
Differentiable Structure Learning for General Binary Data

Anonymous Author(s)

Affiliation

Address

email

Abstract

Existing methods for differentiable structure learning in discrete data typically assume that the data are generated from specific structural equation models. However, these assumptions may not align with the true data-generating process, which limits the general applicability of such methods. Furthermore, current approaches often ignore the complex dependence structure inherent in discrete data and consider only linear effects. We propose a differentiable structure learning framework that is capable of capturing *arbitrary* dependencies among discrete variables. We show that although general discrete models are unidentifiable from purely observational data, it is possible to characterize the complete set of compatible parameters and structures. Additionally, we establish identifiability up to Markov equivalence under mild assumptions. We formulate the learning problem as a single differentiable optimization task in the most general form, thereby avoiding the unrealistic simplifications adopted by previous methods. Empirical results demonstrate that our approach effectively captures complex relationships in discrete data.

1 Introduction

Causal relationships are often represented by directed acyclic graphs (DAGs), where nodes correspond to variables and directed edges indicate cause-effect links [30, 38, 32]. Learning a DAG from observational data (structure learning, also known as causal discovery) is a well-known NP-complete problem [7, 9]. Recent advances have recast this combinatorial search as a differentiable constrained program, enabling the use of gradient-based methods for structure learning [46, 47]. In this approach, the adjacency matrix of the graph is treated as a continuous variable and one optimizes a score (e.g. negative log-likelihood) subject to a differentiable acyclicity constraint that ensures the solution is a valid DAG [46, 4]. Early work in this area has largely been limited to continuous data and relied on specific structural equation models (SEMs)—for example, linear or nonlinear functions with additive Gaussian noise [33]. While such assumptions facilitate tractable optimization, they may misrepresent the true generative process for non-Gaussian or discrete data, leading to biased or inconsistent structure learning.

Many real-world datasets involve binary or other discrete variables (e.g. presence/absence of a condition, binary genetic markers, survey responses), whose complex dependency structures are not well handled by methods designed for continuous data. Traditional Gaussian or continuous assumptions often break down in these cases. However, only a few works have attempted to extend differentiable structure learning to discrete data. These existing approaches typically impose specific parametric forms on the data-generating process. For instance, a recent method [45] for discrete differentiable structure learning assumes a generalized linear SEM. Moreover, the theoretical identifiability guarantees of these methods rely on particular distributional constraints that may not

36 hold for general discrete data. These gaps call for a more flexible approach that can capture the rich
 37 dependence patterns in discrete data without relying on untested generative assumptions.

38 Outside the differentiable paradigm, decades of work have produced both *score-based* and *constraint-*
 39 *based* algorithms for discrete causal discovery. Score-based searches (e.g., greedy equivalence
 40 search [8]) perform well on small graphs, but their greedy, local exploration of a super-exponential
 41 DAG space often traps them in local optima and limits scalability. Constraint-based procedures—such
 42 as the PC algorithm and its variants [38, 10]—rely on conditional-independence tests that become
 43 unreliable with finite samples or measurement noise. Many discrete methods also impose additional
 44 simplifications (additive-noise models [31], hidden compact representations [6], linear effects [24],
 45 latent Gaussian variables [1]), which can lead to model misspecification. Even the most recent
 46 differentiable approaches evaluated on discrete data lack a formal justification. For instance, Bello
 47 et al. [4] applies continuous optimization methods to discrete data without fully accounting for
 48 discrete-specific characteristics, introducing implicit assumptions and evaluation metrics. These
 49 observations underscore the need for a general, theoretically-sound framework for differentiable
 50 structure learning in discrete data.

51 In this work, we propose a general differentiable structure learning framework for discrete data,
 52 which captures arbitrary dependencies without assuming a specific data-generating process. We
 53 first show that, in this general setting, DAGs are non-identifiable from observational data alone.
 54 Nevertheless, we characterize the complete set of compatible structures and parameters. To select
 55 among these, we adopt a sparsity principle, aiming for the sparsest DAG consistent with the observed
 56 distribution. We formulate this as a single differentiable optimization problem. Lastly, we establish
 57 theoretical guarantees showing that our method recovers the correct Markov equivalence class under
 58 mild assumptions. Unlike prior work, our approach avoids assumptions like linearity or additivity,
 59 enabling it to model richer causal dependencies.

60 In summary, our key contributions are as follows:

- 61 1. We start with general binary data, without assuming a particular data generating process,
 62 and use the *multivariate Bernoulli* distribution (Definition 1) as a general representation
 63 for any such model. We prove that, in this fully general setting, the underlying DAG
 64 is *non-identifiable* and show all the graph-parameter pairs compatible with the observed
 65 distribution (Theorem 1).
- 66 2. We formulate learning of the *sparsest* graph and its parameters as a single differentiable
 67 optimization (13). We establish that any global minimizer yields the sparsest valid DAG,
 68 and that under the *faithfulness* assumption all such DAGs are within in the same Markov
 69 equivalence class (Theorem 3). Moreover, based on our framework, we provide theoretical
 70 justification for prior works [12, 4].
- 71 3. We conduct experiments demonstrating that our method can capture the most general causal
 72 relationships. Furthermore, we introduce a two-stage approach, NOTEARS-MLP-REG,
 73 which reduces computational complexity and outperforms existing baselines.

74 2 Related work

75 In this section, we survey existing approaches to structure learning and causal discovery, highlighting
 76 methods for discrete data, differentiable DAG learning in continuous settings, and recent extensions
 77 of continuous-data techniques to discrete domains. We discuss their key ideas, assumptions, and
 78 limitations, which motivate the need for a general framework for multivariate discrete data.

79 **Traditional structure learning for discrete data** Classical causal-discovery methods for discrete
 80 data fall into two main categories. *Constraint-based* algorithms (e.g. PC [37], MMHC [39], Copula-
 81 PC [10]) use conditional-independence tests to infer edges, but finite samples and measurement
 82 noise often render these tests unreliable. *Score-based* approaches assign each candidate DAG a
 83 goodness-of-fit score (e.g. BIC, BDeu [26, 18], generalized scores [20, 24]) and then search for the
 84 optimal structure. Greedy strategies such as GES [8] can work on small graphs but suffer from NP
 85 hardness [7, 9] and can become trapped in local optima [23, 2]. Both paradigms typically rely on
 86 strong parametric assumptions—additive-noise models [31], hidden compact representations [6],
 87 linear effects [24], latent Gaussian variables [1]—which may not hold in practice. A few methods

target bivariate causal discovery [41, 25, 5], but they do not fit to multivariate problem common in modern applications. In summary, existing discrete causal-discovery techniques either lack statistical robustness in finite samples or impose restrictive assumptions, motivating the need for scalable, assumption-light methods that handle high-dimensional binary data.

Continuous DAG learning for continuous data Since the introduction of a smooth characterization of acyclicity by Zheng et al. [46], which allows treating the adjacency matrix as a continuous optimization variable, a large body of work has emerged in differentiable structure learning. Subsequent research has extended this framework in several directions: toward nonlinear models [47, 44, 22, 48], theoretical analysis from an optimization perspective [40, 29, 13], designing more stable and computationally efficient acyclicity constraints [4, 27, 44], understanding score function properties [28, 15, 36], and incorporating extra side information [3]. Despite this progress, most existing methods are fundamentally designed for *continuous* variables and do not address the specific challenges of discrete data. The common reliance on Gaussianity or continuous-variable assumptions—leading to objectives like mean-squared error or Gaussian log-likelihood—introduces potential biases and inconsistencies when applied to discrete data. Thus, naively transferring continuous-data DAG learning methods to discrete domains is problematic and necessitates principled modifications.

Continuous DAG learning for discrete data Only a handful of recent studies have extended differentiable structure learning to discrete data by imposing specific parametric forms on the conditionals. For example, most of works [45, 4, 12] assume a linear SEM in which each discrete node’s probability is given by a logistic link on a linear combination of its parents. Such restrictive assumptions can fail to capture the rich, higher-order dependencies present in general discrete distributions. Another line of work in differentiable structure learning studies nonlinear models via neural networks [47, 22, 48], which in principle could accommodate discrete inputs. However, these methods treat inputs as generic real-valued vectors and do not formally address the unique features of discrete data, neither in theory nor in empirical evaluation. A recent contribution by Deng et al. [15] considers broad parametric families that include general binary distributions, but it remains unclear how to extend their approach to the discrete variables. Collectively, these limitations motivate the development of a truly general differentiable framework capable of learning arbitrary discrete causal relationships without relying on narrow parametric assumptions.

3 Preliminaries

Let $G = (V, E)$ denote a directed graph on p nodes, with vertex set $V = [p] := \{1, \dots, p\}$ and edge set $E \subset V \times V$, where $(i, j) \in E$ indicates the presence of a directed edge from node i to node j . We associate each node $i \in V$ to a random variable X_i , and let $X = (X_1, \dots, X_p)$. In this work we focus on *discrete* random variables. Any variable with T categories can be represented by $(T - 1)$ binary indicators [42]. For instance, if $Z \in \{0, 1, 2\}$, then introduce $Z^k := \mathbb{1}\{Z = k\}$, $k = 0, 1, 2$, indicating whether $Z = k$ for $k = 0, 1, 2$. so that Z is replaced by three binary indicators. Hence, without loss of generality, we only consider binary data in the sequel. Extending the results to general discrete data only need additional notation and does not require new technical ideas.

We review the *multivariate Bernoulli distribution* (MVB) [11], a flexible family that captures *all* joint dependencies among high-dimensional binary variables. Because it assigns a probability to every possible configuration, it allows us to study binary data without imposing additional assumptions.

Definition 1 (Multivariate Bernoulli distribution [11]). *Let $X = (X_1, \dots, X_p)$ be a vector of possibly correlated Bernoulli random variables with $X_i \in \{0, 1\}$ for $i = 1, \dots, p$. We say that X follows a multivariate Bernoulli distribution with $\mathbf{p} \in \mathbb{R}^{2^p}$, written $X \sim \text{MultiBernoulli}(\mathbf{p})$, if*

$$P(X_1 = x_1, X_2 = x_2, \dots, X_p = x_p) = p(0, 0, \dots, 0) \prod_{j=1}^p (1-x_j) \times p(1, 0, \dots, 0) [x_1 \prod_{j=2}^p (1-x_j)] \\ \times p(0, 1, \dots, 0) [(1-x_1)x_2 \prod_{j=3}^p (1-x_j)] \times \dots \times p(1, 1, \dots, 1) \prod_{j=1}^p x_j, \quad (1)$$

where each entry of $\mathbf{p} = (p(0, 0, \dots, 0), \dots, p(1, \dots, 1)) \in \mathbb{R}^{2^p}$ is the probability mass of a distinct configuration. Such \mathbf{p} is called the general parameter. Let $\mathbf{1}_{2^p} \in \mathbb{R}^{2^p}$ denote the all-ones vector; normalization requires $\mathbf{1}_{2^p}^\top \mathbf{p} = 1$.

135 It is hard to figure out the dependence between each component based on the form of probability
 136 mass function in (1). As a consequence, define the interaction function B where $B^{j_1, \dots, j_r}(x) =$
 137 $x_{j_1} x_{j_2} \dots x_{j_r}$. Rewrite (1) in exponential-family form:

$$P(X = x) = \exp \left(f^0 + \sum_{r=1}^p \left(\sum_{1 \leq j_1 < j_2 < \dots < j_r \leq p} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}(x) \right) \right) \quad (2)$$

138 The natural-parameter vector is $\mathbf{f} = (f^0, f^1, \dots, f^p, \dots, f^{1 \dots p}) \in \mathbb{R}^{2^p}$. Throughout, the superscript
 139 set S in f^S is treated as *unordered*; e.g. for $S = \{1, 2\}$ we have $f^{12} = f^{21}$. Because the MVB is
 140 an exponential family, there is a one-to-one correspondence between the natural parameters \mathbf{f} and
 141 the general parameter \mathbf{p} ; explicit conversion formulas are provided in Appendix C.1. Expressing the
 142 model as in (2) will later allow us to identify variable dependencies directly from the coefficients \mathbf{f}
 143 when reconstructing causal graphs. A simple but useful closure property follows immediately.

144 **Corollary 1.** *If $X \sim \text{MultiBernoulli}(\mathbf{p})$, then every marginal distribution and every conditional*
 145 *distribution of X is again multivariate Bernoulli.*

146 4 General discrete data: a nonidentifiable model

147 We first establish that, for fully general binary data, neither the causal structure nor the associated
 148 parameters can be identified from observational samples alone. The intuition is simple: for any fixed
 149 topological sort (or order) of the variables (Definition 2) there exists a unique causal structure and
 150 associated parameters that reproduces the observed distribution exactly. Because a set of p variables
 151 admits up to $p!$ distinct topological orders, it renders the model fundamentally non-identifiable.

152 4.1 Conditional distribution: higher order interaction

153 **Definition 2** (Topological sort). *Let $G = (V, E)$ be a directed graph with vertex set $V = [p]$. A*
 154 *topological sort is a permutation π of V such that $X_{\pi(i)} \rightarrow X_{\pi(j)} \implies i < j$. Equivalently, the*
 155 *permutation orders the nodes so that every directed edge points from an earlier to a later position. A*
 156 *directed graph is acyclic iff at least one such ordering exists; the ordering need not be unique.*

157 Given a permutation π , the joint distribution admits the standard factorization

$$P(X) = \prod_{j=1}^p P(X_{\pi(j)} \mid X_{\pi(1), \dots, \pi(j-1)})$$

158 Let us use the joint law $P(X)$ given by the exponential form in (2). We examine the conditional
 159 distribution $P(X_{\pi(j)} \mid X_{\pi(1)}, \dots, X_{\pi(j-1)})$. Without loss of generality let $\pi = (1, \dots, p)$ and focus
 160 on $j = p$; other choices of π and j lead to the same algebra with heavier notation. Then

$$\begin{aligned} & P(X_p = 1 \mid X_1 = x_1, \dots, X_{p-1} = x_{p-1}) \\ &= \text{logistic} \left(\sum_{r=1}^p \left(\sum_{1 \leq j_1 < j_2 < \dots < j_r = p \leq p} f^{j_1 \dots j_{r-1} p} x_{j_1} \dots x_{j_{r-1}} x_p \right) \right) \\ &= \text{logistic} (f^p + f^{1p} x_1 + f^{2p} x_2 \dots f^{p-1, p} x_{p-1} + f^{12p} x_1 x_2 + \dots + f^{1 \dots p} x_1 \dots x_{p-1}) \end{aligned} \quad (3)$$

161 where $\text{logistic}(z) = 1/(1 + e^{-z})$. The full derivation appears in Appendix C.2. Equation (3)
 162 highlights that, for general binary data, all higher order interaction terms are present [11, 16].
 163 Omitting these higher-order interactions—e.g., restricting attention to first-order (additive) effects
 164 [43, 45]—can misrepresent real data.

