13}$$

Proof. We exploit the fact that $\langle u, X\rangle \leq \|u\|_1 \max_{j\in V} |X_j|$. Therefore, we have

$$P(|\langle u, X\rangle)| \geq \delta \log \eta) \leq P(\max_{j\in V} |X_j| \geq \frac{\delta}{\|u\|_1} \log \eta).$$

Using the union bound, we have

$$P(\max_{j\in V} |X_j| \geq \frac{\delta}{\|u\|_1} \log \eta) \leq p \cdot \max_{j\in V} P(|X_j| \geq \frac{\delta}{\|u\|_1} \log \eta).$$

Applying the Chernoff bounding technique and we obtain

$$P(\max_{j\in V} |X_j| \geq \frac{\delta}{\|u\|_1} \log \eta) \leq M \cdot \eta^{-\frac{\delta}{\|u\|_1}}.$$

Therefore we compete the proof. □

B.1.3.3 Proof for Lemma B.1

Proof. Recall that each entry of the sample score function W^n in (B.5) has the additive form $W_t^n = \frac{1}{n}\sum_{i=1}^n W_t^{(i)}$ for any $t \in \S$. In addition, $W_t^n = 0$ for all $t \notin \S$ since $[\theta_M^*]_t = 0$ by the construction of $\theta_M^* \in \Theta_M$ in (4.7). For any $i \in \{1, 2, \cdots, n\}$ and $t \in S$, it is straightforward to see that the variables

$$W_t^{(i)} = X_t^{(i)} X_j^{(i)} - D'(\langle \theta_\S^*, X_\S^{(i)}\rangle) X_t^{(i)}$$

are independent and have zero expectations.

Now, we show that for all $i \in \{1, 2, \cdots, n\}$, $|W_t^{(i)}|$ is bounded with high probability given the following event ξ_1 so as to use the Hoeffding's inequality. The event ξ_1 is as follows:

$$\xi_1 \quad := \quad \{\max_{i\in\{1,\cdots,n\}} |X_j^{(i)}| < 3\log(\eta)\}.$$

Clearly given ξ_1, $\theta^*_{\S}, X_{\S}^{(i)} < 3\|\theta^*_{\S}\|_1 \log(\eta)$, and therefore $\max_{i\in\{1,2,\cdots,n\}} |D'(\langle\theta^*_{\S}, X_{\S}^{(i)}\rangle)| \leq n^{\kappa_2}$ by Assumption 4.5. Furthermore given ξ_1, $X_t^{(i)} X_j^{(i)} < 9\log(\eta)^2$. Therefore there exists a positive constant C_0 such that $\max_{i\ \in\{1,2,\cdots,n\}} |W_t^{(i)}| \leq C_0 n^{\kappa_2} \log(\eta)$.

Recall that d is the maximum degree of the moralized graph and hence $|\S| \leq d$. Applying the union bound, we have

$$P(\|W^n\|_\infty > \delta, \xi_1) \leq d \cdot \max_{t\in\S} P(|W_t^n| > \delta, \xi_1).$$

Since $|W_t^{(i)}| \leq C_0 n^{\kappa_2} \log(\eta)$ given ξ_1, using the Hoeffding's inequality we obtain

$$P(\|W^n\|_\infty > \delta, \xi_1) \leq 2d \cdot \exp(-\frac{n\delta^2}{2(C_0 n^{\kappa_2}\log(\eta))^2}).$$

Suppose that $\delta = \frac{\lambda_n \alpha}{4(2-\alpha)}$ and $\lambda_n \geq \frac{n^{\kappa_2}\log(\eta)}{n^a}$ for some $a \in [0, 1/2)$. We then have the following result.

$$\begin{aligned} P(\frac{\|W^n\|_\infty}{\lambda_n} > \frac{\alpha}{4(2-\alpha)}, \xi_1) &\leq 2d \cdot \exp\big(-\frac{n\lambda_n^2\alpha^2}{32 \cdot C_0^2(2-\alpha)^2(n^{\kappa_2}\log(\eta))^2}\big) \\ &\leq 2d \cdot \exp\big(-\frac{n^{1-2a}\alpha^2}{32 \cdot C_0^2(2-\alpha)^2}\big). \end{aligned} \tag{B.14}$$

Note that $P(A) = P(A \cap B) + P(A \cap B^c) \leq P(A \cap B) + P(B^c)$ for any sets A and B. Then,

$$P(\frac{\|W^n\|_\infty}{\lambda_n} > \frac{\alpha}{4(2-\alpha)}) \quad \leq \quad 2P(\|W^n\|_\infty > \delta, \xi_1) + P(\xi_1^c)$$

we obtain $P(\xi_1^c) \leq M.\eta^{-2}$ by Proposition B.1. Then, we complete the proof.

$$P(\frac{\|W^n\|_\infty}{\lambda_n} > \frac{\alpha}{4(2-\alpha)}) \leq 2d \cdot \exp(-n^{1-2a}\frac{\alpha^2}{32 \cdot C_0^2(2-\alpha)^2}) + M.\eta^{-2}.$$

□

B.1.3.4 Proof for Lemma B.2

Proof. In order to establish the error bound $\|\widetilde{\theta}_{\S} - \theta^*_{\S}\| \leq B$ for some radius B, several works [77, 57, 56] proved that it suffices to show $F(u_{\S}) > 0$ for all $u_{\S} := \widetilde{\theta}_{\S} - \theta^*_{\S}$ such that $\|u_{\S}\|_2 = B$ for some radius $B > 0$ where

$$F(a) := \ell(\theta^*_{\S} + a; x) - \ell(\theta^*_{\S}; x) + \lambda_n(\|\theta^*_{\S} + a\|_1 - \|\theta^*_{\S}\|_1). \tag{B.15}$$

Since $u_{\S} = \widetilde{\theta}_{\S} - \theta^*_{\S}$ is the minimizer of F and $F(0) = 0$, by the construction of (B.15), $F(u_{\S}) \leq 0$. Note that F is convex, and therefore we must have $F(u_{\S}) < 0$. We then claim that $\|u_{\S}\|_2 \leq B$. In fact, if $u_{\S}$ lay outside the ball of radius B, then the convex combination $v \cdot \widetilde{u}_{\S} + (1 - v) \cdot 0$ would lie on the boundary of the ball, for an appropriately chosen $v \in (0, 1)$. By convexity,

$$F(v \cdot u_{\S} + (1 - v) \cdot 0) \leq v \cdot F(u_{\S}) + (1 - v) \cdot 0 \leq 0 \tag{B.16}$$

contradicting the assumed strict positivity of F on the boundary.

It thus suffices to establish strict positivity of F on the boundary of the ball with radius $B = M_1 \lambda_n \sqrt{d}$ where $M_1 > 0$ is a parameter to be chosen later in the proof. Let $u_{\S} \in \mathbb{R}^{|\S|}$ be an arbitrary vector with $\|u_{\S}\|_2 = B$. Note that $|\S| \leq d$ since d is the maximum degree of the moralized graph. By the Taylor series expansion of F (B.15), we have

$$F(u_{\S}) = (W^n_{\S})^T u_S + u^T_{\S}[\nabla^2 \ell(\theta^*_M + v u_{\S}; x)] u_S + \lambda_n(\|\theta^*_{\S} + u_{\S}\|_1 - \|\theta^*_{\S}\|_1), \tag{B.17}$$

for some $v \in [0, 1]$. For the first term of Equation (B.17), we have the bound

$$|(W^n_{\S})^T u_{\S}| \leq \|W^n_{\S}\|_\infty \|u_{\S}\|_1 \leq \|W^n_{\S}\|_\infty \sqrt{d} \|u_{\S}\|_2 \leq (\lambda_n \sqrt{d})^2 \frac{M_1}{4},$$

since $\|W^n_{\S}\|_\infty \leq \frac{\lambda_n}{4}$ by the assumption.

Applying the triangle inequality to the last term of Equation (B.17), we have the bound

$$\lambda_n(\|\theta^*_{\S} + u_{\S}\|_1 - \|\theta^*_{\S}\|_1) \geq -\lambda_n \|u_{\S}\|_1 \geq -\lambda_n \sqrt{d} \|u_{\S}\|_2 = -M_1 (\lambda_n \sqrt{d})^2.$$

The second term also has the bound from the Taylor series expansion of the Hessian.

$$
\begin{aligned}
q^* &:= \lambda_{\min}\left(\bigtriangledown^2 \ell(\theta_{\S}^* + v u_{\S})\right) \\
&\geq \min_{v\in[0,1]} \lambda_{\min}\left(\bigtriangledown^2 \ell(\theta_{\S}^* + v u_{\S})\right) \\
&\geq \lambda_{\min}\left(\bigtriangledown^2 \ell(\theta_{\S}^*)\right) - \max_{v\in[0,1]} \|\frac{1}{n}\sum_{i=1}^{n} D'''(\langle \theta_{\S}^* + v u_{\S}, X_{\S}\rangle) u_{\S}^T X_{\S}^{(i)} X_{\S}^{(i)} (X_{\S}^{(i)})^T\|_2 \\
&\geq \lambda_{\min} - \max_{v\in[0,1]} \max_{y:\|y\|_2=1} \frac{1}{n}\sum_{i=1}^{n} |D'''(\langle \theta_{\S}^* + v u_{\S}, X_{\S}\rangle)| |u_{\S}^T X_{\S}^{(i)}| (y^T X_{\S}^{(i)})^2 \qquad \text{(B.18)}
\end{aligned}
$$

We set a new event in order to control the first term $D'''(\langle \theta_{\S}^* + v u_{\S}, X_{\S}\rangle)$;

$$
\xi_2 := \{ \max_{i\in\{1,\cdots,n\}} \langle \theta_{\S}^* + v u_{\S}, X_{\S}^{(i)}\rangle < \kappa_1 \log \eta \}.
$$

Provided ξ_2, Assumption 4.5 yields that

$$
D'''(\langle \theta_{\S}^* + v u_{\S}, X_{\S}\rangle) \leq n^{\kappa_2}. \qquad \text{(B.19)}
$$

In addition, we show the bound of the second term of (B.18). Recall that $\|X_{\S}^{(i)}\|_\infty \leq 3\log(\eta)$ for all $i \in \{1, 2, \cdots, n\}$ given ξ_1. Since $\|u_{\S}\|_1 \leq \sqrt{d}\|u_{\S}\|_2$ and $\|u_{\S}\|_2 = M_1 \lambda_n \sqrt{d}$, we obtain

$$
|u_{\S}^T X_{\S}^{(i)}| \leq 3\log(\eta)\sqrt{d}\|u_{\S}\|_2 \leq 3\log(\eta)\cdot M_1 \lambda_n d. \qquad \text{(B.20)}
$$

Lastly, it is clear that $\max_{y:\|y\|_2=1} (y^T X_{\S}^{(i)})^2 \leq \lambda_{\max}$ by the definition of the maximum eigenvalue and Assumption 4.2. Together with the above two bounds of (B.19) and (B.20), for given ξ_1 and ξ_2 we have

$$
q^* \leq \lambda_{\min} - 3n^{\kappa_2}\log(\eta)\cdot M_1 \lambda_n d\ \lambda_{\max}.
$$

For $\lambda_n \leq \frac{\lambda_{\min}}{6n^{\kappa_2}\log(\eta) M_1 d \lambda_{\max}}$, we have $q^* \leq \frac{\lambda_{\min}}{2}$. Therefore,

$$
F(u) \geq (\lambda_n\sqrt{n})^2 \Big\{ -\frac{1}{4}M_1 + \frac{\lambda_{\min}}{2} M_1^2 - M_1 \Big\},
$$

which is strictly positive for $M_1 = \frac{5}{\lambda_{\min}}$. Therefore for $\lambda_n \leq \frac{\lambda_{\min}^2}{30n^{\kappa_2}\log(\eta)d\lambda_{\max}}$,

$$\|\widetilde{\theta}_S - \theta_S^*\|_2 \leq \frac{5}{\lambda_{\min}}\sqrt{d}\lambda_n$$

with the high probability of at least $1 - P(\xi_1^c) - P(\xi_2^c)$.