165 4.2 Nonidentifiability and Equivalence

166 In the example above, X_p is the last node in the ordering π , so any variable in (X_1, \dots, X_{p-1}) may
 167 serve as a potential parent of X_p . A variable X_j is deemed a parent precisely when some interaction
 168 coefficient involving j and p is non-zero; equivalently,

$$X_j \rightarrow X_p \Leftrightarrow \text{There exists a set } S \subseteq [p] \setminus \{j, p\}, \text{ such that } f^{j, p, S} \neq 0 \Leftrightarrow \sum_{S \subseteq [p] \setminus \{j, p\}} (f^{j, p, S})^2 > 0 \quad (4)$$

169 All coefficients in (3) can be estimated *simultaneously* via a single logistic regression [21]; deciding
 170 whether an edge $X_j \rightarrow X_p$ exists thus reduces to checking if the corresponding coefficient set is
 171 non-zero. Because (3) involves higher-order interaction terms, we introduce the following notation.

172 Let $2^{[p]}$ denote the power set of the index set $[p] = \{1, \dots, p\}$. For any vector $X = (X_1, \dots, X_p)$
 173 $\in \mathbb{R}^p$ and subset $S \subseteq [p]$, define the *interaction feature* $B^S(X) = \prod_{j \in S} X_j$ with $B^\emptyset(X) = 1$.
 174 Collecting all 2^p such terms yields the *extended feature map*: $\Phi(X) = [B^S(X)]_{S \in 2^{[p]}} \in \mathbb{R}^{2^p}$,
 175 ordered according to the graded-lexicographic rule described in Appendix C.3. Explicitly,

$$\Phi(X) = [1, X_1, \dots, X_p, X_1 X_2, X_1 X_3, \dots, X_{p-1} X_p, X_1 X_2 X_3, \dots, X_1 \cdots X_p]. \quad (5)$$

176 For example, when $p = 3$ and $X = (X_1, X_2, X_3)$, then the extended feature vector $\Phi(X) =$
 177 $[1, X_1, X_2, X_3, X_1 X_2, X_1 X_3, X_2 X_3, X_1 X_2 X_3] \in \mathbb{R}^8$. When Φ is applied to data matrix $\mathbf{X} \in$
 178 $\mathbb{R}^{n \times p}$, it operates *row-wise*.

179 For each j and order π , define the relevant natural parameters in the vector $\mathbf{f}_{\pi,j} \in \mathbb{R}^{2^{j-1}}$ stored in
 180 the *same* graded-lexicographic order, but relative to the $(X_{\pi(1)}, \dots, X_{\pi(j-1)})$. Concretely,

$$\mathbf{f}_{\pi,j} = [\underbrace{f_{\pi,j}^{\pi(j)}}_{\text{constant}}, \underbrace{f_{\pi,j}^{\pi(1)\pi(j)}}_{X_{\pi(1)}}, \underbrace{f_{\pi,j}^{\pi(2)\pi(j)}}_{X_{\pi(2)}}, \dots, \underbrace{f_{\pi,j}^{\pi(j-1)\pi(j)}}_{X_{\pi(j-1)}}, \underbrace{f_{\pi,j}^{\pi(1)\pi(2)\pi(j)}}_{X_{\pi(1)}X_{\pi(2)}}, \dots, \underbrace{f_{\pi,j}^{\pi(1)\dots\pi(j-1)\pi(j)}}_{X_{\pi(1)}\dots X_{\pi(j-1)}}]. \quad (6)$$

181 Each underbrace indicates the interaction term—either a constant, a single feature, or a product of
 182 features from $\{X_{\pi(1)}, \dots, X_{\pi(j-1)}\}$ —that the coefficient corresponds to. Thus, the elements of $\mathbf{f}_{\pi,j}$
 183 align exactly with the entries of the vector $\Phi((X_{\pi(1)}, \dots, X_{\pi(j-1)}))$. The special case used in (3) is
 184 recovered by taking $\pi = (1, \dots, p)$ and $j = p$.

185 With these definitions in place we can now recover the full causal structure and its param-
 186 eters. Fix an ordering π and consider any $j \in [p]$. By Corollary 1, both the marginal
 187 $P(X_{\pi(j)}, X_{\pi(1)}, \dots, X_{\pi(j-1)})$ and conditional $P(X_{\pi(j)} \mid X_{\pi(1), \dots, \pi(j-1)})$ remain multivariate
 188 Bernoulli. Let $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_p) \in \mathbb{R}^{n \times p}$ be n i.i.d. samples from $X \sim \text{MultiBernoulli}(\mathbf{p})$,
 189 one sample per row. Hence we can apply the same logic as in Section 4.1

- 190 • Form extended feature vector $\Phi((\mathbf{X}_{\pi(1)}, \dots, \mathbf{X}_{\pi(j-1)}))$ and run a single logistic regression
 191 of $\mathbf{X}_{\pi(j)}$ on these features to recover natural parameters $\mathbf{f}_{\pi,j} \in \mathbb{R}^{2^{j-1}}$
- 192 • Apply the edge criterion in (4) to $\mathbf{f}_{\pi,j}$ to determine the parent set $\text{PA}(\pi(j))$.
- 193 • Repeat steps above for $j \in [p]$ to construct the DAG G_π and parameter $\mathbf{f}_\pi = \{\mathbf{f}_{\pi,j}\}_{j=1}^p$.
- 194 • Iterate over all permutations π to enumerate every graph-parameter pair.

195 Complete details on such procedure can be founded in Appendix C.4.

196 **Theorem 1.** Let $\mathbf{p} > 0$ and fix any topological sort π . Consider the population case ($n \rightarrow \infty$). Let
 197 (\mathbf{f}_π, G_π) be the output of procedure above. If $Y \in \mathbb{R}^p$ are generated by the following structural
 198 equations

$$Y_{\pi(j)} \sim \text{Bernoulli} \left(\text{logistic} \left(\mathbf{f}_{\pi,j}^\top \Phi((Y_{\pi(1)}, \dots, Y_{\pi(j-1)})) \right) \right) \quad \forall j = 1, \dots, p \quad (7)$$

199 then $Y \sim \text{MultiBernoulli}(\mathbf{p})$.

200 We consider in the population case to eliminate finite sample estimation error. The assumption
 201 $\mathbf{p} > 0$ is a mild regularity condition routinely adopted in structure learning problems [45, 33, 19] and
 202 guarantees that the logistic regressions used in the procedure above always have unique solutions.

203 **Remark 1.** Equation (7) is written as a form of structural equation model (SEM)¹ [32], a framework
 204 widely used for structure learning because it provides a generative description of how each variable
 205 arises from its direct causes. A central appeal of SEMs is their universality: every probability
 206 distribution can, in principle, be represented by a suitably chosen SEM (Proposition 7.1 in [32]).
 207 In practice, however, SEM-based methods almost always impose strong parametric forms—linear,
 208 additive, or low-order interactions [43, 45, 17]—which risk ignoring important structure in real

¹Formal introduction can be founded in Appendix C.5

209 data [11, 16]. The problem is particularly acute for discrete data, where higher-order dependencies
 210 are easily overlooked, sharply limiting the model’s expressive power. By contrast, Theorem 1 shows
 211 that general binary distributions necessarily involve higher-order interaction terms. Omitting these
 212 terms therefore precludes an exact representation and may lead to incorrect causal conclusions.

213 Thus, the Theorem 1 indicates that, for every topological order π we can construct a distinct SEM that
 214 reproduces the same distribution. The model is therefore *non-identifiable*: multiple parameter–graph
 215 pairs (\mathbf{f}_π, G_π) obtained by procedure before give rise to the identical law $\text{MultiBernoulli}(\mathbf{p})$. Collect
 216 all such pairs into the *equivalence class*

$$\mathcal{E}(\mathbf{p}) = \{(\mathbf{f}_\pi, G_\pi) : (\mathbf{f}_\pi, G_\pi), \forall \pi\}$$

217 The cardinality of $\mathcal{E}(\mathbf{p})$ is at most $p!$, corresponding to the number of permutations of p variables.

218 **Remark 2.** It is well-known that causal DAGs are non-identifiable from observational data alone [18].
 219 However, Theorem 1 provides a novel and comprehensive characterization of all DAG–parameter
 220 pairs that exactly reproduce a given distribution within the general multivariate Bernoulli framework.

221 4.3 Minimal equivalence class

222 Because observational data cannot distinguish among members of $\mathcal{E}(\mathbf{p})$, we seek the simplest
 223 representation. Specifically, the DAG with the fewest edges. Let s_G denote the number of directed
 224 edges in a graph G ; our objective is to find the pair $(\mathbf{f}_\pi, G_\pi) \in \mathcal{E}(\mathbf{p})$ with $\min s_{G_\pi}$.

225 **Definition 3** (Minimality [14]). (\mathbf{f}_π, G_π) is the minimal element in equivalence class $\mathcal{E}(\mathbf{p})$ if
 226 $s_{G_\pi} \leq s_{G_{\tilde{\pi}}}, \forall (\mathbf{f}_{\tilde{\pi}}, G_{\tilde{\pi}}) \in \mathcal{E}(\mathbf{p})$. The set of all the minimal element in equivalence class $\mathcal{E}(\mathbf{p})$ is
 227 referred to as the minimal equivalence class $\mathcal{E}_{\min}(\mathbf{p})$

$$\mathcal{E}_{\min}(\mathbf{p}) = \{(\mathbf{f}_\pi, G_\pi) : (\mathbf{f}_\pi, G_\pi) \text{ is the minimal element}, (\mathbf{f}_\pi, G_\pi) \in \mathcal{E}(\mathbf{p})\} \quad (8)$$

228 For the procedure in Section 4.2, if we retains those pairs (\mathbf{f}_π, G_π) with the fewest edges, thereby
 229 recovering $\mathcal{E}_{\min}(\mathbf{p})$. Minimality is closely related to the *Sparest Markov Representation* (SMR)
 230 assumption [35, 23], which posits that, among all DAGs compatible with a distribution, the sparsest
 231 one is unique up to Markov equivalence, i.e., encoding the same conditional independence relationship.
 232 SMR is weaker than the well-know *faithfulness* assumption [32]: if a distribution P is faithful to
 233 a DAG G , then (G, P) satisfies SMR. Hence, targeting the sparsest graph is both principled and
 234 practically appealing. Let $\mathcal{M}(G)$ denote the Markov Equivalence class of a DAG G , i.e. the set of all
 235 DAGs encoding the same conditional-independence relations. Formal definitions of these concepts
 236 appears in Appendix C.6.

237 **Theorem 2.** For any $\mathbf{p} > 0$ and consider the population case ($n \rightarrow \infty$). Under the SMR assumption
 238 (or faithfulness assumption), if $(\mathbf{f}_{\pi_1}, G_{\pi_1}), (\mathbf{f}_{\pi_2}, G_{\pi_2}) \in \mathcal{E}_{\min}(\mathbf{p})$, then $\mathcal{M}(G_{\pi_1}) = \mathcal{M}(G_{\pi_2})$

239 5 Continuous structure learning for general binary data

240 The procedure in previous section is purely combinatorial and thus fails to scale: it requires fitting $p!$
 241 logistic regressions, so the computational cost grows exponentially with the number of nodes p . To
 242 overcome this bottleneck, we leverage recent advances in differentiable structure learning [46, 47].
 243 Specifically, we recast the search for the sparsest binary–data DAG as a *single* differentiable program,
 244 which can then be tackled by standard gradient-based optimizers.

245 **Parameterization** Let $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_p)$ be n i.i.d. samples from $X \sim \text{MultiBernoulli}(\mathbf{p})$, one
 246 sample per row. Define the parameter matrix

$$\mathbf{H}_j = (\underbrace{h^{j,0}}_{\text{constant}}, \underbrace{h^{j,1}, \dots, h^{j,p}}_{\text{first order}}, \underbrace{h^{j,12}, \dots, h^{j,(p-1)p}}_{\text{second order}}, \underbrace{\dots}_{\text{third to } (p-1)\text{-th order}}, \underbrace{h^{j,123\dots p}}_{\text{p-th order}})^\top \in \mathbb{R}^{2^p} \quad (9)$$

247 Let $\mathbf{H} = (\mathbf{H}_1, \dots, \mathbf{H}_p) \in \mathbb{R}^{2^p \times p}$. Here \mathbf{H}_j plays the role of column j in the weighted adjacency
 248 matrix. Unlike a linear model, where a single entry W_{ij} signals the edge $X_i \rightarrow X_j$ [46], the multiple
 249 coefficients in \mathbf{H}_j jointly determine whether such an edge exists.

250 **Weighted adjacency matrix** Define the induced adjacency matrix

$$[W(\mathbf{H})]_{ij} = \sum_{S \subseteq [p], i \in S} (h^{j,S})^2 \quad (10)$$

251 so $[W(\mathbf{H})]_{ij} > 0$ if and only if some interaction involving X_i contributes to the equation for X_j , that
252 is some coefficient $h^{j,S}$ with $i \in S$ is non-zero. Self-loops are forbidden, so we impose

$$h^{j,S} = 0 \quad \text{whenever } j \in S, \quad \forall j \in [p], S \subseteq [p].$$

253 **Score function** By Theorem 1 the negative log-likelihood of the multivariate Bernoulli model—i.e.
254 the logistic or cross entropy loss—provides a suitable score. Applying the extended feature map (5)
255 row-wise to the data matrix \mathbf{X} yields the score function

$$\ell(\mathbf{H}; \mathbf{X}) = \frac{1}{n} \sum_{i=1}^p \mathbf{1}_n^\top (\log(\mathbf{1}_n + \exp(\Phi(\mathbf{X})\mathbf{H})) - \mathbf{X}_i \circ (\Phi(\mathbf{X})\mathbf{H}))$$

256 where \circ denotes the Hadamard product and \exp is applied element-wise, see Appendix C.7 for details.

257 **Regularization** While $\ell(\mathbf{H}; \mathbf{X})$ identifies the equivalence class $\mathcal{E}(\mathbf{p})$, our objective is a *minimal*
258 *equivalence class* $\mathcal{E}_{\min}(\mathbf{p})$. Simply adding a ℓ_0 [18] penalty would break differentiability, and an
259 ℓ_1 penalty is both biased and imprecise in edge counting. Instead, we adopt the smooth *quasi*
260 minimax-concave penalty (MCP) [15]:

$$\text{quasi-MCP:} \quad p_{\lambda,\delta}(t) = \lambda \left[\left(|t| - \frac{t^2}{2\delta} \right) \mathbb{1}(|t| < \delta) + \frac{\delta}{2} \mathbb{1}(|t| > \delta) \right] \quad (11)$$

261 where $\mathbb{1}\{\cdot\}$ is the indicator function. $\psi_{\lambda,\delta}$ is quadratic on $[0, \delta]$ and flat beyond δ , smoothly approxi-
262 mating an ℓ_0 penalty and thereby encouraging a sparse $W(\mathbf{H})$ without sacrificing differentiability.
263 Let $p_{\lambda,\delta}(W(\mathbf{H})) = \sum_{i \neq j} p_{\lambda,\delta}([W(\mathbf{H})]_{ij})$. Define the regularized score

$$s(\mathbf{H}; \lambda, \delta, \mathbf{X}) = s(\mathbf{H}; \mathbf{X}) + p_{\lambda,\delta}(W(\mathbf{H})) \quad (12)$$

264 **Recovering $\mathcal{E}_{\min}(\mathbf{p})$** We formulate this task as the single continuous optimization problem

$$\begin{aligned} \min_{\mathbf{H}} \quad & s(\mathbf{H}; \lambda, \delta, \mathbf{X}) \\ \text{subject to} \quad & h(W(\mathbf{H})) = 0 \\ & h^{j,S} = 0 \text{ if } j \in S \quad \forall j \in [p], \forall S \subseteq [p] \end{aligned} \quad (13)$$

265 where $h : \mathbb{R}^{p \times p} \rightarrow [0, \infty)$ is the differentiable acyclicity constraint [46, 4, 27, 44], satisfying
266 $h(W) = 0$ iff W is a DAG. To study the optimal solutions of (13), define the set of global minimizers

$$\mathcal{O}_{n,\lambda,\delta} = \{(\mathbf{H}^*, G(W(\mathbf{H}^*))) : \mathbf{H}^* \text{ is a minimizer of (13)}\} \quad (14)$$

267 where $G(W)$ denotes the graph encoded by the adjacency matrix W via (10). Although (13) optimizes
268 only over \mathbf{H} , we include the induced graph to match the notation of $\mathcal{E}(\mathbf{p})$ and $\mathcal{E}_{\min}(\mathbf{p})$.

269 Ideally, we want the optimal solution set to coincide with the minimal equivalence class, i.e. $\mathcal{O}_{n,\lambda,\delta} =$
270 $\mathcal{E}_{\min}(\mathbf{p})$. However, it is unclear whether suitable values of (λ, δ) exist. The next result shows
271 that, in the population limit, such values can always be chosen. Let $\mathcal{O}_{\infty,\lambda,\delta}$ denote the set of
272 minimizers of (13) when the empirical loss $s(\mathbf{H}; \lambda, \delta, \mathbf{X})$ is replaced by its population counterpart
273 $\mathbb{E}[s(\mathbf{H}; \lambda, \delta, \mathbf{X})]$.

274 **Theorem 3.** *For any $\mathbf{p} > 0$, there exist $\lambda, \delta > 0$ sufficiently small such that $\mathcal{O}_{\infty,\lambda,\delta} =$
275 $\mathcal{E}_{\min}(\mathbf{p})$. Moreover, under Sparsest Markov Representation (or faithfulness) assumption, for any
276 $(\mathbf{f}_{\pi_1}, G_{\pi_1}), (\mathbf{f}_{\pi_2}, G_{\pi_2}) \in \mathcal{O}_{\infty,\lambda,\delta}$, it holds $\mathcal{M}(G_{\pi_1}) = \mathcal{M}(G_{\pi_2})$.*

277 Here, $\mathcal{M}(G)$ is the Markov equivalence class of G . The proof adapts Theorem 4 of Deng et al.
278 [15], but requires additional technical work to accommodate the multivariate Bernoulli setting; The
279 theorem implies that, with appropriately small regularization, every global optimum of (13) recovers
280 a sparsest DAG and preserves all conditional-independence relations.

281 **Connection to prior continuous methods on discrete data.** Existing empirical studies [12, 4]
 282 often generate binary data by passing a *linear* combination of parent variables through a logistic link,
 283 then apply continuous-optimization techniques without formal justification. Theorem 3 provides
 284 that justification: such simulations instantiate a special case of our general model. To formalize this
 285 setting, we characterize this with the following assumption.

286 **Assumption A** (First-order logistic model). *There exist a DAG G^0 such that $X = (X_1, \dots, X_p)$ are*
 287 *generated according to the following structure equation model*

$$X_j \mid X_{\text{PA}(j)} \sim \text{Bernoulli}(\text{logistic}(w_j^\top X + c_j)) \quad (15)$$

288 where $X_{\text{PA}(j)}$ are the parents node of X_j in G^0 . Here $[w_j]_{[p] \setminus \text{PA}(j)} = 0$, $[w_j]_{\text{PA}(j)} \neq 0$, $c_j \in \mathbb{R}$.

289 Under Assumption A, only *first-order* interaction terms are present; all higher-order coefficients in (9)
 290 vanish. Consequently \mathbf{H} shrinks from $2^p \times p$ to $(p+1) \times p$, and the optimization problem becomes
 291 tractable even for moderate p . A detailed formulation and corresponding consistency theorem are
 292 provided in Appendix C.8.

293 6 Experiments

294 We solve (13) using the DAGMA optimization framework of Bello et al. [4], which tackles a
 295 sequence of unconstrained problems by incorporating the acyclicity constraint as a penalty term with
 296 an increasing weight. Our method—denoted DAGMA-HO (for “higher-order”)—is compared against
 297 several baselines, including DAGMA [4], PC [37], and FGES [34]. Our primary empirical results
 298 are shown in Figure 1 and 2. We use the structural Hamming distance (SHD) as the main metric to
 299 evaluate the difference between the estimated and the ground truth graph. Lower SHD indicates better
 300 estimation accuracy. Given the multivariate Bernoulli distribution is nonidentifiable, we compare the
 301 completed partially directed acyclic graph (CPDAG) of the estimated and ground truth graph. Full
 302 experimental details are provided in Appendix D.

303 In Figure 1(a), we simulate data \mathbf{X} according to the SEM in (7), including both first-order effects
 304 and second-order interactions among parents whenever a node has multiple parents. We observe
 305 that DAGMA-HO (ours) achieves competitive performance relative to the baselines. In contrast,
 306 DAGMA-1ST (the original DAGMA), which models only first-order effects, exhibits markedly poor
 307 recovery when second-order interactions are present. These results underscore that a purely linear
 308 form is insufficient to capture the complex dependencies in general discrete data and can lead to
 309 substantial estimation errors whenever higher-order terms play a role. In Figure 1(b), we repeat
 310 the simulation while ablating all first-order effects for nodes with multiple parents, retaining only
 311 the highest-order interaction term. Under these conditions, DAGMA-HO (ours) consistently attains
 312 near-optimal recovery performance across all settings, showing the robustness of our method.

313 In Figure 2, we consider a larger DAG, where including all possible interaction terms would cause the
 314 feature map $\Phi(\mathbf{X})$ to grow exponentially and render the optimization in (13) intractable. To address
 315 this, we propose a two-stage approach, which we call NOTEARS-MLP-REG. First, we adapt the
 316 original NOTEARS-MLP framework of Zheng et al. [47] to handle discrete data, and use it to learn
 317 an initial graph. From this graph we extract a topological ordering π . In the second (finetuning) stage,
 318 we follow the procedure of Section 4.2: we construct only those interaction features consistent with
 319 π , and then fit logistic regressions with first order and second order to recover the final edge structure
 320 G . NOTEARS-MLP-REG attains performance on par with existing baselines. While all methods
 321 degrade as graph size and density increase, our approach exhibits greater robustness and consistently
 322 recovers meaningful structures.

323 Further details on the experimental setup and additional results can be found in the appendix.

324 7 Conclusion

325 We have introduced a fully general differentiable framework for structure learning in binary data,
 326 based on the multivariate Bernoulli distribution. Our key contributions include (i) a complete
 327 characterization of non-identifiability in the general discrete setting and a constructive procedure
 328 to recover the minimal equivalence class (Theorems 1–2), (ii) a fully differentiable constrained
 329 programming that provably recovers a sparsest DAG up to Markov equivalence (Theorem 3), and

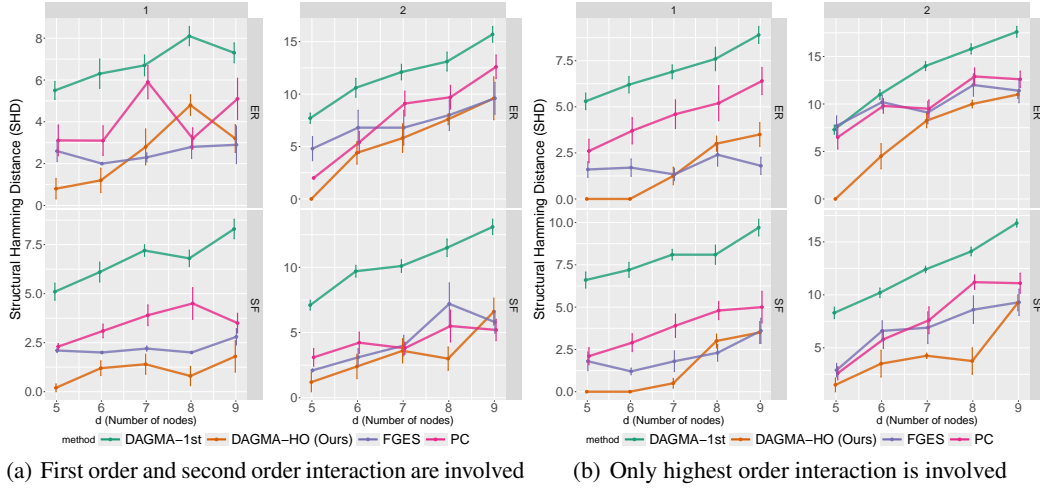


Figure 1: Results in terms of SHD between MECs of estimated graph and ground truth. Lower is better. Column: $k = \{1, 2\}$. Row: random graph types. $\{ER, SF\}$ - $k = \{\text{Scale-Free, Erdős-Rényi}\}$ graphs with kd expected edges. Here $p = \{5, 6, 7, 8, 9\}$. DAGMA [4] is renamed as “DAGMA-1st”, to emphasize only linear term is used. Error bars denote the standard error computed over 10 replications.

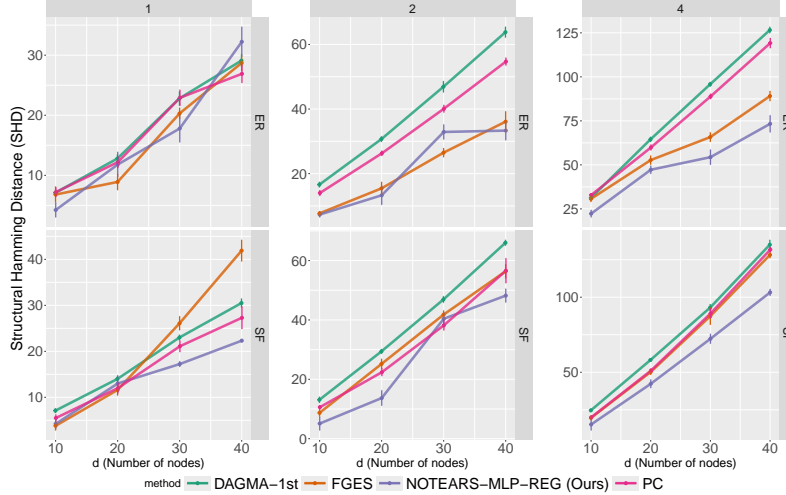


Figure 2: Results in terms of SHD between MECs of estimated graph and ground truth. Lower is better. Column: $k = \{1, 2, 4\}$. Row: random graph types. $\{ER, SF\}$ - $k = \{\text{Scale-Free, Erdős-Rényi}\}$ graphs with kd expected edges. Here $p = \{10, 20, 30, 40\}$. NOTEARS-MLP-REG is our two stage approach. DAGMA [4] is renamed as “DAGMA-1st”, to emphasize only linear term is used.

330 (iii) empirical validation on synthetic graphs of varying size, demonstrating that our method captures
 331 complex higher-order dependencies where existing approaches fail.

332 To scale beyond small graphs, we further propose a practical two-stage heuristic, NOTEARS-MLP-
 333 REG, which first learns a coarse structure via NOTEARS-MLP adapted to discrete data and then
 334 refines edge estimates by fitting logistic regressions along the inferred topological order. While this
 335 approach yields strong numerical performance on larger networks, its theoretical properties remain
 336 unexplored. In future work, we aim to develop principled, scalable algorithms for discrete structure
 337 learning that retain both computational tractability and rigorous identifiability guarantees in regimes
 338 with thousands of nodes.