Here we show the probability bound of ξ_2^c.

$$\begin{aligned} P(\xi_2^c) &\overset{(a)}{\leq} n \max_{i\in\{1,2,\cdots,n\}} P(\langle \theta_M^* + vu_{\S}, X_{\S}^{(i)}\rangle > \kappa_1 \log\eta) \\ &\overset{(b)}{\leq} n \cdot M \cdot \eta^{-\frac{\kappa_1}{2\|\theta_M^*\|_1}} \\ &\overset{(c)}{\leq} M \cdot \eta^{-2}. \end{aligned}$$

(a) follows from the union bound and (b) follows from Proposition B.2 and the given setting $B \leq \|\theta_M^*\|_1$ because $\min_{j\in V}\min_{t\in\mathcal{N}(j)}|[\theta_M^*]_t| \geq \frac{10}{\lambda_{\min}}\sqrt{d}\lambda_n$. Lastly (c) is from Assumption 4.5 that $\kappa_1 \geq 8\|\theta_M^*\|_1$.

In addition the probability bound of ξ_1^c is provided in Proposition B.1. Therefore we prove that

$$P\left(\|\widetilde{\theta}_S - \theta_S^*\|_2 \leq \frac{5}{\lambda_{\min}}\sqrt{d}\ \lambda_n\right) \geq 1 - 2M\cdot\eta^{-2}.$$

□

B.1.3.5 Proof for Lemma B.3

Proof. In this section, we show the bound of R^n in (B.6). According to the definition, R_t^n for a fixed $t \in \S$ can be written as

$$\begin{aligned} R_t^n &= \frac{1}{n}\sum_{i=1}^n [\nabla^2\ell(\theta_M^*; x) - \nabla^2\ell(\bar{\theta}^{(t)}; x)]_t^T(\widetilde{\theta} - \theta_M^*) \\ &= \frac{1}{n}\sum_{i=1}^n [D''(\langle\theta_M^*, X_{\S}^{(i)}\rangle) - D''(\langle\bar{\theta}_{\S}, X_{\S}^{(i)}\rangle)][X_{\S}^{(i)}(X_{\S}^{(i)})^T]_t^T(\widetilde{\theta} - \theta_M^*) \end{aligned}$$

for $\bar{\theta}^{(t)}$ which is a point in the line between $\widetilde{\theta}_M$ and θ_M^*, i.e, $\bar{\theta}^{(t)} = v\cdot\widetilde{\theta}_M + (1-v)\cdot\theta_M^*$ for some $v \in [0,1]$.

By the mean value theorem, we have

$$R_t^n = \frac{1}{n}\sum_{i=1}^{n}\left\{D'''(\langle\bar{\bar{\theta}}^{(t)}, X_{\S}^{(i)}\rangle)X_t^{(i)}\right\}\left\{v(\widetilde{\theta}_M - \theta_M^*)^T X_{\S}^{(i)}(X_{\S}^{(i)})^T(\widetilde{\theta}_M - \theta_M^*)\right\}$$

for $\bar{\bar{\theta}}^{(t)}$ which is a point in the line between $\bar{\theta}^{(t)}$ and θ_M^*.

We have $|X_j^{(i)}| \leq 3\log(\eta)$ for all $i \in \{1, 2, \cdots, n\}$ given ξ_1 by Proposition B.1. Furthermore we showed that $D'''(\langle\bar{\bar{\theta}}, X_{\S}\rangle) \leq n^{\kappa_2}$ given ξ_2 in Section B.1.3.4. Therefore, given ξ_1 and ξ_2 the following result is straightforward.

$$|R_t^n| \leq 3n^{\kappa_2}\log(\eta)\lambda_{\max}\|\widetilde{\theta} - \theta_M^*\|_2^2.$$

In addition, Lemma B.2 represents that $\|\widetilde{\theta} - \theta_M^*\|_2 \leq \frac{5}{\lambda_{\min}}\sqrt{d}\lambda_n$ for $\lambda_n \leq \frac{\alpha}{10(2-\alpha)}\frac{\lambda_{\min}^2}{30n^{\kappa_2}\log(\eta)d\ \lambda_{\max}}$ provided ξ_1 and ξ_2. Therefore we have

$$\|R^n\|_\infty \leq \frac{75\ n^{\kappa_2}\log(\eta)\ d\ \lambda_{\max}\ \lambda_n^2}{\lambda_{\min}^2} \leq \frac{\alpha\lambda_n}{4(2-\alpha)}$$

with high probability of at least $1 - P(\xi_1^c) - P(\xi_2^c)$. Putting the probability bound of ξ_1^c and ξ_2^c shown in Proposition B.1 and Section B.1.3.4 together, we prove that

$$P\left(\|R^n\|_\infty \leq \frac{\alpha\lambda_n}{4(2-\alpha)}\right) \geq 1 - 2M\eta^{-2}.$$

□

B.1.4 Proof for Theorem 4.8

Proof. Let $X^{(i)} = (X_1^{(i)}, \cdots, X_p^{(i)})$ for $i \in \{1, 2, \cdots, n\}$ and $x = (X^{(1)}, X^{(2)}, \cdots, X^{(n)})$ be the iid n samples from the given QVF DAG model $(G, \mathbb{P})$ with quadratic variance coefficients (β_0, β_1) in (4.1). In addition, let π^* be the true causal ordering of a DAG G. Without loss of generality, assume that the true causal ordering is $\pi^* = (1, 2, \cdots, p)$. For an arbitrary permutation or causal ordering π, let π_j represent its j^{th} element.

Let $T_j(X_j) = \omega_j X_j$ where $\omega_j = (\beta_0 + \beta_1\mathbb{E}(X_j \mid X_{\mathrm{pa}(j)}))^{-1}$ such that $\mathrm{Var}(T_j(X_j) \mid X_{\mathrm{pa}(j)}) = \mathbb{E}(T_j(X_j) \mid X_{\mathrm{pa}(j)})$. For any node $j \in V$ and $S \subset V \setminus \{j\}$, let $\mu_{j|S}$ and $\sigma_{j|S}^2$ represent $\mathbb{E}(T_j(X_j) \mid X_S)$ and $\mathrm{Var}(T_j(X_j) \mid X_S)$, respectively. Furthermore for some

realizations of $x_S \in X_S$, let $\mu_{j|S}(x_S)$ and $\sigma^2_{j|S}(x_S)$ denote $\mathbb{E}(T_j(X_j) \mid X_S = x_S)$ and $\mathrm{Var}(T_j(X_j) \mid X_S = x_S)$, respectively. We will also use the convenient notation $\widehat{\cdot}$ to denote an estimate based on the data. We use $n(x_S) = \sum_{i=1}^n \mathbf{1}(X_S^{(i)} = x_S)$ to denote a total conditional sample size, and $n_S = \sum_{x_S} n(x_S)\mathbf{1}(n(x_S) \geq c_0.n)$ for an arbitrary $c_0 \in (0,1)$ to denote a truncated conditional sample size.

Let E^m denote the set of undirected edges corresponding to the *moralized* graph (i.e., the directed edges without directions and edges between nodes with common children). Recall the definitions $\mathcal{N}(j) = \{k \in V : (j,k) \text{ or } (k,j) \in E^m\}$ denotes the neighborhood set of a node j in the moralized graph, $K(j) = \{k : k \in \mathcal{N}(j-1) \cap \{j, \cdots, p\}\}$ denotes a candidate set for π_j, and $C_{jk} = \mathcal{N}(k) \cap \{\pi_1, \pi_2, \cdots, \pi_{j-1}\}$ denotes a candidate parents set. We assume that the true set of undirected edges corresponding to the *moralized* graph is provided. Hence, $\widehat{K}(j) = K(j)$ and $\widehat{C}_{jk} = C_{jk}$. for all nodes $j \in V$ and $k \in K(j)$.

The overdispersion score of a node $k \in K(j)$ for the j^{th} component of the causal ordering only considers elements of $\mathcal{X}(\widehat{C}_{jk}) = \{x \in \{X^{(1)}_{\widehat{C}_{jk}}, X^{(2)}_{\widehat{C}_{jk}}, \cdots, X^{(n)}_{\widehat{C}_{jk}}\} : n(x) \geq c_0 \cdot n\}$, so we only count up elements that occur sufficiently frequently.

According to the generalized ODS algorithm, the truncated sample conditional mean and variance of $T_j(X_j)$ given $X_S = y$ for $j \in \{1, 2, \cdots, p\}$ and any subset $S \subset \{1, 2, \cdots, p\} \setminus \{j\}$ are following:

$$\begin{aligned}
\widehat{\mu}_{j|S}(y) &:= \frac{1}{n_S(y)} \sum_{i=1}^n T_j(X_j^{(i)}) \mathbf{1}(X_S^{(i)} = y) \\
\widehat{\sigma}^2_{j|S}(y) &:= \frac{1}{n_S(y) - 1} \sum_{i=1}^n (T_j(X_j^{(i)}) - \widehat{\mu}_{j|S}(y))^2 \mathbf{1}(X_S^{(i)} = y).
\end{aligned}$$

We re-state the overdispersion score of a node $k \in K(j)$ for the j^{th} element of the causal ordering (4.5):

$$\begin{aligned}
\widehat{\mathcal{S}}(1,k) &:= \left[\left(\frac{\widehat{\sigma}_j}{\beta_0 + \beta_1 \widehat{\mu}_j} \right)^2 - \frac{\widehat{\mu}_j}{\beta_0 + \beta_1 \widehat{\mu}_j} \right] \\
\widehat{\mathcal{S}}(j,k) &:= \sum_{y \in \mathcal{X}(\widehat{C}_{jk})} \frac{n(y)}{n_{\widehat{C}_{jk}}} \left[\left(\frac{\widehat{\sigma}_{j|\widehat{C}_{jk}}(y)}{\beta_0 + \beta_1 \widehat{\mu}_{j|\widehat{C}_{jk}}(y)} \right)^2 - \frac{\widehat{\mu}_{j|\widehat{C}_{jk}}(y)}{\beta_0 + \beta_1 \widehat{\mu}_{j|\widehat{C}_{jk}}(y)} \right].
\end{aligned}$$

For notational convenience, let each entry of the overdispersion score $\widehat{\mathcal{S}}(j,k)$ for $y \in \mathcal{X}(\widehat{C}_{jk})$ be

$$\widehat{\mathcal{S}}(j,k)(y) := \left(\frac{\widehat{\sigma}_{j|\widehat{C}_{jk}}(y)}{\beta_0 + \beta_1 \widehat{\mu}_{j|\widehat{C}_{jk}}(y)}\right)^2 - \frac{\widehat{\mu}_{j|\widehat{C}_{jk}}(y)}{\beta_0 + \beta_1 \widehat{\mu}_{j|\widehat{C}_{jk}}(y)}. \tag{B.21}$$

Note that the true overdispersion scores are as follows:

$$\begin{aligned}
\mathcal{S}(1,k)^* &:= \left[\left(\frac{\sigma_j}{\beta_0 + \beta_1 \mu_j}\right)^2 - \frac{\mu_j}{\beta_0 + \beta_1 \mu_j}\right], \\
\mathcal{S}(j,k)^* &:= \sum_{y \in \mathcal{X}(C_{jk})} \frac{n(y)}{n_{C_{jk}}} \left[\left(\frac{\sigma_{j|C_{jk}}(y)}{\beta_0 + \beta_1 \mu_{j|C_{jk}}(y)}\right)^2 - \frac{\mu_{j|C_{jk}}(y)}{\beta_0 + \beta_1 \mu_{j|C_{jk}}(y)}\right], \\
\mathcal{S}(j,k)^*(y) &:= \left(\frac{\sigma_{j|C_{jk}}(y)}{\beta_0 + \beta_1 \mu_{j|C_{jk}}(y)}\right)^2 - \frac{\mu_{j|C_{jk}}(y)}{\beta_0 + \beta_1 \mu_{j|C_{jk}}(y)} \qquad \text{for } y \in \mathcal{X}(\widehat{C}_{jk}).
\end{aligned}$$

For ease of notation we introduce the following assumption followed by Assumptions 4.4 and 4.7.

Assumption B.1. *For all $j \in V$, $K \subset pa(j)$ and $S \subset V \setminus (nd(j) \cup K)$, there exists $m_0 > 0$ such that*

$$Var(T_j(X_j) \mid X_S) - \mathbb{E}(T_j(X_j) \mid X_S) > m_0.$$

We proved in Section B.1.1 that for all $j \in V$, $K \subset \mathrm{pa}(j)$ and all $S \subset V \setminus (\mathrm{nd}(j) \cup K)$, $\mathrm{Var}(X_j \mid X_S) - \mathbb{E}(X_j \mid X_S) = (\beta_0 + \beta_1 \mathbb{E}(X_j \mid X_S))^{-4}(1+\beta_1)\mathrm{Var}(\mathbb{E}(X_j \mid X_{pa(j)}) \mid X_S)$. Therefore, given the setting $\beta_1 > -1$ and Assumptions 4.4 and 4.7 guarantee that the above assumption is satisfied. Assumption B.1 ensures that the each component of the true overdispersion score $\mathcal{S}(j,k)^*(y)$ is bounded away from m_0.

Now we show the probability bound of $\widehat{\pi} \neq \pi^*$ given the true moralized graph using the following two events: for any $j \in V$ and $k \in K(j)$,

$$\begin{aligned}
\xi_3 &:= \{\max_{j,k} |\widehat{\mathcal{S}}(j,k) - \mathcal{S}(j,k)^*| < \frac{m_0}{2}\} \\
\xi_4 &:= \{\max_{j} \max_{i \in \{1,2,\cdots,n\}} |X_j^{(i)}| < 3\log(\eta)\}.
\end{aligned}$$

Then, we have

$$\begin{aligned} P(\widehat{\pi} \neq \pi^*) &\overset{(a)}{\leq} P(\widehat{\pi} \neq \pi^*, \xi_3) + P(\xi_3^c, \xi_4) + P(\xi_4^c) \\ &\overset{(b)}{\leq} P(\widehat{\pi}_1 \neq \pi_1^*, \xi_3) + P(\widehat{\pi}_2 \neq \pi_2^*, \xi_3 \mid \widehat{\pi}_1 = \pi_1^*) + \\ &\quad \cdots + P(\widehat{\pi}_p \neq \pi_p^*, \xi_3 \mid \widehat{\pi}_1 = \pi_1^*, \cdots, \widehat{\pi}_{p-1} = \pi_{p-1}^*) + P(\xi_3^c, \xi_4) + P(\xi_4^c) \end{aligned} \tag{B.22}$$

(a) follows from $P(A) \leq P(A \cap B) + P(B^c)$ for some events A and B and (b) follows from $P(A \cup B) = P(A) + P(B \cap A^c) = P(A) + P(B \mid A^c)P(A^c) \leq P(A) + P(B \mid A^c)$ for some events A and B.

We prove the probability bound of (B.22) by induction that requires p steps to recover the causal ordering of a given DAG. Recall that given the true moralized graph, $\widehat{K}(j) = K(j)$ and $\widehat{C}_{jk} = C_{jk}$ for all nodes $j \in V$ and $k \in K(j)$. For the first step $m = 1$, overdispersion scores of π_1 in (4.4) are used where a set of candidate element of π_1 is $K(1) = \{1, 2, \cdots, p\}$. Then, we have

$$\begin{aligned} P(\widehat{\pi}_1 \neq \pi_1^*, \xi_3) &= P\left(\exists k \in K(1) \setminus \{\pi_1^*\} \text{ such that } \widehat{\mathcal{S}}(1, \pi_1^*) > \widehat{\mathcal{S}}(1, k), \xi_3\right) \\ &\overset{(a)}{\leq} (p-1) \max_{k \in K(1) \setminus \{\pi_1^*\}} P\left(\mathcal{S}(1, \pi_1^*)^* + \frac{m_0}{2} > \mathcal{S}(1, k)^* - \frac{m_0}{2}, \xi_3\right) \\ &\overset{(b)}{=} (p-1) \max_{k \in K(1) \setminus \{\pi_1^*\}} P\left(m_0 > \mathcal{S}(1, k)^*, \xi_3\right) \\ &\overset{(c)}{=} 0. \end{aligned}$$

(a) follows from the union bound and the definition of ξ_3. In addition, (b) follows from that $\mathcal{S}(1, \pi_1^*)^* = 0$ by the definition of the transformation $T_j(\cdot)$, and (c) is from Assumption B.1 that overdispersion scores of incorrect nodes are greater than m_0.

For the $m = j - 1$ step, assume that the first $j - 1$ elements of the estimated causal ordering are correct $(\widehat{\pi}_1, \widehat{\pi}_2, \cdots, \widehat{\pi}_{j-1}) = (\pi_1^*, \cdots, \pi_{j-1}^*)$. Then for the $m = j$ step, we consider the probability of a false recovery of π_j^* given $(\pi_1^*, \cdots, \pi_{j-1}^*)$.

$$
\begin{aligned}
P(\widehat{\pi}_j \neq \pi_j^*, \xi_3 \mid \pi_1^*, \cdots, \pi_{j-1}^*) &= P\left(\exists k \in K(j) \setminus \{\pi_j^*\} \text{ such that } \widehat{\mathcal{S}}(j, \pi_j^*) > \widehat{\mathcal{S}}(j,k), \xi_3\right) \\
&\overset{(a)}{\leq} |K(j)| \max_{k \in K(j)\setminus\{\pi_j^*\}} P\left(\mathcal{S}(j,\pi_j^*)^* + \frac{m_0}{2} > \mathcal{S}(j,k)^* - \frac{m_0}{2}, \xi_3\right) \\
&\overset{(b)}{=} |K(j)| \max_{k \in K(j)\setminus\{\pi_j^*\}} P\left(m_0 > \mathcal{S}(j,k)^*, \xi_3\right) \\
&\overset{(c)}{=} 0.
\end{aligned}
$$

Again (a) follows from the union bound and the definition of ξ_3. In addition, (b) follows from that $\mathcal{S}(j, \pi_j^*)^* = 0$ by the definition of the transformation $T_j(\cdot)$, and (c) is from Assumption B.1 that overdispersion scores of incorrect nodes are greater than m_0.

This completes the following statement by induction: for any $j \in V$,

$$
P(\widehat{\pi}_j \neq \pi_j^*, \xi_3 \mid \widehat{\pi}_1 = \pi_1^*, \cdots, \widehat{\pi}_{j-1} = \pi_{j-1}^*) = 0.
$$

Then, the probability bound (B.22) is reduced to

$$
P(\widehat{\pi} \neq \pi^*) \leq P(\xi_3^c, \xi_4) + P(\xi_4^c).
$$

Now we focus on the upper bound of $P(\xi_3^c, \xi_4)$ and $P(\xi_4^c)$.

$$
\begin{aligned}
P(\xi_4^c) &= P(\max_{i \in \{1,2,\cdots,n\}} \max_{j \in \{1,2,\cdots,p\}} |X_j^{(i)}| > 3\log(\eta)) \\
&\overset{(a)}{\leq} n \cdot p \max_{i \in \{1,2,\cdots,n\}} \max_{j \in \{1,2,\cdots,p\}} P(|X_j^{(i)}| > 3\log(\eta)) \\
&\overset{(b)}{\leq} n \cdot p \cdot \eta^{-3} \max_{i \in \{1,2,\cdots,n\}} \max_{j \in \{1,2,\cdots,p\}} \mathbb{E}[\exp(|X_j^{(i)}|)] \\
&\overset{(c)}{\leq} \eta^{-1} M
\end{aligned}
$$

(a) follows from the union bound and (b) follows from the Chernoff bound. Furthermore (c) is from the Assumption 4.4.

For the upper bound of $P(\xi_3^c, \xi_4)$, we introduce the following lemma.

Lemma B.4. *There exist some positive constants C_1 and C_2 such that*

$$P(\xi_3^c, \xi_4) \leq C_1 p^2 c_0^{-1} exp\left(-C_2 \frac{c_0 \cdot n}{(\log(\eta))^4}\right).$$

where c_0 is a sample cut-off parameter.

Lastly, we represent a condition on a sample cut-off parameter c_0. Intuitively, if c_0 is too small, estimated overdispersion scores may be biased due to the lack of sample. In contrast, if c_0 is too big, all components of a condition set C_{jk} may not have enough samples size ($> c_0 \cdot n$), and therefore there is no overdispersion scores. Hence the following proposition provides a maximum value of c_0 ensuring that overdispersion scores exist in worst case.

Proposition B.3. *Given ξ_4, $c_0 \leq (3\log(\eta))^{-d}$ is sufficiently small that at least one component of a condition set C_{jk} of the overdispersion scores has a large sample size which is greater than $c_0.n$.*

Putting Lemmas B.4 and Proposition B.3 together, we complete the proof. For some positive constants C_1 and C_2

$$P(\widehat{\pi} \neq \pi^*) \leq C_1 p^2 (\log(\eta))^d \exp\left(-C_2 \frac{n}{(\log(\eta))^{4+d}}\right) + \frac{M}{\eta}.$$

□

B.1.4.1 Proof for Lemma B.4

Proof. For ease of notation, let $n_{jk} = n_{C_{jk}}$ and $n_{jk}(y) = n_{C_{jk}}(y)$ for $y \in \mathcal{X}(C_{jk})$. Using the union bound, we have for $j \in V$ and $k \in K(j)$

$$P(\xi_3^c, \xi_4) = P(\max_{j,k} |\widehat{\mathcal{S}}(j,k) - \mathcal{S}(j,k)^*| > \frac{m_0}{2}, \xi_4) \leq p^2 \max_{j,k} P(|\widehat{\mathcal{S}}(j,k) - \mathcal{S}(j,k)^*| > \frac{m_0}{2}, \xi_4).$$

Since overdispersion scores have additive forms, we obtain

$$P(|\widehat{\mathcal{S}}(j,k) - \mathcal{S}(j,k)^*| > \frac{m_0}{2}, \xi_4) \leq P(\sum_{y \in \mathcal{X}(C_{jk})} \frac{n_{jk}(y)}{n_{jk}} |\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(y)| > \frac{m_0}{2}, \xi_4).$$

Applying $P(\sum_i Y_i > \delta) \leq \sum_i P(Y_i > \omega_i \delta)$ for any $\delta \in \mathbb{R}$ and $\omega_i \in \mathbb{R}^+$ such that $\sum_i \omega_i = 1$, we have

$$P(\sum_{y \in \mathcal{X}(C_{jk})} \frac{n_{jk}(y)}{n_{jk}} |\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(y)| > \frac{m_0}{2}, \xi_4)$$
$$\leq \sum_{y \in \mathcal{X}(C_{jk})} P(|\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(y)| > \frac{m_0}{2}, \xi_4).$$

Applying the union bound,

$$\sum_{y \in \mathcal{X}(C_{jk})} P(|\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(x)| > \frac{m_0}{2}, \xi_4)$$
$$\leq |\mathcal{X}(C_{jk})| \max_{y \in \mathcal{X}(C_{jk})} P(|\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(y)| > \frac{m_0}{2}, \xi_4).$$

By the definition of the sample cut-off parameter c_0, $n_{jk}(y) \geq c_0 \cdot n$ for all $y \in \mathcal{X}(C_{jk})$. Furthermore since total truncated sample size is less than original sample size, $c_0 \cdot n \cdot |\mathcal{X}(C_{jk})| \leq n$. Therefore the cardinality of a set C_{jk} is at most c_0^{-1}. It implies that

$$|\mathcal{X}(C_{jk})| \max_{y \in \mathcal{X}(C_{jk})} P(|\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(y)| > \frac{m_0}{2}, \xi_4)$$
$$\leq c_0^{-1} \max_{y \in \mathcal{X}(C_{jk})} P(|\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(y)| > \frac{m_0}{2}, \xi_4).$$