References

- [1] Andrews, B., Ramsey, J. and Cooper, G. F. [2019], Learning high-dimensional directed acyclic graphs with mixed data-types, in ‘The 2019 ACM SIGKDD Workshop on Causal Discovery’, PMLR, pp. 4–21.
- [2] Andrews, B., Ramsey, J., Sanchez Romero, R., Camchong, J. and Kummerfeld, E. [2023], ‘Fast scalable and accurate discovery of dags using the best order score search and grow shrink trees’, *Advances in neural information processing systems* **36**, 63945–63956.
- [3] Ban, T., Chen, L., Wang, X., Wang, X., Lyu, D. and Chen, H. [2024], ‘Differentiable structure learning with partial orders’, *Advances in Neural Information Processing Systems* **37**, 117426–117455.
- [4] Bello, K., Aragam, B. and Ravikumar, P. [2022], ‘Dagma: Learning dags via m-matrices and a log-determinant acyclicity characterization’, *Advances in Neural Information Processing Systems* **35**, 8226–8239.
- [5] Budhathoki, K. and Vreeken, J. [2018], Accurate causal inference on discrete data, in ‘2018 IEEE International Conference on Data Mining (ICDM)’, IEEE, pp. 881–886.
- [6] Cai, R., Qiao, J., Zhang, K., Zhang, Z. and Hao, Z. [2018], ‘Causal discovery from discrete data using hidden compact representation’, *Advances in neural information processing systems* **31**.
- [7] Chickering, D. M. [1996], ‘Learning bayesian networks is np-complete’, *Learning from data: Artificial intelligence and statistics V* pp. 121–130.
- [8] Chickering, D. M. [2002], ‘Optimal structure identification with greedy search’, *Journal of machine learning research* **3**(Nov), 507–554.
- [9] Chickering, D. M., Heckerman, D. and Meek, C. [2004], ‘Large-sample learning of bayesian networks is np-hard’, *Journal of Machine Learning Research* **5**(Oct), 1287–1330.
- [10] Cui, R., Groot, P. and Heskes, T. [2016], Copula pc algorithm for causal discovery from mixed data, in ‘Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part II 16’, Springer, pp. 377–392.
- [11] Dai, B., Ding, S. and Wahba, G. [2013], ‘Multivariate bernoulli distribution’.
- [12] Deng, C., Bello, K., Aragam, B. and Ravikumar, P. K. [2023], Optimizing notears objectives via topological swaps, in ‘International Conference on Machine Learning’, PMLR, pp. 7563–7595.
- [13] Deng, C., Bello, K., Ravikumar, P. and Aragam, B. [2023], ‘Global optimality in bivariate gradient-based dag learning’, *Advances in Neural Information Processing Systems* **36**, 17929–17968.
- [14] Deng, C., Bello, K., Ravikumar, P. and Aragam, B. [2024a], ‘Likelihood-based differentiable structure learning’, *arXiv preprint arXiv:2410.06163*.
- [15] Deng, C., Bello, K., Ravikumar, P. and Aragam, B. [2024b], ‘Markov equivalence and consistency in differentiable structure learning’, *Advances in Neural Information Processing Systems* **37**, 91756–91797.
- [16] Enouen, J. and Sugiyama, M. [2024], ‘A complete decomposition of kl error using refined information and mode interaction selection’, *arXiv preprint arXiv:2410.11964*.
- [17] Gu, J., Fu, F. and Zhou, Q. [2019], ‘Penalized estimation of directed acyclic graphs from discrete data’, *Statistics and Computing* **29**(1), 161–176.
- [18] Heckerman, D., Geiger, D. and Chickering, D. M. [1995], ‘Learning bayesian networks: The combination of knowledge and statistical data’, *Machine learning* **20**, 197–243.
- [19] Hoyer, P., Janzing, D., Mooij, J. M., Peters, J. and Schölkopf, B. [2008], ‘Nonlinear causal discovery with additive noise models’, *Advances in neural information processing systems* **21**.

- [20] Huang, B., Zhang, K., Lin, Y., Schölkopf, B. and Glymour, C. [2018], Generalized score functions for causal discovery, in ‘Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining’, pp. 1551–1560.
- [21] Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M. and Klein, M. [2002], *Logistic regression*, Springer.
- [22] Lachapelle, S., Brouillard, P., Deleu, T. and Lacoste-Julien, S. [2019], ‘Gradient-based neural dag learning’, *arXiv preprint arXiv:1906.02226*.
- [23] Lam, W.-Y., Andrews, B. and Ramsey, J. [2022], Greedy relaxations of the sparsest permutation algorithm, in ‘Uncertainty in Artificial Intelligence’, PMLR, pp. 1052–1062.
- [24] Li, C. and Shimizu, S. [2018], ‘Combining linear non-gaussian acyclic model with logistic regression model for estimating causal structure from mixed continuous and discrete data’, *arXiv preprint arXiv:1802.05889*.
- [25] Liu, F. and Chan, L. [2016], ‘Causal inference on discrete data via estimating distance correlations’, *Neural computation* **28**(5), 801–814.
- [26] Maxwell Chickering, D. and Heckerman, D. [1997], ‘Efficient approximations for the marginal likelihood of bayesian networks with hidden variables’, *Machine learning* **29**, 181–212.
- [27] Nazaret, A., Hong, J., Azizi, E. and Blei, D. [2023], ‘Stable differentiable causal discovery’, *arXiv preprint arXiv:2311.10263*.
- [28] Ng, I., Ghassami, A. and Zhang, K. [2020], ‘On the role of sparsity and dag constraints for learning linear dags’, *Advances in Neural Information Processing Systems* **33**, 17943–17954.
- [29] Ng, I., Lachapelle, S., Ke, N. R., Lacoste-Julien, S. and Zhang, K. [2022], On the convergence of continuous constrained optimization for structure learning, in ‘International Conference on Artificial Intelligence and Statistics’, Pmlr, pp. 8176–8198.
- [30] Pearl, J. [2009], *Causality*, Cambridge university press.
- [31] Peters, J., Janzing, D. and Schölkopf, B. [2010], Identifying cause and effect on discrete data using additive noise models, in ‘Proceedings of the thirteenth international conference on artificial intelligence and statistics’, JMLR Workshop and Conference Proceedings, pp. 597–604.
- [32] Peters, J., Janzing, D. and Schölkopf, B. [2017], *Elements of causal inference: foundations and learning algorithms*, The MIT Press.
- [33] Peters, J., Mooij, J. M., Janzing, D. and Schölkopf, B. [2014], ‘Causal discovery with continuous additive noise models’, *JMLR*.
- [34] Ramsey, J., Glymour, M., Sanchez-Romero, R. and Glymour, C. [2017], ‘A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images’, *International journal of data science and analytics* **3**, 121–129.
- [35] Raskutti, G. and Uhler, C. [2018], ‘Learning directed acyclic graph models based on sparsest permutations’, *Stat* **7**(1), e183.
- [36] Seng, J., Zečević, M., Dhimi, D. S. and Kersting, K. [2024], Learning large dags is harder than you think: Many losses are minimal for the wrong dag, in ‘The Twelfth International Conference on Learning Representations’.
- [37] Spirtes, P. and Glymour, C. [1991], ‘An algorithm for fast recovery of sparse causal graphs’, *Social science computer review* **9**(1), 62–72.
- [38] Spirtes, P., Glymour, C. N. and Scheines, R. [2000], *Causation, prediction, and search*, MIT press.
- [39] Tsamardinos, I., Aliferis, C. F., Statnikov, A. R. and Statnikov, E. [2003], Algorithms for large scale markov blanket discovery., in ‘FLAIRS’, Vol. 2, pp. 376–81.

- 431 [40] Wei, D., Gao, T. and Yu, Y. [2020], ‘Dags with no fears: A closer look at continuous optimization
432 for learning bayesian networks’, *Advances in Neural Information Processing Systems* **33**, 3895–
433 3906.
- 434 [41] Wei, Y., Li, X., Lin, L., Zhu, D. and Li, Q. [2022], ‘Causal discovery on discrete data via
435 weighted normalized wasserstein distance’, *IEEE Transactions on Neural Networks and Learn-
436 ing Systems* **35**(4), 4911–4923.
- 437 [42] Wenjuan, W., Lu, F. and Chunchen, L. [2018], Mixed causal structure discovery with application
438 to prescriptive pricing, in ‘Proceedings of the 27th International Joint Conference on Artificial
439 Intelligence’, pp. 5126–5134.
- 440 [43] Ye, Q., Amini, A. A. and Zhou, Q. [2024], ‘Federated learning of generalized linear causal
441 networks’, *IEEE Transactions on Pattern Analysis and Machine Intelligence* .
- 442 [44] Yu, Y., Chen, J., Gao, T. and Yu, M. [2019], Dag-gnn: Dag structure learning with graph neural
443 networks, in ‘International conference on machine learning’, PMLR, pp. 7154–7163.
- 444 [45] Zeng, Y., Shimizu, S., Matsui, H. and Sun, F. [2022], Causal discovery for linear mixed data, in
445 ‘Conference on Causal Learning and Reasoning’, PMLR, pp. 994–1009.
- 446 [46] Zheng, X., Aragam, B., Ravikumar, P. K. and Xing, E. P. [2018], ‘Dags with no tears: Continu-
447 ous optimization for structure learning’, *Advances in neural information processing systems*
448 **31**.
- 449 [47] Zheng, X., Dan, C., Aragam, B., Ravikumar, P. and Xing, E. [2020], Learning sparse non-
450 parametric dags, in ‘International Conference on Artificial Intelligence and Statistics’, Pmlr,
451 pp. 3414–3425.
- 452 [48] Zhu, S., Ng, I. and Chen, Z. [2019], ‘Causal discovery with reinforcement learning’, *arXiv
453 preprint arXiv:1906.04477* .

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: Each claim in the abstract can be founded in the main paper.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: In the conclusion section, we have briefly discussed about future work, which address the limitation of the paper.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: For each theorem statement, we include the condition/assumption needed, see Theorem 1,2,3.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We can provide the experiments' details in the appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code will be available if the paper get accepted.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: See the experiment section in the appendix for reproducible purpose.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Error bars are provided in the Experiment in the main paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Such details are included in the appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: Our work is more related to theory and methodology.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification: We mainly focus on the theoretical findings.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: It is not relevant to our research topic.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [NA]

Justification: It is unclear to me.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: Unclear to me.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Not relevant.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: No human involved.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [No]

768

Justification: Our work is original.

769

Guidelines:

770

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.

771

772

- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

773

SUPPLEMENTARY MATERIAL

Differentiable Structure Learning for General Binary Data

A Preliminary Technical Results

In this appendix, we present several key technical results that are essential to our proofs.

Lemma 1. *Suppose $X \sim \text{MultiBernoulli}(\mathbf{p})$ with $\mathbf{p} > 0$. Then for every subset $S \subseteq [p]$, the marginal vector X_S satisfies*

$$X_S \sim \text{MultiBernoulli}(\mathbf{p}_S),$$

where $\mathbf{p}_S > 0$.

The marginal vector X_S remains multivariate Bernoulli with natural parameter \mathbf{p}_S , and positivity of \mathbf{p} indicates the positivity of \mathbf{p}_S .

Lemma 2. *Suppose $X \sim \text{MultiBernoulli}(\mathbf{p})$ in the natural parameterization, or equivalently $X \sim \text{MultiBernoulli}(\mathbf{f})$ in the general parameterization. Then*

$$\mathbf{p} > 0 \iff |\mathbf{f}| < \infty.$$

The general parameter vector \mathbf{p} is strictly positive if and only if its corresponding natural parameter \mathbf{f} is strictly finite.

Lemma 3. *Let $\mathbf{p} > 0$ and suppose $X \sim \text{MultiBernoulli}(\mathbf{p})$. Fix any topological order π and index $j \in [p]$. Define the population negative log-likelihood*

$$\ell(w) = \mathbb{E} \left[\log(1 + \exp(w^\top \Phi(X_{\pi(1)}, \dots, X_{\pi(j-1)}))) - X_{\pi(j)} w^\top \Phi(X_{\pi(1)}, \dots, X_{\pi(j-1)}) \right], \quad w \in \mathbb{R}^{2^{j-1}}.$$

Then $\ell(w)$ is strictly convex and therefore admits a unique minimizer $w_{\pi,j}^*$.

This optimization problem is equivalent to fitting, in the population limit, a logistic regression of $X_{\pi(j)}$ on $\Phi((X_{\pi(1)}, \dots, X_{\pi(j-1)}))$. Because $\mathbf{f}_{\pi,j}$, as defined in Section 4.2, is one solution to the optimization above, and we show optimal solution is unique, then $w_{\pi,j}^* = \mathbf{f}_{\pi,j}$. Consequently, logistic regression perfectly recovers the true natural parameter.

Corollary 2. *Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be differentiable and m -strongly convex, i.e.*

$$f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \frac{m}{2} \|y - x\|^2 \quad \forall x, y \in \mathbb{R}^d$$

Denote by x^* the unique minimizer f , so $f(x^*) = \min_x f(x)$. Then for any constant $c \geq f(x^*)$ the subset

$$L_c = \{x \in \mathbb{R}^p : f(x) \leq c\}$$

is bounded.

Corollary 2 asserts that any strongly convex function possesses bounded level sets, which is used in the proof of Theorem 3.

B Proofs

In this appendix, we provide detailed proofs of the main results.

B.1 Proof of Lemma 1

From the proof of Corollary 1, we know that $X_S \sim \text{MultiBernoulli}(\mathbf{p}_S)$. Moreover, because $\mathbf{p} > 0$, then

$$P(X_S = x_s) = \sum_{x_{[p] \setminus S} \in \{0,1\}^{p-|S|}} P(X_S = x_s, X_{[p] \setminus S} = x_{[p] \setminus S}) > 0 \quad (16)$$

Therefore, for any x_s , $P(X_S = x_s) > 0$. So, $\mathbf{p}_S > 0$.

806 B.2 Proof of Lemma 2

807 **Sufficiency** By (62), each natural-parameter probability satisfies

$$\exp(f^{j_1 j_2 \dots j_r}) \quad (17)$$

$$= \frac{\prod \text{p(even \# zeros among } j_1, j_2, \dots, j_r \text{ components and other components are all zero)}}{\prod \text{p(odd \# zeros among } j_1, j_2, \dots, j_r \text{ components and other components are all zero)}}, \quad (18)$$

808 and since $p > 0$, every term $\exp(f^{j_1 \dots j_r})$ is strictly positive and finite. Hence

$$0 < \exp(f^{j_1 \dots j_r}) < \infty \iff |f^{j_1 \dots j_r}| < \infty. \quad (19)$$

809 **Necessity** Note that

$$S^{j_1 j_2 \dots j_r} = \sum_{1 \leq s \leq r} f^{j_s} + \sum_{1 \leq s < t \leq r} f^{j_s j_t} + \dots + f^{j_1 j_2 \dots j_r} \quad (20)$$

810 If $|f| < \infty$, then $|S^{j_1 \dots j_r}| < \infty$, so

$$0 < \exp(S^{j_1 j_2 \dots j_r}) < \infty \quad (21)$$

811 The joint probability of observing ones in positions j_1, \dots, j_r is

$$\begin{aligned} & \text{p}(j_1, j_2, \dots, j_r \text{ positions are one, others are zero}) \\ &= \frac{\exp(S^{j_1 j_2 \dots j_r})}{\exp(b(f))}. \\ &= \frac{\exp(S^{j_1 j_2 \dots j_r})}{\sum_{r=1}^K \left[1 + \left(\sum_{1 \leq j_1 < j_2 < \dots < j_r \leq K} \exp[S^{j_1 j_2 \dots j_r}] \right) \right]} \end{aligned} \quad (22)$$

812 which lies in the interval $(0, 1)$:

$$0 < p(j_1, j_2, \dots, j_r \text{ positions are one, others are zero}) < 1 \quad (23)$$

813 B.3 Proof of Lemma 3

814 Fix a topological order π and an index $j \in [p]$. To simplify notation, set

$$Y = X_{\pi(j)} \in \{0, 1\}, \quad Z = \Phi((X_{\pi(1)}, \dots, X_{\pi(j-1)})) \in \{0, 1\}^{2^{j-1}}, \quad (24)$$

815 and let $w \in \mathbb{R}^{2^{j-1}}$ be the parameter vector. The population objective is

$$\ell(w) = \mathbb{E}[\log(1 + \exp(w^\top Z)) - Y(w^\top Z)]. \quad (25)$$

816 A straightforward calculation shows

$$\nabla^2 \ell(w) = \mathbb{E}[\sigma(w^\top Z)(1 - \sigma(w^\top Z)) Z Z^\top], \quad \sigma(c) = \frac{1}{1 + e^{-c}}. \quad (26)$$

817 Since $Z \in \{0, 1\}^{2^{j-1}}$, every inner product $w^\top Z$ is finite and thus $\sigma(w^\top Z)(1 - \sigma(w^\top Z)) > 0$.