Since an overdispersion score is a difference between a conditional mean and a conditional variance, the remainder of this problem is reduced to the consistency rate of a sample conditional mean and variance. Suppose that $\epsilon := \widehat{\mu}_{k|C_{jk}}(y) - \mu_{k|C_{jk}}(y)$ and $\kappa \cdot \epsilon := \widehat{\sigma}^2_{k|C_{jk}}(y) - \sigma^2_{k|C_{jk}}(y)$ for some $\kappa \in \mathbb{R}$. By the definition of the overdispersion scores in (B.21), we have

$$\{\epsilon : |\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(y)| > \frac{m_0}{2}\}$$
$$\subset \left\{ \epsilon : \left| \left(\frac{\sigma_{j|\widehat{C}_{jk}}(y) + \kappa\epsilon}{\beta_0 + \beta_1 \mu_{j|\widehat{C}_{jk}}(y) + \epsilon} \right)^2 - \frac{\mu_{j|\widehat{C}_{jk}}(y) + \epsilon}{\beta_0 + \beta_1 \mu_{j|\widehat{C}_{jk}}(y) + \epsilon} \right.\right.$$
$$\left.\left. - \left(\frac{\sigma_{j|C_{jk}}(y)}{\beta_0 + \beta_1 \mu_{j|C_{jk}}(y)} \right)^2 - \frac{\mu_{j|C_{jk}}(y)}{\beta_0 + \beta_1 \mu_{j|C_{jk}}(y)} \right| > \frac{m_0}{2} \right\}$$
$$= \{\epsilon : \epsilon \in (\epsilon_1, \epsilon_2) \cup (\epsilon_3, \epsilon_4)\}$$

where $\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4$ are highly depending on some constants $\mu, \sigma^2, \beta_0, \beta_1, m$, and κ. More precisely, let

$$\begin{aligned}
\zeta_1(\mu, \sigma^2, \beta_0, \beta_1, m, \kappa) &= \beta_0^3(1 + \beta_1 m) - \beta_1^4 m\mu^3 + 2\beta_1^2\mu^2\kappa\sigma^2 - 2\beta_1^2\mu\sigma^4 \\
&\quad + \beta_0^2(-2\beta_1\mu - 3\beta_1^2 m\mu + 2\kappa\sigma^2) - \beta_0\beta_1\big\{\beta_1\mu^2 + 3\beta_1^2 m\mu^2 + 2\sigma^2(-2\kappa\mu + \sigma^2)\big\}, \\
\zeta_2(\mu, \sigma^2, \beta_0, \beta_1, m, \kappa) &= (\beta_0 + \beta_1\mu)^2\Big[\beta_0^4(1 + 2\kappa\mu) + 2\beta_1^2(\kappa\mu - \sigma^2)^2(\beta_1^2\mu^2 m + 2\sigma^4) \\
&\quad + 4\beta_0\beta_1(\kappa\mu - \sigma^2)\big\{\beta_1^2\mu m(2\kappa\mu - \sigma^2) + \beta_1\mu\sigma^2 - 2\kappa\sigma^2\big\} \\
&\quad + 2\beta_0^3\big\{ - 2\kappa\sigma^2 + \beta_1(\mu + 4m\kappa^2\mu - 2m\kappa\sigma^2)\big\} \\
&\quad + \beta_0^2\big\{4\kappa^2\sigma^4 + 4\beta_1\sigma^2(-2\kappa\mu + \sigma^2) + \beta_1^2(\mu^2 + 12m\kappa^2\mu^2 - 12m\mu\kappa\sigma^2 + 2m\sigma^4)\big\}\Big], \\
\zeta_3(\mu, \sigma^2, \beta_0, \beta_1, m, \kappa) &= \beta_0^2(-2\kappa^2 + 2\beta_1 + \beta_1^2 m) + 2\beta_0\beta\mu(\beta_1 + \beta_1^2 m - \kappa^2) \\
&\quad + \beta_1^2(\beta_1^2 m\mu^2 + 2\sigma^4 - 2\kappa^2\mu^2).
\end{aligned}$$

With the $\zeta_1, \zeta_2, \zeta_3$, we define

$$\epsilon_1' = \frac{\zeta_1(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, m_0, \kappa) + \sqrt{\zeta_2(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, m_0, \kappa)}}{\zeta_3(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, m_0, \kappa)}$$

$$\epsilon_2' = \frac{-\zeta_1(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, m_0, \kappa) + \sqrt{\zeta_2(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, m_0, \kappa)}}{\zeta_3(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, m_0, \kappa)}$$

$$\epsilon_3' = \frac{\zeta_1(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, -m_0, \kappa) + \sqrt{\zeta_2(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, -m_0, \kappa)}}{\zeta_3(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, -m_0, \kappa)}$$

$$\epsilon_4' = \frac{-\zeta_1(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, -m_0, \kappa) + \sqrt{\zeta_2(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, -m_0, \kappa)}}{\zeta_3(\mu_{j|C_{jk}}(x), \sigma^2_{j|C_{jk}}(x), \beta_0, \beta_1, -m_0, \kappa)}$$

Let ϵ_1 be the minimum value of $(\epsilon_1', \epsilon_2', \epsilon_3', \epsilon_4')$, ϵ_2 be the second smallest value, ϵ_3 be the third smallest value, and ϵ_4 be the largest value. If we set $m_0 = 0$, the solutions of $|\widehat{\mathcal{S}}(j,k)(y) - \mathcal{S}(j,k)^*(y)| > 0$ are $\epsilon \in (a_1, 0) \cup (0, a_2)$ for some constants $a_1 < 0$ and $a_2 > 0$. If we set $m_0 > 0$, $\epsilon \in (a_1, a_2) \cup (a_3, a_4)$ for some constants $a_1, a_2 < 0$ and $a_3, a_4 > 0$.

For ease of notation, we define $\epsilon_{\min} = \min\{|\epsilon_2|, |\epsilon_3|\}$. Then, we obtain

$$\{\epsilon : |\widehat{\mathcal{S}}(j,k)(x) - \mathcal{S}(j,k)^*(x)| > \frac{m_0}{2}\} \subset (-\infty, -\epsilon_{\min}) \cup (\epsilon_{\min}, \infty)$$

Note that samples are independent and$\max_{i\in\{1,2,\cdots,n\}} \max_{j\in V} |X_j^{(i)}|$ are bounded by $3\log(\eta)$ given ξ_4. Furthermore recall that $n_{jk}(x) \geq c_0 \cdot n$. Applying Hoeffding's inequality technique, we obtain

$$P(|\widehat{\mu}_{j|C_{jk}}(y) - \mu_{j|C_{jk}}(y)| > \epsilon_{\min}, \xi_4) \leq 2\exp\left(-\frac{\epsilon^2_{\min} c_0 . n}{18(\log(\eta))^2}\right).$$

Note that a sample variance can be decomposed to the following form:

$$\frac{1}{n-1}\left(\sum_i^n X_i^2 - \frac{1}{n}(\sum_i^n X_i)^2\right) = \frac{1}{n}\sum_i^n X_i^2 - \frac{1}{n(n-1)}\sum_{i\neq j} X_i X_j.$$

Applying Hoeffding's inequality technique to the above decomposed sample variance, we have

$$P(|\widehat{\sigma}^2_{j|C_{jk}}(x) - \sigma^2_{j|C_{jk}}(x)| > |\kappa| \cdot \epsilon_{\min}, \xi_4) \leq 2\exp\left(-\frac{\kappa^2\epsilon^2_{\min} c_0 \cdot n}{32(3\log(\eta))^4}\right) + 2\exp\left(-\frac{\kappa^2\epsilon^2_{\min} c_0 \cdot n}{64(3\log(\eta))^4}\right).$$

Therefore, there exist some constants C_1 and C_2 such that

$$P(\xi_3^c, \xi_4) \leq C_1 p^2 c_0^{-1} \exp\left(-C_2 \frac{c_0 \cdot n}{(\log(\eta))^4}\right).$$

□

B.1.4.2 Proof for Proposition B.3

Proof. Let $|X_S|$ denote the cardinality of a set $\{X_S^{(1)}, X_S^{(2)}, \cdots, X_S^{(n)}\}$ and $|\mathcal{X}(S)|$ denote the cardinality of a truncated set $\mathcal{X}(S) := \{y \in \{X_S^{(1)}, X_S^{(2)}, \cdots, X_S^{(n)}\} : n(y) \geq c_0 \cdot n\}$.

In worst case where $|\mathcal{X}(S)| = 1$, for all $y \in \{X_S^{(1)}, X_S^{(2)}, \cdots, X_S^{(n)}\}$, $n_S(y) = c_0 \cdot n - 1$ except for only one component $z \in \mathcal{X}(S)$ such that $n_S(z) \geq c_0.n$. In this case, the total sample size $n = n_S(z) + (|X_S| - 1)(c_0 \cdot n - 1)$. It yields that

$$n_S(z) = n - (|X_S| - 1)(c_0.n - 1) = n - c_0 \cdot n \cdot |X_S| + c_0 \cdot n + |X_S| - 1.$$

Since $c_0 \cdot n \leq n_S(z)$, we obtain

$$c_0 \leq \frac{n + |X_S| - 1}{n \cdot |X_S|}.$$

Note that $\frac{1}{|X_S|} \leq \frac{n+|X_S|-1}{n \cdot |X_S|}$ and $|X_j^{(i)}| \leq 3\log(\eta)$ for all $j \in V$ and $i \in \{1, 2, \cdots, n\}$ given ξ_4. Then the maximum cardinality of a set X_S is $(3\log(\eta))^{|S|}$. Hence if $c_0 \leq (3\log(\eta))^{-|S|}$ there exists $z \in \mathcal{X}(S)$.

Recall that the size of a candidate parents set C_{jk} is bounded by the maximum degree of the moralized graph d. Therefore if $c_0 \leq 3\log(\eta)^{-d}$, there exists at least one $z \in \mathcal{X}(C_{jk})$.

□

B.1.5 Proof for Theorem 4.9

Proof. The proof for Theorem 4.9 is similar to the proof for Theorem 4.6 in Section B.1.5. Suppose that there are n iid samples $x = \{X^{(1)}, X^{(2)}, \cdots, X^{(n)}\}$ and $X^{(i)} =$

$\{X_1^{(i)}, X_2^{(i)}, \cdots, X_p^{(i)}\}$ for all $i \in \{1, 2, \cdots, n\}$ from a given DAG model $(G, \mathbb{P})$. Without loss of generality, we assume that the true causal ordering is $\pi^* = (1, 2, \cdots, p)$.

For notational simplicity, let $X_{1:j} = \{X_1, X_2, \cdots, X_j\}$ and $\mathcal{T} = \text{pa}(j)$ for a node $j \in V$. Recall that $\langle \cdot, \cdot \rangle$ represents the inner product and $[\cdot]_k$ is an element of a vector corresponding to a variable X_k. Then the negative surrogate conditional log-likelihood of GLM (4.10) for a variable X_j given $X_{1:j-1}$ is as follows.

$$\ell_D(\theta; x) = \frac{1}{n} \sum_{i=1}^{n} \left(-X_j^{(i)} \langle \theta, X_{1:j-1}^{(i)} \rangle + D(\langle \theta, X_{1:j-1}^{(i)} \rangle) \right)$$

where $D(\cdot)$ is the log-normalization constant determined by the choice of GLM and $\theta \in \mathbb{R}^{j-1}$

The main goal of the proof is to find the minimizer of the following convex problem for any node $j \in V$:

$$\widehat{\theta}_D := \arg \min_{\theta \in \mathbb{R}^{j-1}} \mathcal{L}(\theta, \lambda_n) = \arg \min_{\theta \in \mathbb{R}^{j-1}} \{\ell_D(\theta; x) + \lambda_n \|\theta\|_1\} \tag{B.23}$$

Using the *sub-differential* technique, $\widehat{\theta}_D$ must hold the following condition:

$$\bigtriangledown_\theta \mathcal{L}_D(\widehat{\theta}_D, \lambda_n) = \bigtriangledown_\theta \ell_D(\widehat{\theta}_D; x) + \lambda_n \widehat{z} = 0 \tag{B.24}$$

where $\widehat{z} \in \mathbb{R}^{j-1}$ and an element of $\widehat{z}$ corresponding to a parameter $[\widehat{\theta}_D]_t$ is $\widehat{z}_t = \text{sign}([\widehat{\theta}_D]_t)$ if a node $t \in \mathcal{T}$ otherwise $|\widehat{z}_t| < 1$.

Similar to the proof for the Step 1) in Section B.1.3, the main idea of the proof is *primal-dual-witness* method which asserts that there is a dual problem $\widetilde{\theta}_D = \widehat{\theta}_D$ if the following conditions are satisfied.

(a) We determine the vector $\widetilde{\theta}_D \in \Theta$ where $\Theta = \{\theta \in \mathbb{R}^{j-1} : \theta_{\mathcal{T}^c} = 0\}$ by solving the following restricted objective problem.

$$\widetilde{\theta}_D := \arg \min_{\theta \in \Theta} \mathcal{L}_D(\theta, \lambda_n) = \arg \min_{\theta \in \Theta} \{\ell_D(\theta; x) + \lambda_n \|\theta\|_1\}. \tag{B.25}$$

(b) We choose $\widetilde{z}$ as a member of the sub-differential of regularizer $\|.\|_1$ evaluated by $\widetilde{\theta}_D$.

(c) For any $t \in \mathcal{T}$, $\widetilde{z}_t = \text{sign}([\widetilde{\theta}_D]_t)$;

(d) For any $t \notin \mathcal{T}$, $|\widetilde{z}_t| < 1$.

The conditions (a), (b), and (c) suffice to obtain a pair $(\widetilde{\theta}_D, \widetilde{z})$ that satisfy the optimality conditions (B.24), and therefore the remainder of the proof is to show $|\widetilde{z}_t| < 1$ for all $t \notin \mathcal{T}$.

Equation (B.24) with the dual solution $(\widetilde{\theta}_D, \widetilde{z})$ can be represented as $\bigtriangledown^2 \ell_D(\theta_D^*; x)(\widetilde{\theta}_D - \theta_D^*) = -\lambda_n \widetilde{z} - W^n + R^n$ by using mean value theorem where

(a) W^n is the sample score function

$$W^n := - \bigtriangledown \ell_D(\theta_D^*; x) \tag{B.26}$$

(b) $R^n = (R_1^n, R_2^n, \cdots, R_{j-1}^n)$ and R_k^n is the remainder term by applying coordinate-wise mean value theorem

$$R_k^n := [\bigtriangledown^2 \ell_D(\theta_D^*; x) - \bigtriangledown^2 \ell_D(\bar{\theta}^{(k)}; x)]_k^T (\widetilde{\theta}_D^{(k)} - \theta_D^*) \tag{B.27}$$

where $\bar{\theta}^{(j)}$ is a vector on the line between $\widetilde{\theta}_D$ and θ_D^* and $[\cdot]_k^T$ is the k^{th} row of a matrix.

Let $Q = \bigtriangledown^2 \ell_D(\theta_D^*; x)$ be the Hessian of the surrogate negative conditional log-likelihood of a GLM and $Q_{\mathcal{T}\mathcal{T}}$ be a sub-matrix corresponding to variables $X_{\mathcal{T}}$. Since the set $\widetilde{\theta}_{\mathcal{T}^c} = 0$ in our primal-dual construction, we can re-state condition (B.24) in a block form as follows:

$$\begin{aligned} Q_{\mathcal{T}^c\mathcal{T}}[\widetilde{\theta}_{\mathcal{T}} - \theta_{\mathcal{T}}^*] &= W_{\mathcal{T}^c}^n - \lambda_n \widetilde{z}_{\mathcal{T}^c} + R_{\mathcal{T}^c}^n. \\ Q_{\mathcal{T}\mathcal{T}}[\widetilde{\theta}_S - \theta_S^*] &= W_{\mathcal{T}}^n - \lambda_n \widetilde{z}_{\mathcal{T}} + R_{\mathcal{T}}^n. \end{aligned}$$

Since the matrix $Q_{\mathcal{T}\mathcal{T}}$ is invertible, the above equations can be rewritten as

$$Q_{\mathcal{T}^c\mathcal{T}} Q_{\mathcal{T}\mathcal{T}}^{-1} [W_{\mathcal{T}}^n - \lambda_n \widetilde{z}_{\mathcal{T}} - R_{\mathcal{T}}^n] = W_{\mathcal{T}^c}^n - \lambda_n \widetilde{z}_{\mathcal{T}^c} - R_{\mathcal{T}^c}^n,$$

Then, we have

$$[W^n_{\mathcal{T}^c} - R^n_{\mathcal{T}^c}] - Q_{\mathcal{T}^c\mathcal{T}}Q^{-1}_{\mathcal{T}\mathcal{T}}[W^n_{\mathcal{T}} - R^n_{\mathcal{T}}] + \lambda_n Q_{\mathcal{T}^c\mathcal{T}}Q^{-1}_{\mathcal{T}\mathcal{T}}\widetilde{z}_{\mathcal{T}} = \lambda_n \widetilde{z}_{\mathcal{T}^c}.$$

Taking the ℓ_∞ norm of both sides yields

$$\|\widetilde{z}_{\mathcal{T}^c}\|_\infty \;\leq\; |||Q_{\mathcal{T}^c\mathcal{T}}Q^{-1}_{\mathcal{T}\mathcal{T}}|||_\infty \left[\frac{\|W^n_{\mathcal{T}}\|_\infty}{\lambda_n} + \frac{\|R^n_{\mathcal{T}}\|_\infty}{\lambda_n} + 1\right] + \frac{\|W^n_{\mathcal{T}^c}\|_\infty}{\lambda_n} + \frac{\|R^n_{\mathcal{T}^c}\|_\infty}{\lambda_n}.$$

Recalling Assumptions 4.3, we obtain $|||Q_{\mathcal{T}^c\mathcal{T}}Q^{-1}_{\mathcal{T}\mathcal{T}}|||_\infty \leq (1-\alpha)$, so that we have

$$\begin{aligned}\|\widetilde{z}_{\mathcal{T}^c}\|_\infty &\leq (1-\alpha)\left[\frac{\|W^n_{\mathcal{T}}\|_\infty}{\lambda_n} + \frac{\|R^n_{\mathcal{T}}\|_\infty}{\lambda_n} + 1\right] + \frac{\|W^n_{\mathcal{T}^c}\|_\infty}{\lambda_n} + \frac{\|R^n_{\mathcal{T}^c}\|_\infty}{\lambda_n} \\ &\leq (1-\alpha) + (2-\alpha)\left[\frac{\|W^n\|_\infty}{\lambda_n} + \frac{\|R^n\|_\infty}{\lambda_n}\right].\end{aligned}$$

We apply the following Corollaries B.1, B.2, and B.3 to show $\|z_{\S^c}\|_\infty < 1$. These corollaries directly follows from Lemma B.1, B.2, and B.3, respectively because only differences are re-defined $Q_{\mathcal{T}\mathcal{T}}$, W_n in (B.26) and R_n in (B.27). For ease of notation, let $\eta = \max\{n, p\}$. Suppose that Assumptions 4.2, 4.3, 4.4, and 4.5 are satisfied.

Corollary B.1. *Suppose that* $\lambda_n \geq \frac{n^{\kappa_2}\log(\eta)}{n^a}$ *Then, for any* $a \in [0, 1/2)$ *we have*

$$P(\frac{\|W^n\|_\infty}{\lambda_n} \leq \frac{\alpha}{4(2-\alpha)}) \geq 1 - 2d \cdot exp(-n^{1-2a}\frac{\alpha^2}{32(2-\alpha)^2}) + M \cdot \eta^{-2}.$$

Corollary B.2. *Suppose that* $\|W^n\|_\infty \leq \frac{\lambda_n}{4}$. *For* $\lambda_n \leq \frac{\lambda^2_{\min}}{30n^{\kappa_2}\log(\eta)d\lambda_{\max}}$,

$$P\left(\|\widetilde{\theta}_{\mathcal{T}} - \theta^*_S\|_2 \leq \frac{5}{\lambda_{\min}}\sqrt{d}\lambda_n\right) \geq 1 - 2M \cdot \eta^{-2}.$$

Corollary B.3. *Suppose that* $\|W^n\|_\infty \leq \frac{\lambda_n}{4}$. *For* $\lambda_n \leq \frac{\alpha}{300(2-\alpha)}\frac{\lambda^2_{\min}}{n^{\kappa_2}\log(\eta)d\lambda_{\max}}$,

$$P\left(\|R^n\|_\infty \leq \frac{\alpha\lambda_n}{4(2-\alpha)}\right) \geq 1 - 2M \cdot \eta^{-2}.$$

As we discussed in Section B.1.3, we consider the choice of regularization parameter $\lambda_n = \frac{n^{\kappa_2}\log(\eta))^2}{n^a}$ for some constants $a \in (2\kappa_2, 1/2)$. Then, the condition for Corollary B.1 is satisfied ,and hence $\|W_n\|_\infty \leq \frac{\lambda_n}{4}$. Moreover, for a sufficiently large sample size $n \geq D'(d\log(\eta)^2)^{\frac{1}{a-2\kappa_2}}$ for some positive constants D', the conditions for

Corollary B.2 and B.3 are satisfied. Therefore, there exist some positive constants D_1, D_2 and D_3 such that

$$\|\widetilde{z}_{\S^c}\|_\infty \leq (1-\alpha) + (2-\alpha)\left[\frac{\|W^n\|_\infty}{\lambda_n} + \frac{\|R^n\|_\infty}{\lambda_n}\right] \leq (1-\alpha) + \frac{\alpha}{4} + \frac{\alpha}{4} < 1, \quad \text{(B.28)}$$

with probability of at least $1 - D_1 d\exp(-D_2 n^{1-2a}) - D_3\eta^{-2}$.

For the sign recovery, it is sufficient to show that $\|\widehat{\theta}_D - \theta^*_D\|_\infty \leq \frac{\|\theta^*_D\|_{\min}}{2}$. By Corollary B.2, we have $\|\widehat{\theta}_D - \theta^*_D\|_\infty \leq \|\widehat{\theta}_D - \theta^*_D\|_2 \leq \frac{5}{\lambda_{\min}}\sqrt{d}\ \lambda_n \leq \frac{\|\theta^*_D\|_{\min}}{2}$ as long as $\|\theta^*_D\|_{\min} \geq \frac{10}{\lambda_{\min}}\sqrt{d}\ \lambda_n$.

Since the solution of GLMLasso is sufficiently close to the solution of GLM, the assumption $\|\theta^*_D\|_{\min} \geq 0$ guarantees that surrogate GLMLasso recovers the parents of each node with high probability.

Furthermore, since we have p regression problems if a sample size $n \geq D'(d\log(\eta)^2)^{\frac{1}{a-2\kappa_2}}$), the DAG structure can be recovered with high probability:

$$P(\widehat{G} = G) \geq 1 - D_1 d \cdot p \cdot \exp(-D_2 n^{1-2a}) - D_3\eta^{-1}. \quad \text{(B.29)}$$

□

Appendix C

Proofs for Chapter 5

C.1 Appendix

C.2 Proof for Theorem 5.1

Theorem C.1. *Consider a DAG $G = (V, E)$ with the maximum degree of the moralized graph, d. If single-node interventions are performed at every node and n_0 measurements are made per intervention, then Alg. 1 recovers the true DAG wit high probability:*

$$P(\widehat{G} = G) \geq 1 - \sum_{p_0=1}^{p} p_0 \left\{\delta(n_0(p-1), p_0 - 1, d) + \delta(n_0, \min(d, p_0 - 1), d)\right\}, \quad \text{(C.1)}$$

where $\delta(n, p-1, d)$ is an error bound for estimating a moralized graph with sample size n, possible neighborhood size $p-1$, and the maximum degree of moralize graph d.

Proof. Consider a step in Alg. 1 when the number of remainingNodes is p_0. The first step in the while loop is to find the leaf nodes. In order to determine if a node j is a leaf node, the function FindLeaves finds the moralized neighbors $\mathcal{N}(j)$ of j and compares them to the intervened neighbors $\mathcal{N}_I(j)$ of a node j.