818 Define

$$m(w) = \min_{z \in \{0, 1\}^{2^{j-1}}} \sigma(w^\top z)(1 - \sigma(w^\top z)) > 0. \quad (27)$$

819 The marginal distribution of $(X_{\pi(1)}, \dots, X_{\pi(j-1)})$ is still a multivariate Bernoulli distribution, with
820 natural parameters $\mathbf{p}_{\pi, j-1}$. i.e., $(X_{\pi(1)}, \dots, X_{\pi(j-1)}) \sim \text{MultiBernoulli}(\mathbf{p}_{\pi, j-1})$. By lemma 1,

821 $q_{\min, j-1} = \min_{i \in [2^{j-1}]} [\mathbf{p}_{\pi, j-1}]_i > 0$. Hence for any nonzero $\beta \in \mathbb{R}^{2^{j-1}}$,

$$\begin{aligned} \beta^\top \nabla^2 \ell(w) \beta &= \sum_{z \in \{0, 1\}^{2^{j-1}}} p(z) \sigma(w^\top z)(1 - \sigma(w^\top z)) (\beta^\top z)^2 \\ &\geq q_{\min, j-1} m(w) \sum_{z \in \{0, 1\}^{2^{j-1}}} (\beta^\top z)^2 > 0, \end{aligned} \quad (28)$$

822 because the sum is over all $z \in \{0, 1\}^{2^{j-1}}$. So $(\beta^\top z)^2 > 0$ for at least one z when $\beta \neq 0$.
823 Since the Hessian is everywhere positive definite, $\ell(w)$ is strictly convex and therefore has a unique
824 minimizer.

825 **B.4 Proof of Theorem 1**

826 Let π be any permutation of $\{1, \dots, p\}$. Then the joint distribution of X can be written as

$$P(X) = \prod_{j=1}^p P(X_{\pi(j)} \mid X_{\pi(1), \dots, \pi(j-1)}) \quad (29)$$

827 In the population limit ($n \rightarrow \infty$), each conditional law has the Bernoulli form

$$P(X_{\pi(j)} \mid X_{\pi(1)}, \dots, X_{\pi(j-1)}) = q^{X_{\pi(j)}} (1 - q)^{1 - X_{\pi(j)}} \quad (30)$$

828 where

$$q = \text{logistic} \left(\mathbf{f}_{\pi, j}^\top \Phi((X_{\pi(1)}, \dots, X_{\pi(j-1)})) \right) \quad (31)$$

829 Lemma 3 guarantees that logistic regression uniquely recovers $\mathbf{f}_{\pi, j}$. Moreover, the structural equation
830 model

$$Y_{\pi(j)} \sim \text{Bernoulli} \left(\text{logistic} \left(\mathbf{f}_{\pi, j}^\top \Phi((Y_{\pi(1)}, \dots, Y_{\pi(j-1)})) \right) \right) \quad (32)$$

831 It induces exactly the same conditionals as X . i.e.,

$$Y_{\pi(j)} \mid Y_{\pi(1)}, \dots, Y_{\pi(j-1)} \stackrel{d}{=} X_{\pi(j)} \mid X_{\pi(1)}, \dots, X_{\pi(j-1)}, \quad (33)$$

832 Hence

$$Y \sim \text{MultiBernoulli}(\mathbf{p}), \quad (34)$$

833 establishing the desired result.

834 **B.5 Proof of Theorem 2**

835 By Theorem 1, for every $(\mathbf{f}_\pi, G_\pi) \in \mathcal{E}_{\min}(\mathbf{p})$, the vector \mathbf{f}_π defines, via the structural equation
836 model (7), a distribution

$$X \sim \text{MultiBernoulli}(\mathbf{p}) \quad (35)$$

837 that is Markov with respect to G_π . By definition of $\mathcal{E}_{\min}(\mathbf{p})$, each G_π is a sparsest graph in the
838 equivalence class $\mathcal{E}(\mathbf{p})$. Since all sparsest Markov representations lie in the same Markov equivalence
839 class, it follows that for any two pairs

$$(\mathbf{f}_{\pi_1}, G_{\pi_1}), (\mathbf{f}_{\pi_2}, G_{\pi_2}) \in \mathcal{E}_{\min}(\mathbf{p}), \quad (36)$$

840 their Markov classes coincide:

$$\mathcal{M}(G_{\pi_1}) = \mathcal{M}(G_{\pi_2}). \quad (37)$$

841 **B.6 Proof of Theorem 3**

842 The proof relies on Theorems 4 and 5 of Deng et al. [15]. We verify Assumptions A and B from that
843 work.

844 **Assumption A (1)** This requires the equivalence class to be finite. Since

$$|\mathcal{E}_{\min}(\mathbf{p})| \leq p! \quad (38)$$

845 the condition holds.

846 **Assumption A (2)** This requires the weighted adjacency matrix $W(\mathbf{H})$ to be L -Lipschitz. Recall
 847 that in (10),

$$[W(\mathbf{H})]_{ij} = \sum_{S \subseteq [p], i \in S} (h^{j,S})^2. \quad (39)$$

848 For simplicity we instead use the equivalent form

$$[W(\mathbf{H})]_{ij} = \sum_{S \subseteq [p], i \in S} |h^{j,S}|. \quad (40)$$

849 Let \mathbf{H}_1 and \mathbf{H}_2 be two parameter values. We show there exists L such that

$$\|W(\mathbf{H}_1) - W(\mathbf{H}_2)\|_2 \leq L \|\mathbf{H}_1 - \mathbf{H}_2\|_2. \quad (41)$$

850 First,

$$\|W(\mathbf{H}_1) - W(\mathbf{H}_2)\|_2 = \sqrt{\sum_j \sum_i \left(\sum_{S \subseteq [p], i \in S} |h_1^{j,S}| - \sum_{S \subseteq [p], i \in S} |h_2^{j,S}| \right)^2}. \quad (42)$$

851 Meanwhile,

$$\|\mathbf{H}_1 - \mathbf{H}_2\|_2 = \sqrt{\sum_j \sum_{S \subseteq [p]} (h_1^{j,S} - h_2^{j,S})^2}. \quad (43)$$

852 By Cauchy–Schwarz,

$$\begin{aligned} \left(\sum_{S \subseteq [p], i \in S} |h_1^{j,S}| - \sum_{S \subseteq [p], i \in S} |h_2^{j,S}| \right)^2 &= \left(\sum_{S \subseteq [p], i \in S} (|h_1^{j,S}| - |h_2^{j,S}|) \right)^2 \\ &\leq |S \subseteq [p], i \in S| \sum_{S \subseteq [p], i \in S} (|h_1^{j,S}| - |h_2^{j,S}|)^2 \\ &\leq 2^{p-1} \sum_{S \subseteq [p], i \in S} (h_1^{j,S} - h_2^{j,S})^2 \\ &\leq 2^{p-1} \sum_{S \subseteq [p]} (h_1^{j,S} - h_2^{j,S})^2 \end{aligned} \quad (44)$$

853 Hence

$$\begin{aligned} \|W(\mathbf{H}_1) - W(\mathbf{H}_2)\|_2 &= \sqrt{\sum_j \sum_i \left(\sum_{S \subseteq [p], i \in S} |h_1^{j,S}| - \sum_{S \subseteq [p], i \in S} |h_2^{j,S}| \right)^2} \\ &\leq \sqrt{\sum_j \sum_i 2^{p-1} \sum_{S \subseteq [p]} (h_1^{j,S} - h_2^{j,S})^2} \\ &\leq \sqrt{2^{p-1} p \sum_j \sum_{S \subseteq [p]} (h_1^{j,S} - h_2^{j,S})^2} \\ &= \sqrt{p} 2^{(p-1)/2} \sqrt{\sum_j \sum_{S \subseteq [p]} (h_1^{j,S} - h_2^{j,S})^2} \\ &= \sqrt{p} 2^{(p-1)/2} \|\mathbf{H}_1 - \mathbf{H}_2\|_2 \end{aligned} \quad (45)$$

854 Thus one may take $L = \sqrt{p} 2^{(p-1)/2}$.

855 **Assumption B** This requires $\mathbb{E}[s(\mathbf{H}; \mathbf{X})]$ to have bounded level sets. From Lemma 3 we know that
 856 under $\mathbf{p} > 0$, each population logistic loss is strongly convex. Since $\mathbb{E}[s(\mathbf{H}; \mathbf{X})]$ is a sum of p such
 857 strongly convex terms, it is itself strongly convex. By Corollary 2, any strongly convex function has
 858 bounded sublevel sets.

859 B.7 Proof of Theorem 4

860 Let π^* be the topological ordering of G guaranteed by Assumption A. Then

$$P(X) = \prod_{j=1}^p P(X_{\pi^*(j)} \mid X_{\pi^*(1), \dots, \pi^*(j-1)}) \quad (46)$$

861 Write $S_{\pi, j} = \{\pi^*(1), \dots, \pi^*(j-1)\}$. Under the linear SEM assumption,

$$X_{\pi^*(j)} \mid X_{S_{\pi, j}} = x_{S_{\pi, j}} \sim \text{Bernoulli}(\text{logistic}([w_{\pi^*(j)}]_{S_{\pi, j}}^\top X_{S_{\pi, j}} + c_{\pi^*(j)})) \quad (47)$$

862 where $w_{\pi^*(j)}$ and $c_{\pi^*(j)}$ are finite. Hence for any $x_{S_{\pi, j}}$,

$$0 < \text{logistic}(w_{\pi^*(j), S_{\pi, j}}^\top x_{S_{\pi, j}} + c_{\pi^*(j)}) < 1, \quad (48)$$

863 so both

$$P(X_{\pi^*(j)} = 1 \mid X_{S_{\pi, j}} = x_{S_{\pi, j}}) \quad \text{and} \quad P(X_{\pi^*(j)} = 0 \mid X_{S_{\pi, j}} = x_{S_{\pi, j}}) \quad (49)$$

864 lie in $(0, 1)$, implying $\mathbf{p} > 0$.

865 Although every $(\mathbf{f}_\pi, G_\pi) \in \mathcal{E}(\mathbf{p})$ generates $X \sim \text{MultiBernoulli}(\mathbf{p})$ via (7), the linear optimization
 866 (94) only admits first-order models. Thus any (\mathbf{f}_π, G_π) with higher-order terms is misspecified.
 867 Define

$$\mathcal{E}^{\text{linear}}(\mathbf{p}) = \{(\mathbf{f}_\pi, G_\pi) : (\mathbf{f}_\pi, G_\pi) \text{ where } \mathbf{f}_\pi \text{ only has first order term, } \forall \pi\} \quad (50)$$

868 By Assumption A, $(W^0, G^0) \in \mathcal{E}^{\text{linear}}(\mathbf{p})$. Let

$$\mathcal{E}_{\min}^{\text{linear}}(\mathbf{p}) = \{(\mathbf{f}_\pi, G_\pi) : (\mathbf{f}_\pi, G_\pi) \text{ is the minimal element, } (\mathbf{f}_\pi, G_\pi) \in \mathcal{E}^{\text{linear}}(\mathbf{p})\} \quad (51)$$

869 If (W^0, G^0) is not already minimal, we simply replace it by the sparsest element in $\mathcal{E}^{\text{linear}}(\mathbf{p})$, so
 870 that $(W^0, G^0) \in \mathcal{E}_{\min}^{\text{linear}}(\mathbf{p})$.

871 In the proof of Theorem 3, we verified that our model meets Assumptions A and B of Deng et al.
 872 [15]. Therefore, by Theorem 4 of Deng et al. [15], we have

$$\mathcal{O}_{\infty, \lambda, \delta}^{\text{linear}} = \mathcal{E}_{\min}^{\text{linear}}(\mathbf{p}), \quad \text{and hence} \quad (W^0, G^0) \in \mathcal{O}_{\infty, \lambda, \delta}^{\text{linear}}. \quad (52)$$

873 B.8 Proof of Corollary 1

874 Since $X \sim \text{MultiBernoulli}(\mathbf{p})$ and $X = (X_1, \dots, X_p)$. Then, $\forall S \subseteq [p]$, the range of $X_S \in$
 875 $\{0, 1\}^{|S|}$. The corresponding density mass

$$P(X_S = x_S) = \sum_{x_{[p] \setminus S} \in \{0, 1\}^{p-|S|}} P(X_S = x_S, X_{[p] \setminus S} = x_{[p] \setminus S}) \quad (53)$$

876 In such way, we could enumerate all the value for the $x_S \in \{0, 1\}^{|S|}$, and put them together to get the
 877 natural parameter for X_S , i.e., \mathbf{p}_S . As a consequence, $X_S \sim \text{MultiBernoulli}(\mathbf{p}_S)$.

878 For any $j \in [p]$, and any $S \in [p] \setminus j$, since $X_j \in \{0, 1\}$, so the conditional distribution $P(X_j \mid X_S)$ is
 879 Bernoulli distribution, and the probability

$$\begin{aligned} P(X_j = x_j \mid X_S = x_S) &= \frac{P(X_j = x_j, X_S = x_S)}{P(X_S = x_S)} \\ &= \frac{P(X_j = x_j, X_S = x_S)}{P(X_j = 1, X_S = x_S) + P(X_j = 0, X_S = x_S)} \end{aligned} \quad (54)$$

880 B.9 Proof of Corollary 2

881 Since f is strongly convex,

$$f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \frac{m}{2} \|y - x\|^2 \quad \forall x, y \in \mathbb{R}^d \quad (55)$$

882 Take $x = x^*$, then

$$f(y) \geq f(x^*) + \frac{m}{2} \|y - x^*\|^2 \quad \forall x, y \in \mathbb{R}^d \quad (56)$$

883 Rearranging,

$$f(y) \leq c \Rightarrow \frac{m}{2} \|y - x^*\|^2 \leq c - f(x^*) \Rightarrow \|y - x^*\| \leq \sqrt{\frac{2(c - f(x^*))}{m}} \quad (57)$$

884 Thus, $L_c \subseteq \left\{ y : \|y - x^*\| \leq \sqrt{\frac{2(c - f(x^*))}{m}} \right\}$, a bounded ball.

885 C Supplementary Technical Details and Examples

886 In this appendix, we collect additional technical derivations, algorithms, and definitions that support
887 our main theorems.