To determine the moralized neighbors of a node j, FindLeaves calls FindNeighbors with all the measurements where the node j was not intervened ($size =$

$(p-1)n_0$), and the set remainingNodes as the search set (size $= p_0 - 1$). Hence the probability that we do not find the correct moralized neighbors of a node j is $\delta((p-1)n_0, p_0 - 1, d)$. This is the first term in the error bound.

Given the neighbors of a node j, Alg. 1 finds the intervened neighbors $\mathcal{N}_I(j)$ of a node j by calling FindNeighbors with measurements where node j was intervened (size $= n_0$), and the neighbors $\mathcal{N}(j)$ as the search set. Since the maximum degree of the moralized graph is d by assumption, the maximum size of the search set is $\min(d, p_0 - 1)$. Hence, the error of this step is bounded by $\delta(n, \min(d, p_0 - 1), d)$.

During a single FindLeaves iteration, the above two steps are repeated for each node, for a total of p_0 nodes. Finally, in the worst case, FindLeaves returns only 1 leaf node, and the while loop in Alg. 1 is repeated p times giving us the error bound in Thm. 3.1. ? □

C.3 Proof for Lemma 5.1

Proof. Since both nodes $j, k \in V$ are not intervened, the directed edge between (j, k) cannot be eliminated by an intervention. Therefore $(j, k) \notin E_I^m$ implies that the edge between (j, k) is not a directed edge, but is generated by some common child which was intervened. □

C.4 Proof for Lemma 5.2

Proof. Since no components of I are adjacent in G^m, for any node $j \in I, \mathcal{N}(j) \cap I = \emptyset$. This means if an undirected edge connecting to a node j in G^m is eliminated in G_I^m, it can only be due to an intervention at node j. Recall that an intervention eliminates the edges between each component of I and its parents. Hence it is easy to see that $\mathcal{N}(j) \cap \mathcal{N}_I(j)^c \subset \text{pa}(j)$ for any $j \in I$. □

C.5 Proof for Lemma 5.3

Proof. Since all components of I are not adjacent in G^m, for any node $j \in I, \mathcal{N}(j) \cap I = \emptyset$. It means if an undirected edge connecting to a node j in G_I is eliminated in G_I^m, it is due to an intervention of a node j.

For any $k \in S$, let $\ell = \mathcal{N}(j) \cap \mathcal{N}(k)$. Then for any $l \in \ell$, a triple (j, k, l) consists of a triangle. Note that a triangle in G^m can be generated by not only all directed edges but a V-structure. In the following, we show how to distinguish between child and spouse of an intervention node.

(a) If $\ell = \emptyset$, there is no node l such that a triple (j, k, l) makes a triangle. This means that j and k do not have a common child because if the j and k have a common child, it generates undirected edges between j and k in G^m. Therefore $k \notin \mathrm{sp}(j)$.

An intervention eliminates the edges between each component of I and its parents. Since $k \in \mathcal{N}_I(j)$ and $k \notin \mathrm{sp}(j)$, $k \notin \mathrm{pa}(j)$. Therefore $k \in \mathrm{ch}(j)$.

(b) If every node $l \in \ell$ satisfies that $l \to j$, j and k cannot have a common child because components of ℓ are only possible common child of j and k, and every triple has $(j \leftarrow l - k)$. Therefore $k \notin \mathrm{sp}(j)$.

An intervention eliminates the edges between each component of I and its parents. Since $k \in \mathcal{N}_I(j)$ and $k \notin \mathrm{sp}(j)$, $k \notin \mathrm{pa}(j)$. Therefore $k \in \mathrm{ch}(j)$.

(c) Suppose that there exists $t \in V \setminus \ell$ such that $(t, k) \in E$. Then (j, k, t) is an unshielded triple since both (j, k) and (k, t) are adjacent, and $t \notin \ell$. Therefore, $t \notin \mathrm{sp}(j)$ because otherwise (j, k, t) consists a triangle.

By the assumption $j \in \mathrm{an}(t)$, and therefore $k \notin \mathrm{pa}(j)$ otherwise it generates a cycle. Hence we have either $k \in \mathrm{ch}(j)$ or $k \in \mathrm{sp}(j)$. Suppose for the sake of contradiction that $k \in \mathrm{ch}(j)$. Then $(j \to k \leftarrow t)$ consists a V-structure which is contradictory to $t \notin \mathrm{sp}(j)$. Therefore, $k \in \mathrm{sp}(j)$.

□

Appendix D

Proofs for Chapter 6

D.1 Appendix

D.1.1 Examples for Theorem 6.3 (d)

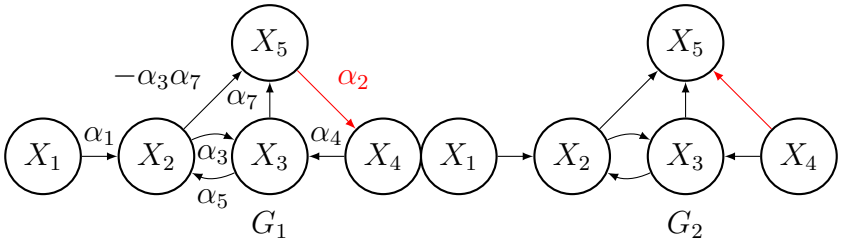


Figure D.1:: 5-node examples for Theorem 6.3 (d)

Suppose that $(G_1, \mathbb{P})$ is a Gaussian linear DCG model with specified edge weights in Figure D.1. With this choice of distribution $\mathbb{P}$ based on G_1 in Figure D.1, we have a set of CI statements which are the same as the set of d-separation rules entailed by G_1 and an additional set of CI statements, $CI(\mathbb{P}) \supset \{X_1 \perp\!\!\!\perp X_4 | \emptyset$, or X_5, $X_1 \perp\!\!\!\perp X_5 | \emptyset$, or $X_4\}$.

It is clear that $(G_2, \mathbb{P})$ satisfies the CMC, $D_{sep}(G_1) \subset D_{sep}(G_2)$ and $D_{sep}(G_1) \neq D_{sep}(G_2)$ (explained in Section 6.3). This implies that $(G_1, \mathbb{P})$ fails to satisfy the P-minimality assumption.

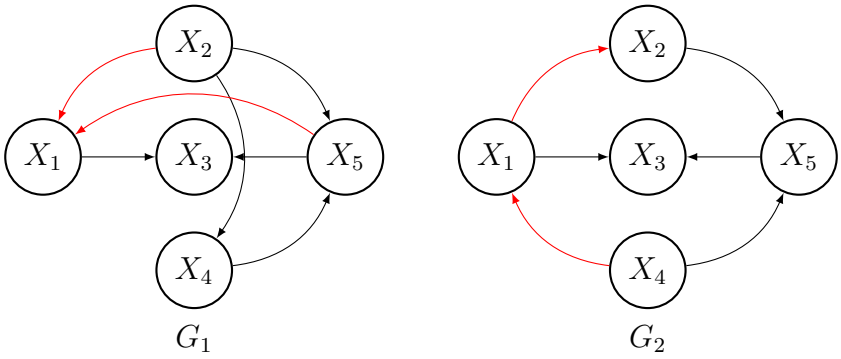


Figure D.2:: 5-node examples for Lemma 6.3.(a)

Now we prove that $(G_1, \mathbb{P})$ satisfies the weak SMR assumption. Suppose that $(G_1, \mathbb{P})$ does not satisfy the weak SMR assumption. Then there exists a G such that $(G, \mathbb{P})$ satisfies the CMC and has fewer edges than G_1. By Lemma 6.2, if $(G, \mathbb{P})$ satisfies the CFC, G satisfies the weak SMR assumption. Note that G_1 does not have edges between (X_1, X_4) and (X_1, X_5). Since the only additional conditional independence statements that are not entailed by G_1 are $\{X_1 \perp\!\!\!\perp X_4 | \emptyset$, or X_5, $X_1 \perp\!\!\!\perp X_5 | \emptyset$, or $X_4\}$, no graph that satisfies the CMC with respect to $\mathbb{P}$ can have fewer edges than G_1. This leads to a contradiction and hence $(G_1, \mathbb{P})$ satisfies the weak SMR assumption.

D.1.2 Proof for Lemma 6.3 (a)

Proof. Here we show that $(G_1, \mathbb{P})$ satisfies the identifiable SMR assumption and and $(G_2, \mathbb{P})$ satisfies the MDR assumption, where $\mathbb{P}$ has the following CI statements:

$$\begin{aligned} CI(\mathbb{P}) = \{&X_2 \perp\!\!\!\perp X_3 \mid (X_1, X_5) \text{ or } (X_1, X_4, X_5); X_2 \perp\!\!\!\perp X_4 \mid X_1; \\ &X_1 \perp\!\!\!\perp X_4 \mid (X_2, X_5) \text{ or } (X_2, X_3, X_5); X_1 \perp\!\!\!\perp X_5 \mid (X_2, X_4); \\ &X_3 \perp\!\!\!\perp X_4 \mid (X_1, X_5), (X_2, X_5), \text{ or } (X_1, X_2, X_5)\}. \end{aligned}$$

Clearly both DAGs G_1 and G_2 do not belong to the same MEC since they have different skeletons. To be explicit, we state all d-separation rules entailed by G_1 and G_2. Both graphs entail the following sets of d-separation rules:

- X_2 is d-separated from X_3 given (X_1, X_5) or (X_1, X_4, X_5).

- X_3 is d-separated from X_4 given (X_1, X_5) or (X_1, X_2, X_5).

The set of d-separation rules entailed by G_1 which are not entailed by G_2 is as follows:

- X_1 is d-separated from X_4 given (X_2, X_5) or (X_2, X_4, X_5).
- X_3 is d-separated from X_4 given (X_2, X_5).

Furthermore, the set of d-separation rules entailed by G_2 which are not entailed by G_1 is as follows:

- X_1 is d-separated from X_5 given (X_2, X_4).
- X_2 is d-separated from X_4 given X_1.

With our choice of distribution, both DAG models $(G_1, \mathbb{P})$ and $(G_2, \mathbb{P})$ satisfy the CMC and it is straightforward to see that G_2 has fewer edges than G_1 while G_1 entails more d-separation rules than G_2.

It can be shown from an exhaustive search that there is no graph G such that G is sparser or as sparse as G_2 and $(G, \mathbb{P})$ satisfies the CMC. Moreover, it can be shown that G_1 entails the maximum d-separation rules amongst graphs satisfying the CMC with respect to the distribution again through an exhaustive search. Therefore $(G_1, \mathbb{P})$ satisfies the MDR assumption and $(G_2, \mathbb{P})$ satisfies the identifiable SMR assumption.

□

D.1.3 Proof for Lemma 6.3 (b)

Proof. Suppose that the pair $(G_2, \mathbb{P})$ is a Gaussian linear DCG model with specified edge weights in Figure D.3, where the non-specified edge weights can be chosen arbitrarily. Once again to be explicit, we state all d-separation rules entailed by G_1 and G_2. Both graphs entail the following sets of d-separation rules:

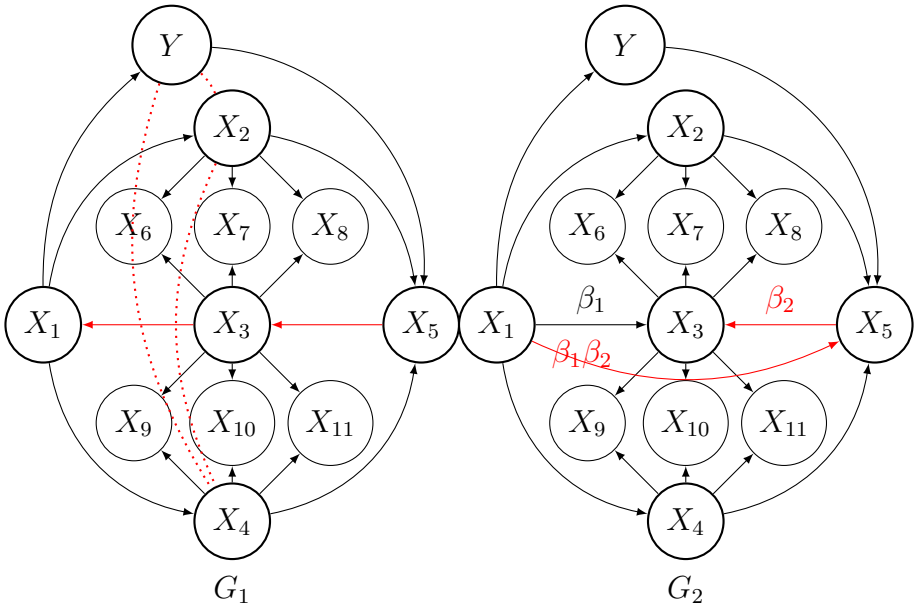


Figure D.3:: 12-node examples for Lemma 6.3.(b)

(1) For any node $A \in \{X_6, X_7, X_8\}$ and $B \in \{X_1, X_5\}$, A is d-separated from B given $\{X_2, X_3\} \cup C$ for any $C \subset \{X_1, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$.