- 888 • In C.1, we give the explicit parameter transformation between the general parameter \mathbf{p} and
889 the natural parameter \mathbf{f} of the multivariate Bernoulli model [11];
- 890 • In C.2, we derive the logistic form of the conditional distributions in equation (3)
- 891 • In C.3, we formalize the graded-lexicographic ordering used to index the interaction features.
- 892 • In C.4, we present the population-level recovery algorithms for each topological order π .
- 893 • In C.5, we define the structural equation model framework underlying Theorem 1.
- 894 • In C.6, we review faithfulness, Markov equivalence, and the Sparsest Markov Representation
895 necessary for Theorems 2 and 3
- 896 • In C.7, we provide the derivation of our score function $s(\mathbf{H}; \mathbf{X})$ (negative log-likelihood
897 function).
- 898 • In C.8, we provide theoretical justification for the previous work [12, 4] with our general
899 framework.

900 C.1 Parameter Transformation in the Multivariate Bernoulli Distribution

901 All the material in this subsection can be found in [11]. We include here for the completeness.

902 The multivariate Bernoulli distribution has two different parameterization, one is using general
903 parameter \mathbf{p} and another one is using natural parameter \mathbf{f} .

904 The density is expressed by general parameter \mathbf{p} .

$$\begin{aligned} P(Y_1 = y_1, Y_2 = y_2, \dots, Y_K = y_K) &= p(y_1, y_2, \dots, y_K) \\ &= p(0, 0, \dots, 0)^{\prod_{j=1}^K (1 - y_j)} \\ &\quad \times p(1, 0, \dots, 0)^{[y_1 \prod_{j=2}^K (1 - y_j)]} \\ &\quad \times p(0, 1, \dots, 0)^{[(1 - y_1) y_2 \prod_{j=3}^K (1 - y_j)]} \dots \\ &\quad \times p(1, 1, \dots, 1)^{\prod_{j=1}^K y_j}, \end{aligned} \quad (58)$$

905 The density is expressed by natural parameter \mathbf{f} .

$$P(Y_1 = y_1, Y_2 = y_2, \dots, Y_K = y_K) = \exp \left(f^0 + \sum_{r=1}^p \left(\sum_{1 \leq j_1 < j_2 < \dots < j_r \leq p} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}(y) \right) \right) \quad (59)$$

906 To simplify the notation, we could define the quantity S to be

$$S^{j_1 j_2 \dots j_r} = \sum_{1 \leq s \leq r} f^{j_s} + \sum_{1 \leq s < t \leq r} f^{j_s j_t} + \dots + f^{j_1 j_2 \dots j_r} \quad (60)$$

907 and also define the interaction function B

$$B^{j_1 j_2 \dots j_r}(y) = y_{j_1} y_{j_2} \dots y_{j_r} \quad (61)$$

908 The following lemma shows the one-to-one mapping between general parameter \mathbf{p} and natural
909 parameter \mathbf{f} .

910 **Lemma 4** (Parameter transformation). *For the multivariate Bernoulli model, the general parameters
911 and natural parameters have the following relationship.*

$$\exp(f^{j_1 j_2 \dots j_r}) \quad (62)$$

$$= \frac{\prod p(\text{even \# zeros among } j_1, j_2, \dots, j_r \text{ components and other components are all zero})}{\prod p(\text{odd \# zeros among } j_1, j_2, \dots, j_r \text{ components and other components are all zero})}, \quad (63)$$

912 where $\#$ refers to the number of zeros among the superscript $y_{j_1} \dots y_{j_r}$ of f . In addition,

$$\exp(S^{j_1 j_2 \dots j_r}) = \frac{p(j_1, j_2, \dots, j_r \text{ positions are one, others are zero})}{p(0, 0, \dots, 0)} \quad (64)$$

913 and conversely the general parameters can be represented by the natural parameters

$$p(j_1, j_2, \dots, j_r \text{ positions are one, others are zero}) = \frac{\exp(S^{j_1 j_2 \dots j_r})}{\exp(b(\mathbf{f}))}. \quad (65)$$

914 where

$$b(\mathbf{f}) = \log \sum_{r=1}^K \left[1 + \left(\sum_{1 \leq j_1 < j_2 < \dots < j_r \leq K} \exp[S^{j_1 j_2 \dots j_r}] \right) \right] \quad (66)$$

915 C.2 Conditional distribution of multivariate Bernoulli distribution

916 In this part, we derive the conditional distribution of multivariate Bernoulli distribution. Especially,

$$\begin{aligned} & P(X_p = 1 \mid X_1 = x_1, \dots, X_{p-1} = x_{p-1}) \\ &= \text{logistic} \left(\sum_{r=1}^p \left(\sum_{1 \leq j_1 < j_2 < \dots < j_r = p \leq p} f^{j_1 \dots j_{r-1} p} x_{j_1} \dots x_{j_{r-1}} x_p \right) \right) \\ &= \text{logistic} (f^p + f^{1p} x_1 + f^{2p} x_2 \dots f^{p-1, p} f_{p-1} + f^{12p} x_1 x_2 + \dots + f^{1 \dots p} x_1 \dots x_{p-1}) \end{aligned} \quad (67)$$

917 It is known the multivariate Bernoulli distribution in exponential form can be written as

$$P(X_1 = x_1, \dots, X_p = x_p) = \exp \left(f^0 + \sum_{r=1}^p \left(\sum_{1 \leq j_1 < j_2 < \dots < j_r \leq p} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}(x) \right) \right) \quad (68)$$

918 Then,

$$\begin{aligned} P(X_p = 1 \mid X_{-p} = x_{-p}) &= \frac{P(X_p = 1, X_{-p} = x_{-p})}{P(X_{-p} = x_{-p})} \\ &= \frac{P(X_p = 1, X_{-p} = x_{-p})}{P(X_p = 1, X_{-p} = x_{-p}) + P(X_p = 0, X_{-p} = x_{-p})} \\ &= \frac{1}{1 + \frac{P(X_p = 0, X_{-p} = x_{-p})}{P(X_p = 1, X_{-p} = x_{-p})}} \end{aligned} \quad (69)$$

919 where $X_{-p} = (X_1, \dots, X_{p-1})$.

$$\begin{aligned}
& P(X_1 = x_1, \dots, X_p = x_p) \\
&= \exp \left(f^0 + \sum_{r=1}^p \left(\sum_{1 \leq j_1 < j_2 < \dots < j_r \leq p} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}(x) \right) \right) \\
&= \exp \left(f^0 + \sum_{r=1}^p \left(\sum_{\substack{1 \leq j_1 < j_2 < \dots < j_r \leq p \\ p \in \{j_1, \dots, j_r\}}} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}(x) + \sum_{\substack{1 \leq j_1 < j_2 < \dots < j_r \leq p \\ p \notin \{j_1, \dots, j_r\}}} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}(x) \right) \right) \tag{70}
\end{aligned}$$

920 Then,

$$\begin{aligned}
& P(X_1 = x_1, \dots, X_p = 1) \\
&= \exp \left(f^0 + \sum_{r=1}^p \left(\sum_{\substack{1 \leq j_1 < j_2 < \dots < j_r \leq p \\ p \in \{j_1, \dots, j_r\}}} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}((x_{-p}, 1)) + \sum_{\substack{1 \leq j_1 < j_2 < \dots < j_r \leq p \\ p \notin \{j_1, \dots, j_r\}}} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}(x) \right) \right) \tag{71}
\end{aligned}$$

921

$$P(X_1 = x_1, \dots, X_p = 0) = \exp \left(f^0 + \sum_{r=1}^p \left(\sum_{\substack{1 \leq j_1 < j_2 < \dots < j_r \leq p \\ p \notin \{j_1, \dots, j_r\}}} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}(x) \right) \right) \tag{72}$$

922 Finally, put them together,

$$\begin{aligned}
& P(X_p = 1 \mid X_{-p} = x_{-p}) \\
&= \frac{1}{1 + \frac{P(X_p=0, X_{-p}=x_{-p})}{P(X_p=1, X_{-p}=x_{-p})}} \\
&= \frac{1}{1 + \exp \left(- \sum_{r=1}^p \left(\sum_{\substack{1 \leq j_1 < j_2 < \dots < j_r \leq p \\ p \in \{j_1, \dots, j_r\}}} f^{j_1 j_2 \dots j_r} B^{j_1 j_2 \dots j_r}((x_{-p}, 1)) \right) \right)} \\
&= \sigma \left(f^p + f^{1p} x_1 + f^{2p} x_2 \dots f^{p-1,p} f_{p-1} + f^{12p} x_1 x_2 + \dots + f^{1 \dots p} x_1 \dots x_{p-1} \right) \tag{73}
\end{aligned}$$

923 where $\sigma(x) = \frac{1}{1 + \exp(-x)}$

924 C.3 Graded-lexicographic order

925 To index the 2^p interaction features and corresponding parameters in a consistent way, we use the
926 *graded-lexicographic* order on subsets of $[p]$, where for any finite set S , $|S|$ denotes its cardinality
927 (the number of elements in S).

928 **Definition 4** (Graded-lexicographic order). *Let π be a permutation of $[p]$. For any two subsets*
929 *$S, T \subseteq [p]$, we say $S \prec_{\text{grlex}} T$ if either*

- 930 1. $|S| < |T|$, or
- 931 2. $|S| = |T|$ and, when listing the elements of S and T in ascending order under π , the first
932 index at which they differ belongs to S .

933 Under this rule, all subsets are grouped by increasing cardinality, and ties are broken by the usual lex
934 order induced by π .

935 **Example.** Take $p = 3$ and the identity order $\pi = (1, 2, 3)$. Then the graded-lexicographic sequence
936 of subsets is

$$\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}. \tag{74}$$

Algorithm 1: RECOVERPARENTS(\mathbf{p}, π, j)

```
1:  $\text{PA}(\pi(j)) \leftarrow \{\}$ , and  $X_{\pi,j} \leftarrow (X_{\pi(1)}, \dots, X_{\pi(j)})$ 
2: Compute the general parameters  $\mathbf{p}_{\pi,j}$  of  $X_{\pi,j}$  // Compute the marginal distribution of  $X_{\pi,j}$ 
3: Convert  $\mathbf{p}_{\pi,j}$  to natural parameters  $\mathbf{f}_{\pi,j}$  utilizing Lemma 4
4: for  $i = 1, \dots, j - 1$  do
    | if  $\sum_{S \subseteq [\pi(1), \dots, \pi(j-1), \pi(j)] \setminus [\pi(i), \pi(j)]} (f_{\pi,j}^{\pi(i), \pi(j), S})^2 > 0$  then
    | |  $\text{PA}(\pi(j)) \leftarrow \text{PA} \cup \{\pi(i)\}$ 
5: return  $(\text{PA}(\pi(j)), \mathbf{f}_{\pi,j})$ 
```

Algorithm 2: RECOVERDAG(\mathbf{p}, π)

Input: Probability vector \mathbf{p} (or empirical count) and topological sort π

Output: DAG G_π , and natural parameters \mathbf{f}_π

```
1  $G_\pi \leftarrow$  empty graph and  $\mathbf{f}_\pi \leftarrow \{\}$ 
2 for  $j = 1, 2, \dots, p$  do
3    $(\text{PA}(\pi(j)), \mathbf{f}_{\pi,j}) \leftarrow \text{RECOVERPARENTS}(\mathbf{p}, \pi, j)$  //  $\text{PA}(\pi(j))$ : the parents of node  $\pi(j)$ 
4   for  $i \in \text{PA}(\pi(j))$  do
5     | Add edge  $X_i \rightarrow X_j$  to  $G_\pi$ 
6    $\mathbf{f}_\pi \leftarrow \mathbf{f}_\pi \cup \{\mathbf{f}_{\pi,j}\}$ 
```

937 Accordingly, the extended feature map $\Phi(X) = [B^S(X)]_{S \subseteq [3]}$ becomes

$$\Phi(X) = [1, X_1, X_2, X_3, X_1X_2, X_1X_3, X_2X_3, X_1X_2X_3]. \quad (75)$$

938 Similarly, for any j and order π , the parameter block $\mathbf{f}_{\pi,j} \in \mathbb{R}^{2^{j-1}}$ is arranged so that its entries
939 align one-to-one with $\Phi(X_{\pi(1)}, \dots, X_{\pi(j-1)})$ in graded-lexicographic order.

940 C.4 Procedure for recovering causal graph and parameters

941 To formalize the recovery procedure from Section 4.2, we present Algorithms 1 and 2.

942 **Algorithm 1: Parent and parameter recovery.** For a fixed topological order π and node index
943 $j \in [p]$, this algorithm

- 944 1. computes the marginal probabilities $\mathbf{p}_{\pi,j}$ of $(X_{\pi(1)}, \dots, X_{\pi(j)})$,
- 945 2. converts $\mathbf{p}_{\pi,j}$ to the natural-parameter block $\mathbf{f}_{\pi,j}$ via Lemma 4, and
- 946 3. selects the parent set $\text{PA}_\pi(j)$ using the nonzero-coefficient criterion in (4).

947 For estimating the natural-parameter block $\mathbf{f}_{\pi,j}$, Section 4.2 uses a logistic regression approach [21].
948 In Algorithm 1, we instead compute $\mathbf{f}_{\pi,j}$ by applying the mapping of Lemma 4 to the marginal
949 probabilities. Under the positivity assumption $\mathbf{p} > 0$, these two methods are equivalent and yield
950 identical estimates for $\mathbf{f}_{\pi,j}$.

951 **Algorithm 2: Equivalence-class enumeration.** This algorithm iterates over all $p!$ permutations
952 $\pi \in \mathfrak{S}_p$ and, for each, calls Algorithm 1 for every $j = 1, \dots, p$. It assembles the corresponding DAG
953 G_π and parameter collection $\mathbf{f}_\pi = \{\mathbf{f}_{\pi,j}\}_{j=1}^p$. The output is the full equivalence class

$$\mathcal{E}(\mathbf{p}) = \{(\mathbf{f}_\pi, G_\pi) : (\mathbf{f}_\pi, G_\pi) \text{ is returned for some } \pi\}.$$

954 Although we state the algorithm in terms of the population vector \mathbf{p} , in practice one can simply input
955 the data matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, since empirical frequencies convert \mathbf{X} into \mathbf{p} .

956 Building on Algorithms 1 and 2, Algorithm 3 then enumerates all graph-parameter pairs in $\mathcal{E}(\mathbf{p})$ and
957 retains only those with the fewest edges, thereby recovering the minimal equivalence class $\mathcal{E}_{\min}(\mathbf{p})$
958 as defined in (8).

Algorithm 3: RECOVERSPARESTDAG(p)

Input: Probability vector p (empirical count)

```
1  $\mathcal{S} \leftarrow \{\}$ 
2 for each  $\pi \in \mathfrak{G}_p$  do
3    $(G_\pi, \mathbf{f}_\pi) \leftarrow \text{RECOVERGRAPH}(p, \pi) // \mathfrak{G}_p$ : set of all the permutation on  $p$  variables
4    $\mathcal{S} \leftarrow \mathcal{S} \cup \{(G_\pi, \mathbf{f}_\pi)\}$ 
5 return  $((G_\pi, \mathbf{f}_\pi) \in \mathcal{S} : s_{G_\pi} \leq s_{G_{\tilde{\pi}}}, \forall (G_{\tilde{\pi}}, \mathbf{f}_{\tilde{\pi}}) \in \mathcal{S})$ 
```

959 C.5 Structural equation model

960 An structural equation model (SEM) [32] $(X, f, P(N))$ over the random vector $X = (X_1, \dots, X_p)$
961 is a collection of p structural equations of the form:

$$X_j = f_j(X, N_j), \quad \partial_k f_j = 0 \text{ if } k \notin \text{PA}(j), \quad (76)$$

962 where $f = (f_j)_{j=1}^p$ is a collection of functions $f_j : \mathbb{R}^{p+1} \rightarrow \mathbb{R}$, here $N = (N_1, \dots, N_p)$ is a vector
963 of independent noises with distribution $P(N)$, and $\text{PA}(j)$ denotes the set of parents of node j . Here,
964 $\partial_k f_j$ denotes the partial derivative of f_j w.r.t. X_k , which is identically zero when f_j is independent
965 of X_k , i.e. $f_j(X, N_j) = f_j(X_{\text{PA}(j)}, N_j)$. The graphical structure induced by the SEM, assumed to
966 be a DAG, will be represented by the following $p \times p$ weighted adjacency matrix B :

$$B = B(f), \quad B_{ij} = \|\partial_i f_j\|_2, \quad (77)$$

967 and we use $G(B)$ to denote the corresponding binary adjacency matrix.

968 The structural-equation framework in (76) provides a fully generative foundation for causal discovery
969 by specifying, for each variable, an explicit functional dependence on its direct causes. Its generality
970 encompasses a wide range of models—linear SEMs, additive-noise models, post-nonlinear models,
971 generalized linear models, and more expressive non-linear SEMs. A major virtue of SEMs is their
972 universality: *any* joint distribution can be represented in the form (76) (Proposition 7.1 of 32). In
973 practice, however, researchers impose strong parametric restrictions—such as linearity, additivity, or
974 low-order interactions—that may fail to capture the full complexity of real-world data.

975 C.6 Faithfulness, sparsest Markov representation, Markov equivalence class

976 We formally define the concepts mentioned in Section 5.

977 **Definition 5** (Faithfulness [32]). *A pair (G, P) is said to be faithful if:*

$$X_i \perp\!\!\!\perp X_j \mid X_K \iff X_i \text{ and } X_j \text{ are } d\text{-separated by } X_K \text{ in } G, \quad (78)$$

978 *for all disjoint subsets $\{i, j\}, K \subseteq V$. That is, every conditional independence in P corresponds*
979 *exactly to a d -separation in G , and vice versa.*

980 **Definition 6** (Markov Equivalence Class [38]). *Two DAGs G_1 and G_2 on the same vertex set V are*
981 *Markov equivalent if they encode the same set of conditional independence relations—equivalently,*
982 *they have the same skeleton (undirected edges) and the same set of v -structures (induced subgraphs*
983 *of the form $i \rightarrow k \leftarrow j$ with i and j not adjacent). The Markov equivalence class of a DAG G is*

$$\mathcal{M}(G) = \{G' : G' \text{ is a DAG and } G' \text{ is Markov equivalent to } G\}. \quad (79)$$

984 **Definition 7** (Sparsest Markov Representation [35]). *A pair (G^0, P) satisfies the Sparsest Markov*
985 *Representation (SMR) assumption if:*

986 1. (G^0, P) satisfies the Markov property, i.e. every d -separation in G^0 implies the correspond-
987 ing conditional independence in P .

988 2. For any other DAG $G \notin \mathcal{M}(G^0)$ satisfying the Markov property with respect to P , we have

$$|E(G)| > |E(G^0)|.$$

989 *Equivalently, G^0 is the (unique up to Markov equivalence) sparsest DAG compatible with P .*

990 C.7 Derivation of the Logistic Loss

991 First, consider a single example (X, y) with feature vector $X \in \mathbb{R}^m$, binary label $y \in \{0, 1\}$, and
 992 parameter vector $w \in \mathbb{R}^m$. Let

$$q = \text{logistic}(w^\top X) = \frac{1}{1 + \exp(-w^\top X)}. \quad (80)$$

993 The (negative) log-likelihood is

$$\begin{aligned} \ell(w; X, y) &= -\log(q^y(1-q)^{1-y}) \\ &= -y \log q - (1-y) \log(1-q). \end{aligned} \quad (81)$$

994 Noting that

$$\log \frac{q}{1-q} = w^\top X, \quad \log(1-q) = -\log(1 + \exp(w^\top X)), \quad (82)$$

995 we obtain the familiar logistic-loss form:

$$\ell(w; X, y) = \log(1 + \exp(w^\top X)) - y(w^\top X). \quad (83)$$

996 In our setting, each “feature” is replaced by the *extended* feature matrix $\Phi(\mathbf{X}) \in \mathbb{R}^{n \times 2^p}$, and each
 997 “label” is one column of the data matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$. Stacking over all p columns and averaging over
 998 n samples yields

$$\begin{aligned} \ell(\mathbf{H}; \mathbf{X}) &= \frac{1}{n} \sum_{j=1}^p \sum_{i=1}^n \left[\log(1 + \exp[\Phi(\mathbf{X})\mathbf{H}]_{ij}) - \mathbf{X}_{ij} [\Phi(\mathbf{X})\mathbf{H}]_{ij} \right] \\ &= \frac{1}{n} \sum_{j=1}^p \mathbf{1}_n^\top \left(\log(\mathbf{1}_n + \exp(\Phi(\mathbf{X})\mathbf{H})) - \mathbf{X}_j \circ (\Phi(\mathbf{X})\mathbf{H}) \right), \end{aligned} \quad (84)$$

999 where \circ denotes the Hadamard product and $\mathbf{1}_n \in \mathbb{R}^n$ is the all-ones vector.

1000 C.8 Theoretical justification for previous works

1001 Under Assumption A, there exists a topological ordering π consistent with G such that each

$$X_{\pi(j)} \text{ is generated by a linear combination of } (X_{\pi(1)}, \dots, X_{\pi(j-1)}) \in \mathbb{R}^{j-1} \text{ via the logistic link,} \quad (85)$$

1002 rather than via the logistic link on $\Phi((X_{\pi(1)}, \dots, X_{\pi(j-1)})) \in \mathbb{R}^{2^{j-1}}$. Importantly, Algorithms 1
 1003 and 2 remain valid under this assumption. For any other topological sort $\tilde{\pi} \neq \pi$, the output $(G_{\tilde{\pi}}, f_{\tilde{\pi}})$
 1004 from Algorithm 2 may include higher-order interaction terms; nevertheless, by the structural equation
 1005 model (7), it still recovers the exact distribution

$$X \sim \text{MultiBernoulli}(\mathbf{p}), \quad (86)$$

1006 so Theorem 1 continues to hold. Moreover, under Assumption A we can reduce the dimensionality of
 1007 the optimization (13) from $\mathbf{H} \in \mathbb{R}^{2^p \times p}$ to $\mathbf{H} \in \mathbb{R}^{(p+1) \times p}$. We formalize this reduction below.

1008 Define the parameter matrix

$$\mathbf{H}_j = (\underbrace{h^{j,0}}_{\text{constant}}, \underbrace{h^{j,1}, \dots, h^{j,p}}_{\text{first order}})^\top \in \mathbb{R}^{p+1} \quad \mathbf{H} = (\mathbf{H}_1, \dots, \mathbf{H}_p) \in \mathbb{R}^{(p+1) \times p} \quad (87)$$

1009 where $h^{j,0}$ is the intercept and $h^{j,1}, \dots, h^{j,p}$ are the first-order coefficients.

1010 The induced adjacency matrix is

$$[W(\mathbf{H})]_{ij} = |h^{j,i}| \quad (88)$$

1011 Self-loops are forbidden, so we impose

$$h^{j,j} = 0 \quad \forall j \in [p] \quad (89)$$

1012 Redefine the feature map row-wise as

$$\Phi(X) = [1, X_1, \dots, X_p], \quad (90)$$

1013 so that for a data matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, $\Phi(\mathbf{X})$ applies Φ to each row.

1014 The score (negative log-likelihood) remains

$$\ell(\mathbf{H}; \mathbf{X}) = \frac{1}{n} \sum_{i=1}^p \mathbf{1}_n^\top (\log(\mathbf{1}_n + \exp(\Phi(\mathbf{X})\mathbf{H})) - \mathbf{X}_i \circ (\Phi(\mathbf{X})\mathbf{H})) \quad (91)$$

1015 We continue to use the quasi-MCP penalty [15], defined by

$$\text{quasi-MCP: } p_{\lambda, \delta}(t) = \lambda \left[\left(|t| - \frac{t^2}{2\delta} \right) \mathbb{1}(|t| < \delta) + \frac{\delta}{2} \mathbb{1}(|t| > \delta) \right] \quad (92)$$

1016 Our final score function is as below

$$s(\mathbf{H}; \lambda, \delta, \mathbf{X}) = s(\mathbf{H}; \mathbf{X}) + p_{\lambda, \delta}(W(\mathbf{H})) \quad (93)$$

1017 We formulate this task as the single continuous optimization problem

$$\begin{aligned} \min_{\mathbf{H}} \quad & s(\mathbf{H}; \lambda, \delta, \mathbf{X}) \\ \text{subject to} \quad & h(W(\mathbf{H})) = 0 \\ & h^{j,j} = 0 \quad \forall j \in [p] \end{aligned} \quad (94)$$

1018 Define the global optimal solution of (94) as

$$\mathcal{O}_{n, \lambda, \delta}^{\text{linear}} = \{(\mathbf{H}^*, G(W(\mathbf{H}^*))) : \mathbf{H}^* \text{ is a minimizer of (94)}\} \quad (95)$$

1019 Let $\mathcal{O}_{\infty, \lambda, \delta}^{\text{linear}}$ denote the set of minimizers of (94) when the empirical loss $s(\mathbf{H}; \lambda, \delta, \mathbf{X})$ is replaced by
1020 its population counterpart $\mathbb{E}[s(\mathbf{H}; \lambda, \delta, \mathbf{X})]$. Let us collect all the parameters in assumption A.

$$H^0 = \begin{bmatrix} c_1 & \dots & c_p \\ w_1 & \dots & w_p \end{bmatrix} \in \mathbb{R}^{p+1 \times p} \quad (96)$$

1021 **Theorem 4.** Suppose Assumption A holds, then $X \sim \text{MultiBernoulli}(\mathbf{p})$ where $\mathbf{p} > 0$. Moreover,
1022 there exist $\lambda, \delta > 0$ sufficiently small such that $(H^0, G^0) \in \mathcal{O}_{\infty, \lambda, \delta}^{\text{linear}}$ where G^0 is the ground truth
1023 graph in Assumption A.

1024 It is important to note that under this assumption

$$\mathcal{O}_{\infty, \lambda, \delta}^{\text{linear}} \neq \mathcal{E}_{\min}(\mathbf{p}), \quad (97)$$

1025 because there may exist a topological sort π for which $(\mathbf{f}_\pi, G_\pi) \in \mathcal{E}_{\min}(\mathbf{p})$ involves higher-order
1026 terms, whereas every solution in $\mathcal{O}_{\infty, \lambda, \delta}^{\text{linear}}$ contains only first-order terms.

1027 D Experiments

1028 In this section, we present comprehensive experimental details, including the graph types evaluated,
1029 the data-generation process, the baseline methods for comparison, the steps required to reproduce our
1030 implementation, and the evaluation metrics employed.

1031 D.1 Experimental Setting

1032 In this section, we outline the process for generating graphs and data. For each model, a random graph
1033 G is generated using one of two types of random graph models: Erdős-Rényi(ER) or Scale-Free (SF).
1034 The models are specified to have, on average, kp edges, where $k \in \{1, 2, 4\}$. These configurations
1035 are denoted as ER k or SF k , respectively.

- 1036 • *Erdős-Rényi(ER)*, Random graphs whose edges are add independently with equal probability.
1037 We simulated models with $p, 2p$ and $4p$ edges (in expectation) each, denoted by ER1, ER2,
1038 and ER4 respectively.
- 1039 • *Scale-free network(SF)*. Network simulated according to the preferential attachment process.
1040 We simulated scale-free network with $p, 2p$ and $4p$ edges and $\beta = 1$, where β is the exponent
1041 used in the preferential attachment process.