(2) For any node $A \in \{X_9, X_{10}, X_{11}\}$ and $B \in \{X_1, X_5\}$, A is d-separated from B given $\{X_3, X_4\} \cup C$ for any $C \subset \{X_1, X_2, X_3, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$.

(3) For any nodes $A, B \in \{X_6, X_7, X_8\}$, A is d-separated from B given $\{X_2, X_3\} \cup C$ for any $C \subset \{X_1, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$.

(4) For any nodes $A, B \in \{X_9, X_{10}, X_{11}\}$, A is d-separated from B given $\{X_3, X_4\} \cup C$ for any $C \subset \{X_1, X_2, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$.

(5) For any nodes $A \in \{X_6, X_7, X_8\}$ and $B \in \{X_4\}$, A is d-separated from B given $\{X_2, X_3\} \cup C$ for any $C \subset \{X_1, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$, or given $\{X_1, X_2, X_5\} \cup D$ for any $D \subset \{X_4, X_6, X_7, X_8, Y\} \setminus \{A, B\}$.

(6) For any nodes $A \in \{X_6, X_7, X_8\}$ and $B \in \{Y\}$, A is d-separated from B given

$\{X_2, X_3\} \cup C$ for any $C \subset \{X_1, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$, or given $\{X_1, X_2, X_5\} \cup D$ for any $D \subset \{X_4, X_6, X_7, X_8, , X_9, X_{10}\ , X_{11}, Y\} \setminus \{A, B\}$.

(7) For any nodes $A \in \{X_9, X_{10}, X_{11}\}$ and $B \in \{X_2\}$, A is d-separated from B given $\{X_3, X_4\} \cup C$ for any $C \subset \{X_1, X_2, X_5, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$, or given $\{X_1, X_4, X_5\} \cup D$ for any $D \subset \{X_2, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$.

(8) For any nodes $A \in \{X_9, X_{10}, X_{11}\}$ and $B \in \{Y\}$, A is d-separated from B given $\{X_3, X_4\} \cup C$ for any $C \subset \{X_1, X_2, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, Y\} \setminus \{A, B\}$, or given $\{X_1, X_4, X_5\} \cup D$ for any $D \subset \{X_2, X_6, X_7, X_8, X_9, X_{10}\ , X_{11}, Y\} \setminus \{A, B\}$.

(9) For any nodes $A \in \{X_6, X_7, X_8\}$, $B \in \{X_9, X_{10}, X_{11}\}$, A is d-separated from B given $\{X_3\} \cup C \cup D$ for $C \subset \{X_1, X_2, X_4\}$, $C \neq \emptyset$ and $D \subset \{X_1, X_2, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X$ $\{A, B, C\}$.

(10) X_2 is d-separated from X_3 given $\{X_1, X_5\} \cup C$ for any $C \subset \{X_1, X_4, X_5, X_9, X_{10}, X_{11}, Y\}$.

(11) X_3 is d-separated from X_4 given $\{X_1, X_5\} \cup C$ for any $C \subset \{X_1, X_4, X_5, X_6$ $, X_7, X_8, Y\}$.

(12) X_3 is d-separated from Y given $\{X_1, X_5\} \cup C$ for any $C \subset \{X_1, X_4, X_5, X_6$ $, X_7, X_8, X_9, X_{10}, X_{11}\}$.

(13) X_2 is d-separated from X_3 given $\{X_1, X_5\} \cup C$ for any $C \subset \{X_4, X_9\ , X_{10}, X_{11}, Y\}$.

(14) X_4 is d-separated from X_3 given $\{X_1, X_5\} \cup C$ for any $C \subset \{X_2, X_6, X_7\ , X_8, Y\}$.

(15) Y is d-separated from X_3 given $\{X_1, X_5\} \cup C$ for any $C \subset \{X_2, X_6, X_7, X_8$ $, X_4, X_9, X_{10}, X_{11}\}$.

The set of d-separation rules entailed by G_1 that is not entailed by G_2 is as follows:

(a) X_1 is d-separated from X_5 given $\{X_2, X_3, X_4, Y\} \cup C$ for any $C \subset \{X_6, X_7$ $, X_8, X_9, X_{10}, X_{11}\}$.

Furthermore, the set of d-separation rules entailed by G_2 that is not entailed by G_1 is as follows:

(b) X_2 is d-separated from X_4 given X_1 or $\{X_1, Y\}$.

(c) X_2 is d-separated from Y given X_1 or $\{X_1, X_4\}$.

(d) X_4 is d-separated from Y given X_1 or $\{X_1, X_2\}$.

It can then be shown that by using the co-efficients specified for G_2 in Figure D.3, $CI(\mathbb{P})$ is the union of the CI statements implied by the sets of d-separation rules entailed by both G_1 and G_2. Therefore $(G_1, \mathbb{P})$ and $(G_2, \mathbb{P})$ satisfy the CMC. It is straightforward to see that G_2 is sparser than G_1 while G_1 entails more d-separation rules than G_2.

Now we prove that $(G_1, \mathbb{P})$ satisfies the MDR assumption and $(G_2, \mathbb{P})$ satisfies the identifiable SMR assumption. First we prove that $(G_2, \mathbb{P})$ satisfies the identifiable SMR assumption. Suppose that $(G_2, \mathbb{P})$ does not satisfy the identifiable SMR assumption. Then there exists a G such that $(G, \mathbb{P})$ satisfies the CMC and G has the same number of edges as G_2 or fewer edges than G_2. Since the only additional CI statements that are not implied by the d-separation rules of G_2 are $X_1 \perp\!\!\!\perp X_5 \mid \{X_2, X_3, X_4, Y\} \cup C$ for any $C \subset \{X_6, X_7, X_8, X_9, X_{10}, X_{11}\}$ and $(G, \mathbb{P})$ satisfies the CMC, we can consider two graphs, one with an edge between (X_1, X_5) and another without an edge between (X_1, X_5). We firstly consider a graph without an edge between (X_1, X_5). Since G does not have an edge between (X_1, X_5) and by Lemma 6.1, G should entail at least one d-separation rule from (a) X_1 is d-separated from X_5 given $\{X_2, X_3, X_4, Y\} \cup C$ for any $C \subset \{X_6, X_7, X_8, X_9, X_{10}, X_{11}\}$. If G does not have an edge between (X_2, X_3), by Lemma 6.1 G should entail at least one d-separation rule from (10) X_2 is d-separated from X_3 given $\{X_1, X_5\} \cup C$ for any $C \subset \{X_1, X_4, X_5, X_9, X_{10}, X_{11}, Y\}$. These two sets of d-separation rules can exist only if a cycle $X_1 \to X_2 \to X_5 \to X_3 \to X_1$ or $X_1 \leftarrow X_2 \leftarrow X_5 \leftarrow X_3 \leftarrow X_1$ exists. In the same way, if G does not have edges between (X_3, X_4) and (X_3, Y), there should be cycles which are $X_1 \to A \to X_5 \to X_3 \to X_1$

or $X_1 \leftarrow A \leftarrow X_5 \leftarrow X_3 \leftarrow X_1$ for any $A \in \{X_4, Y\}$ as occurs in G_1. However these cycles create virtual edges between $(X_2, X_4), (X_2, Y)$ or (X_4, Y) as occurs in G_1. Therefore G should have at least 3 edges either real or virtual edges. This leads to a contradiction that G has the same number of edges of G_2 or fewer edges than G_2.

Secondly, we consider a graph G with an edge between (X_1, X_5) such that $(G, \mathbb{P})$ satisfies the CMC and G has fewer edges than G_2. Note that G_1 entails the maximum number of d-separation rules amongst graphs with an edge between (X_1, X_5) satisfying the CMC because $CI(\mathbb{P}) \setminus \{X_1 \perp\!\!\!\perp X_5 \mid \{X_2, X_3, X_4, Y\} \cup C$ for any $C \subset \{X_6, X_7, X_8, X_9, X_{10}, X_{11}\}$ is exactly matched to the d-separation rules entailed by G_1. This leads to $D_{sep}(G) \subset D_{sep}(G_1)$ and $D_{sep}(G) \neq D_{sep}(G_1)$. By Lemma 6.2, G cannot contain fewer edges than G_1. However since G_2 has fewer edges than G_1, it is contradictory that G has the same number of edges of G_2 or fewer edges than G_2. Therefore, $(G_2, \mathbb{P})$ satisfies the identifiable SMR assumption.

Now we prove that $(G_1, \mathbb{P})$ satisfies the MDR assumption. Suppose that $(G_1, \mathbb{P})$ fails to satisfy the MDR assumption. Then, there is a graph G such that $(G, \mathbb{P})$ satisfies the CMC and G entails more d-separation rules than G_1 or as many d-separation rules as G_1. Since $(G, \mathbb{P})$ satisfies the CMC, in order for G to entail at least the same number of d-separation rules entailed by G_1, G should entail at least one d-separation rule from (b) X_2 is d-separated from X_4 given X_1 or $\{X_1, Y\}$, (c) X_2 is d-separated from Y given X_1 or $\{X_1, X_4\}$ and (d) X_4 is d-separated from Y given X_1 or $\{X_1, X_2\}$. This implies that G does not have an edge between (X_2, X_4), (X_2, Y) or (X_4, Y) by Lemma 6.1. As we discussed, there is no graph satisfying the CMC without edges (X_2, X_4), (X_2, Y), (X_4, Y), and (X_1, X_5) unless G has additional edges as occurs in G_1. Note that the graph G entails at most six d-separation rules than G_1 (the total number of d-separation rules of (b), (c), and (d)). However, adding any edge in the graph G generates more than six more d-separation rules because by Lemma 6.1, G loses an entire set of d-separation rules from the sets (1) to (15) which each contain more than six d-separation rules. This leads to a contradiction that G entails more

d-separation rules than G_1 or as many d-separation rules as G_1.

□

Bibliography

Joaquín Abellán, Manuel Gómez-Olmedo, Serafín Moral, et al. Some variations on the pc algorithm. In *Probabilistic Graphical Models*, pages 1–8, 2006.

Hirotugu Akaike. A bayesian analysis of the minimum aic procedure. In *Selected Papers of Hirotugu Akaike*, pages 275–280. Springer, 1998.

Constantin F Aliferis, Ioannis Tsamardinos, and Alexander Statnikov. Hiton: a novel markov blanket algorithm for optimal variable selection. In *AMIA Annual Symposium Proceedings*, volume 2003, page 21. American Medical Informatics Association, 2003.

Onureena Banerjee, Laurent El Ghaoui, and Alexandre d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research*, 9:485–516, 2008.

Edward T Bullmore and Danielle S Bassett. Brain graphs: graphical models of the human brain connectome. *Annual review of clinical psychology*, 7:113–140, 2011.

Wray Buntine. A guide to the literature on learning probabilistic networks from data. *Knowledge and Data Engineering, IEEE Transactions on*, 8(2):195–210, 1996.

A Colin Cameron and Pravin K Trivedi. Regression-based tests for overdispersion in the poisson model. *Journal of Econometrics*, 46(3):347–364, 1990.

David Maxwell Chickering. Learning bayesian networks is np-complete. In *Learning from data*, pages 121–130. Springer, 1996.

David Maxwell Chickering. Optimal structure identification with greedy search. *The Journal of Machine Learning Research*, 3:507–554, 2003.

David Maxwell Chickering, David Heckerman, and Christopher Meek. Large-sample learning of bayesian networks is np-hard. *Journal of Machine Learning Research*, 5(Oct):1287–1330, 2004.

David Maxwell Chickering and Christopher Meek. Finding optimal bayesian networks. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 94–102. Morgan Kaufmann Publishers Inc., 2002.

Tom Claassen, Joris Mooij, and Tom Heskes. Learning sparse causal models is not np-hard. *arXiv preprint arXiv:1309.6824*, 2013.

Gregory F Cooper and Edward Herskovits. A bayesian method for the induction of probabilistic networks from data. *Machine learning*, 9(4):309–347, 1992.

Robert G Cowell. *Probabilistic networks and expert systems: Exact computational methods for Bayesian networks*. Springer Science & Business Media, 2006.

Charmaine B Dean. Testing for overdispersion in poisson and binomial regression models. *Journal of the American Statistical Association*, 87(418):451–457, 1992.

Kenji Doya. *Bayesian brain: Probabilistic approaches to neural coding*. MIT press, 2007.

Malcolm Forster, Garvesh Raskutti, Reuben Stern, and Naftali Weinberger. The frugal inference of causal relations. *British Journal for the Philosophy of Science*, 2015.

Jerome H Friedman, Trevor Hastie, and Rob Tibshirani. glmnet: lasso and elastic-net regularized generalized linear models, 2010b. *URL http://CRAN. R-project. org/package= glmnet. R package version*, pages 1–1.

Nir Friedman. Inferring cellular networks using probabilistic graphical models. *Science*, 303(5659):799–805, 2004.

Nir Friedman, Michal Linial, Iftach Nachman, and Dana Pe'er. Using bayesian networks to analyze expression data. *Journal of computational biology*, 7(3-4):601–620, 2000.

Nir Friedman, Iftach Nachman, and Dana Peér. Learning bayesian network structure from massive datasets: the "sparse candidate" algorithm. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 206–215. Morgan Kaufmann Publishers Inc., 1999.

Clark Glymour, Richard Scheines, and Peter Spirtes. *Discovering causal structure: Artificial intelligence, philosophy of science, and statistical modeling*. Academic Press, 2014.

Clark Glymour, Richard Scheines, Peter Spirtes, and Kevin Kelly. Discovering causal structure: Artificial intelligence. *Philosophy of science, and Statistical Modeling*, pages 205–212, 1987.

Alain Hauser and Peter Bühlmann. Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *The Journal of Machine Learning Research*, 13(1):2409–2464, 2012.

Alain Hauser and Peter Bühlmann. Two optimal strategies for active learning of causal models from interventional data. *International Journal of Approximate Reasoning*, 55(4):926–939, 2014.

Alain Hauser and Peter Bühlmann. Jointly interventional and observational data: estimation of interventional markov equivalence classes of directed acyclic graphs. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(1):291–318, 2015.

Yang-Bo He and Zhi Geng. Active learning of causal networks with intervention experiments and optimal designs. *Journal of Machine Learning Research*, 9(11), 2008.

Michael Hecker, Sandro Lambeck, Susanne Toepfer, Eugene Van Someren, and Reinhard Guthke. Gene regulatory network inference: data integration in dynamic modelsâĂŤa review. *Biosystems*, 96(1):86–103, 2009.

David Heckerman, Dan Geiger, and David M Chickering. Learning bayesian networks: The combination of knowledge and statistical data. *Machine learning*, 20(3):197–243, 1995.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. In *The Collected Works of Wassily Hoeffding*, pages 409–426. Springer, 1994.

Antti Hyttinen, Frederick Eberhardt, and Patrik O Hoyer. Causal discovery for linear cyclic models with latent variables. *on Probabilistic Graphical Models*, page 153, 2010.

Antti Hyttinen, Frederick Eberhardt, and Patrik O Hoyer. Causal discovery of linear cyclic models from multiple experimental data sets with overlapping variables. *arXiv preprint arXiv:1210.4879*, 2012.

Antti Hyttinen, Frederick Eberhardt, and Patrik O Hoyer. Learning linear cyclic causal models with latent variables. *The Journal of Machine Learning Research*, 13(1):3387–3439, 2012.

Antti Hyttinen, Frederick Eberhardt, and Patrik O Hoyer. Experiment selection for causal discovery. *The Journal of Machine Learning Research*, 14(1):3041–3071, 2013.

Antti Hyttinen, Patrik O Hoyer, Frederick Eberhardt, and Matti Jarvisalo. Discovering cyclic causal models with latent variables: A general sat-based procedure. *arXiv preprint arXiv:1309.6836*, 2013.

Ali Jalali, Pradeep D Ravikumar, Vishvas Vasuki, and Sujay Sanghavi. On learning discrete graphical models using group-sparse regularization. In *International Conference on Artificial Intelligence and Statistics*, pages 378–387, 2011.

Guy Karlebach and Ron Shamir. Modelling and analysis of gene regulatory networks. *Nature Reviews Molecular Cell Biology*, 9(10):770–780, 2008.

Jeffrey O Kephart and Steve R White. Directed-graph epidemiological models of computer viruses. In *Research in Security and Privacy, 1991. Proceedings., 1991 IEEE Computer Society Symposium on*, pages 343–359. IEEE, 1991.

Steffen L Lauritzen. *Graphical models*. Clarendon Press, 1996.

Po-Ling Loh and Peter Bühlmann. High-dimensional learning of linear causal networks via inverse covariance estimation. *The Journal of Machine Learning Research*, 15(1):3065–3105, 2014.

Christopher Meek. Causal inference and causal explanation with background knowledge. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 403–410. Morgan Kaufmann Publishers Inc., 1995.

Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, pages 1436–1462, 2006.

Joris M Mooij, Dominik Janzing, Tom Heskes, and Bernhard Schölkopf. On causal discovery with cyclic additive noise models. In *Advances in neural information processing systems*, pages 639–647, 2011.

Carl N Morris. Natural exponential families with quadratic variance functions. *The Annals of Statistics*, pages 65–80, 1982.

Preetam Nandy, Marloes H Maathuis, and Thomas S Richardson. Estimating the effect of joint interventions from observational data in sparse high-dimensional settings. *arXiv preprint arXiv:1407.2451*, 2014.

Gunwoong Park and Garvesh Raskutti. Learning large-scale poisson dag models based on overdispersion scoring. In *Advances in Neural Information Processing Systems*, pages 631–639, 2015.

Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.

Judea Pearl. Causality: models, reasoning and inference. *Economet. Theor*, 19:675–685, 2003.

Judea Pearl. Causality: models, reasoning and inference. *Economet. Theor*, 19:675–685, 2003.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 2014.

Judea Pearl, Thomas Verma, et al. *A theory of inferred causation*. Morgan Kaufmann San Mateo, CA, 1991.

Jonas Peters and Peter Bühlmann. Identifiability of gaussian structural equation models with equal error variances. *Biometrika*, page ast043, 2013.

Jonas Peters, Joris Mooij, Dominik Janzing, and Bernhard Schölkopf. Identifiability of causal graphs using functional models. *arXiv preprint arXiv:1202.3757*, 2012.

Garvesh Raskutti and Caroline Uhler. Learning directed acyclic graphs based on sparsest permutations. *arXiv preprint arXiv:1307.0366*, 2013.

Pradeep Ravikumar, Martin J Wainwright, John D Lafferty, et al. High-dimensional ising model selection using ℓ_1-regularized logistic regression. *The Annals of Statistics*, 38(3):1287–1319, 2010.

Pradeep Ravikumar, Martin J Wainwright, Garvesh Raskutti, Bin Yu, et al. High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935–980, 2011.

T Richardson. Properties of cyclic graphical models. *MS ThesisCarnegie Mellon Univ*, 1994.

Thomas Richardson. A discovery algorithm for directed cyclic graphs. In *Proceedings of the Twelfth international conference on Uncertainty in artificial intelligence*, pages 454–461. Morgan Kaufmann Publishers Inc., 1996.

Thomas Richardson. A polynomial-time algorithm for deciding markov equivalence of directed cyclic graphical models. In *Proceedings of the Twelfth international conference on Uncertainty in artificial intelligence*, pages 462–469. Morgan Kaufmann Publishers Inc., 1996.

Gideon Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

Marco Scutari. Learning bayesian networks with the bnlearn r package. *arXiv preprint arXiv:0908.3817*, 2009.

Karthikeyan Shanmugam, Murat Kocaoglu, Alexandros G Dimakis, and Sriram Vishwanath. Learning causal graphs with small interventions. In *Advances in Neural Information Processing Systems*, pages 3177–3185, 2015.

Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, and Antti Kerminen. A linear non-gaussian acyclic model for causal discovery. *The Journal of Machine Learning Research*, 7:2003–2030, 2006.

Craig Silverstein, Sergey Brin, Rajeev Motwani, and Jeff Ullman. Scalable techniques for mining causal structures. *Data Mining and Knowledge Discovery*, 4(2-3):163–192, 2000.

Peter Spirtes. Directed cyclic graphical representations of feedback models. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 491–498. Morgan Kaufmann Publishers Inc., 1995.

Peter Spirtes and Clark Glymour. An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, 9(1):62–72, 1991.

Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2000.

Marc Teyssier and Daphne Koller. Ordering-based search: A simple and effective algorithm for learning bayesian networks. *arXiv preprint arXiv:1207.1429*, 2012.

Ioannis Tsamardinos and Constantin F Aliferis. Towards principled feature selection: Relevancy, filters and wrappers. In *AISTATS*, 2003.

Ioannis Tsamardinos, Laura E Brown, and Constantin F Aliferis. The max-min hill-climbing bayesian network structure learning algorithm. *Machine learning*, 65(1):31–78, 2006.

TS Vermal J udea Pearl. Equivalence and synthesis of causal models. In *Proceedings of Sixth Conference on Uncertainty in Artijicial Intelligence*, pages 220–227, 1991.

Caroline Uhler, Garvesh Raskutti, Peter Bühlmann, Bin Yu, et al. Geometry of the faithfulness assumption in causal inference. *The Annals of Statistics*, 41(2):436–463, 2013.

Sara Van de Geer, Peter Bühlmann, et al. ℓ_0-penalized maximum likelihood for sparse directed acyclic graphs. *The Annals of Statistics*, 41(2):536–567, 2013.

Thomas Verma and Judea Pearl. An algorithm for deciding if a set of observed independencies has a causal explanation. In *Proceedings of the Eighth international conference on uncertainty in artificial intelligence*, pages 323–330. Morgan Kaufmann Publishers Inc., 1992.

Martin J Wainwright, John D Lafferty, and Pradeep K Ravikumar. High-dimensional graphical model selection using ℓ_1-regularized logistic regression. In *Advances in neural information processing systems*, pages 1465–1472, 2006.

Eunho Yang, Genevera Allen, Zhandong Liu, and Pradeep K Ravikumar. Graphical models via generalized linear models. In *Advances in Neural Information Processing Systems*, pages 1358–1366, 2012.

Eunho Yang, Aurelie C Lozano, and Pradeep K Ravikumar. Closed-form estimators for high-dimensional generalized linear models. In *Advances in Neural Information Processing Systems*, pages 586–594, 2015.

Ming Yuan and Yi Lin. Model selection and estimation in the gaussian graphical model. *Biometrika*, 94(1):19–35, 2007.

Jiji Zhang. A comparison of three occam's razors for markovian causal models. *The British Journal for the Philosophy of Science*, page axs005, 2012.

Tian Zheng, Matthew J Salganik, and Andrew Gelman. How many people do you know in prison? using overdispersion in count data to estimate social structure in networks. *Journal of the American Statistical Association*, 101(474):409–423, 2006.