Algorithm 4: Generate data matrix \mathbf{X}

Input: DAG G , sample size n , interaction type $\tau \in \{1\text{st}+2\text{nd}, 1\text{st}+\text{pth}, 1\text{st}+2\text{nd}+\text{pth}, 2\text{nd}, \text{pth}\}$

Output: $\mathbf{X} \in \{0, 1\}^{n \times p}$

```

1 Compute a topological ordering  $\pi$  of  $G$ 
2 for  $j \leftarrow 1$  to  $p$  do
3   Sample  $w_{\text{PA}(\pi(j))} = (w_k)_{k=1}^{2^{|\text{PA}(\pi(j))|}} \in \mathbb{R}^{2^{|\text{PA}(\pi(j))|}}$ ,  $w_k \stackrel{\text{iid}}{\sim} \text{Unif}([-2, -1] \cup [1, 2])$ .
4   if  $\tau = 1\text{st}+2\text{nd}$  then
5      $q \leftarrow \text{logistic}(w_{\text{PA}(\pi(j))}^\top \Phi^{1\text{st}+2\text{nd}}(\mathbf{X}_{\text{PA}(\pi(j))}))$ 
6   else if  $\tau = 1\text{st}+\text{pth}$  then
7      $q \leftarrow \text{logistic}(w_{\text{PA}(\pi(j))}^\top \Phi^{1\text{st}+\text{pth}}(\mathbf{X}_{\text{PA}(\pi(j))}))$ 
8   else if  $\tau = 2\text{nd}$  then
9      $q \leftarrow \text{logistic}(w_{\text{PA}(\pi(j))}^\top \Phi^{2\text{nd}}(\mathbf{X}_{\text{PA}(\pi(j))}))$ 
10  else
11     $q \leftarrow \text{logistic}(w_{\text{PA}(\pi(j))}^\top \Phi^{\text{pth}}(\mathbf{X}_{\text{PA}(\pi(j))}))$ 
12   $\mathbf{X}_{\pi(j)} \sim \text{Bernoulli}(q)$ 

```

1042 **General binary data** Since we wish to study structure learning for general binary data, Theorem 1
 1043 implies that for any $p > 0$,

$$X \sim \text{MultiBernoulli}(\mathbf{p}) \quad (98)$$

1044 can be generated via the SEM (7). To allow different interaction orders, define the following extended
 1045 feature maps for $X = (X_1, \dots, X_p)$:

$$\begin{aligned}
 \Phi^{1\text{st}+2\text{nd}}(X) &= (\underbrace{1}_{\text{constant}}, \underbrace{X_1, \dots, X_p}_{\text{first order}}, \underbrace{X_1 X_2, \dots, X_{p-1} X_p}_{\text{second order}}, \underbrace{0}_{\text{third order}}, \underbrace{0}_{\text{forth to } (p-1)\text{-th order}}, \underbrace{0}_{p\text{-th order}})^\top \in \mathbb{R}^{2^p} \\
 \Phi^{1\text{st}+2\text{nd}+\text{pth}}(X) &= (\underbrace{1}_{\text{constant}}, \underbrace{X_1, \dots, X_p}_{\text{first order}}, \underbrace{X_1 X_2, \dots, X_{p-1} X_p}_{\text{second order}}, \underbrace{0}_{\text{third order}}, \underbrace{0}_{\text{forth to } (p-1)\text{-th order}}, \underbrace{X_1 \dots X_p}_{p\text{-th order}})^\top \in \mathbb{R}^{2^p} \\
 \Phi^{1\text{st}+\text{pth}}(X) &= (\underbrace{1}_{\text{constant}}, \underbrace{X_1, \dots, X_p}_{\text{first order}}, \underbrace{0}_{\text{second order}}, \underbrace{0}_{\text{third order}}, \underbrace{0}_{\text{forth to } (p-1)\text{-th order}}, \underbrace{X_1 X_2 \dots X_p}_{p\text{-th order}})^\top \in \mathbb{R}^{2^p} \\
 \Phi^{2\text{nd}}(X) &= (\underbrace{1}_{\text{constant}}, \underbrace{0}_{\text{first order}}, \underbrace{X_1 X_2, \dots, X_{p-1} X_p}_{\text{second order}}, \underbrace{0}_{\text{third order}}, \underbrace{0}_{\text{forth to } (p-1)\text{-th order}}, \underbrace{0}_{p\text{-th order}})^\top \in \mathbb{R}^{2^p} \\
 \Phi^{\text{pth}}(X) &= (\underbrace{1}_{\text{constant}}, \underbrace{0}_{\text{first order}}, \underbrace{0}_{\text{second order}}, \underbrace{0}_{\text{third order}}, \underbrace{0}_{\text{forth to } (p-1)\text{-th order}}, \underbrace{X_1 X_2 \dots X_p}_{p\text{-th order}})^\top \in \mathbb{R}^{2^p}
 \end{aligned} \quad (99)$$

1046 where in each vector the nonzero blocks correspond respectively to the constant term, first-order
 1047 terms, second-order terms, and highest-order term. By convention, if $X = \emptyset$, then all four maps
 1048 reduce to the scalar $(1) \in \mathbb{R}$. When applied to a data matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, each $\Phi(\mathbf{X})$ operates
 1049 row-wise.

1050 Finally, given a random DAG $B \in \{0, 1\}^{p \times p}$ sampled from one of our graph models, we generate \mathbf{X}
 1051 using Algorithm 4, choosing the desired interaction map according to whether we study first+second,
 1052 first+highest, second, or highest-order interactions.

1053 **Simulation** We generate random dataset $\mathbf{X} \in \mathbb{R}^{n \times p}$ by sampling i.i.d from the models described
 1054 above. For each simulation, we produce datasets with n sample cross graphs with p nodes.

- 1055 • **(Small Graph)** $p = \{5, 6, 7, 8, 9\}$, $k = \{1, 2\}$, $n = 10000$ and graph types: {ER,SF}
- 1056 • **(Large Graph)** $p = \{10, 20, 30, 40\}$, $k = \{1, 2, 4\}$, $n = 1000$ and graph types: {ER,SF}

1057 **Implementation** For each dataset, we applied several structural learning algorithms, including
 1058 fast greedy equivalence search (FGES [34]), constraint-based methods (PC [38]), DAGMA [4],

NOTEARS [47]. The implementation details are provided in the following paragraph. After running the algorithms, a post-processing threshold of 0.3 was applied to the estimated B_{est} to prune small values, following the same procedure in [46, 4].

- Fast Greedy Equivalence Search (FGES [34]) is based on greedy search and assumes linear dependency between variables. The implementation is based on the py-tetrad package, available at <https://github.com/cmu-phil/py-tetrad>. We use `search.use_bdeu(sample_prior=10, structure_prior=0)`.
- PC [38] is constraint-based method and based on uses conditional independence induced by causal relationships to learn those causal relationships. The implementation is based on the py-tetrad package, available at <https://github.com/cmu-phil/py-tetrad>. We use `search.use_chi_square(alpha=0.1)`.
- NOTEARS-MLP [47] is a continuous DAG-learning method that employs a least-squares loss with ℓ_1 regularization. Its Python implementation is available at <https://github.com/xunzheng/notears>. In our variant, we insert a sigmoid activation $\sigma(x) = 1/(1 + \exp(-x))$ on the final layer and replace the original loss with the cross-entropy (logistic) loss to accommodate binary data. After estimating the weighted adjacency matrix W_{est} via NOTEARS-MLP, we prune all entries below a threshold of 0.3, compute a topological ordering of the resulting graph, and then apply Algorithm 2 with first and second order to obtain the final structure. Finally, we remove any remaining edges whose weight does not exceed 1.0 to eliminate spurious connections. We name this method as NOTEARS-MLP-REG. These heuristic methods are applied to larger graphs with $d \in \{10, 20, 30, 40\}$.
- DAGMA [4] is a continuous DAG-learning algorithm that achieves improved accuracy and faster computation, with barrier methods. Its implementation can be found at <https://github.com/kevinsbello/dagma>. To highlight that the original DAGMA only models first-order interactions, we refer to it as DAGMA-1ST. By solving (13) while incorporating all higher-order interactions, we arrive at our extended method, denoted DAGMA-HO. For small graphs, we implement the full formulation (13) including all higher-order interactions; hence, DAGMA-HO is applied for $d \in \{5, 6, 7, 8, 9\}$.

Hyperparameter tuning Theorem 3 indicates that one should ideally choose small values of λ and δ for the quasi-MCP penalty. In practice, however, achieving the global optimum of (13) is infeasible, and if λ and δ are too small the penalty becomes ineffective and the algorithm may fail to recover the sparsest solution. To mitigate this, we adopt the continuation strategy of Deng et al. [15]: start with relatively large λ and δ , solve (13) via DAGMA-HO to obtain an initial estimate \mathbf{H}_{est} , then iteratively shrink λ and δ by a factor $\gamma < 1$, using the previous estimate as the warm start for the next run of DAGMA-HO. We terminate when the negative log-likelihood $s(\mathbf{H}_{\text{est}}; \mathbf{X})$ ceases to decrease. Empirically, $\gamma = 0.5$, $\lambda = 0.05$, and $\delta = 0.2$ perform well in our experiments.

Equipment The experiments are conducted in the following CPU architectures

- Intel Broadwell—28 cores @ 2.4 GHz with 64 GB memory per node
- Intel Skylake—40 cores @ 2.4 GHz with 96 GB memory per node

D.2 Metrics

- **Structural Hamming distance (SHD)**: A standard benchmark in the structure learning literature that counts the total number of edges additions, deletions, and reversals needed to convert the estimated graph into the true graph. Since our data specified in (1) is nonidentifiable, the Structural Hamming Distance (SHD) is calculated with respect to the completed partially directed acyclic graph (CPDAG) of the ground truth and B_{est} .

1105 E Additional Figures

1106 E.1 Small graphs

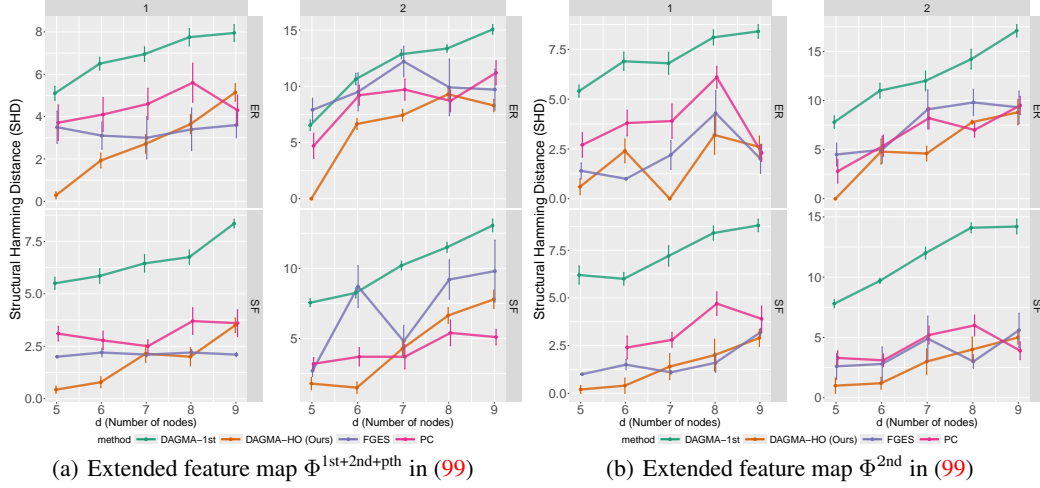


Figure 3: Results in terms of SHD between MECs of estimated graph and ground truth. Lower is better. Column: $k = \{1, 2\}$. Row: random graph types. $\{ER, SF\}$ - $k = \{\text{Scale-Free, Erdős-Rényi}\}$ graphs with kd expected edges. Here $p = \{5, 6, 7, 8, 9\}$. DAGMA [4] is renamed as “DAGMA-1st”, to emphasize only linear term is used. Error bars denote the standard error computed over 10 replications.

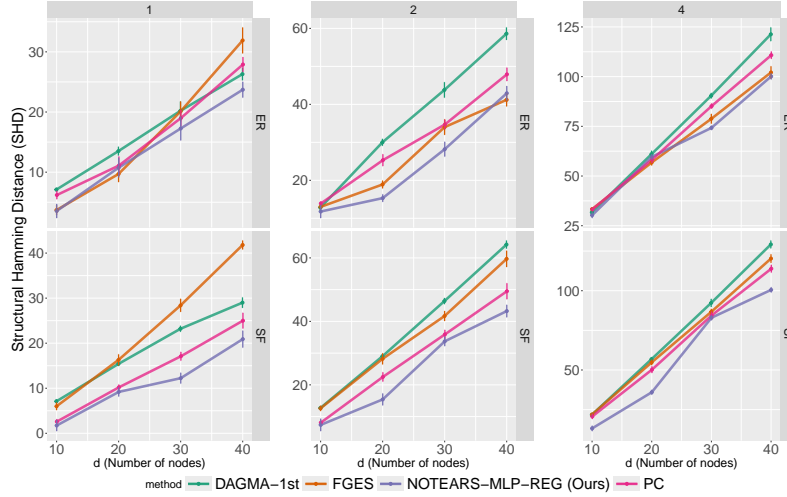


Figure 4: Results in terms of SHD between MECs of estimated graph and ground truth. Lower is better. Data are generated using extended feature map $\Phi^{1st+pth}$ in (99). Column: $k = \{1, 2, 4\}$. Row: random graph types. $\{ER, SF\}-k = \{\text{Scale-Free, Erdős-Rényi}\}$ graphs with kd expected edges. Here $p = \{10, 20, 30, 40\}$. NOTEARS-MLP-REG is our two stage approach. DAGMA [4] is renamed as “DAGMA-1st”, to emphasize only linear term is used.

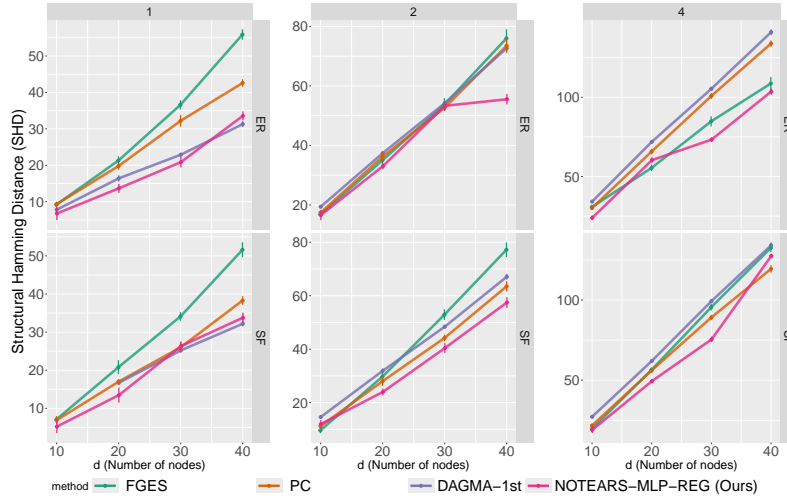


Figure 5: Results in terms of SHD between MECs of estimated graph and ground truth. Lower is better. Data are generated using extended feature map Φ^{2nd} in (99). Column: $k = \{1, 2, 4\}$. Row: random graph types. $\{ER, SF\}-k = \{\text{Scale-Free, Erdős-Rényi}\}$ graphs with kd expected edges. Here $p = \{10, 20, 30, 40\}$. NOTEARS-MLP-REG is our two stage approach. DAGMA [4] is renamed as “DAGMA-1st”, to emphasize only linear term is used.

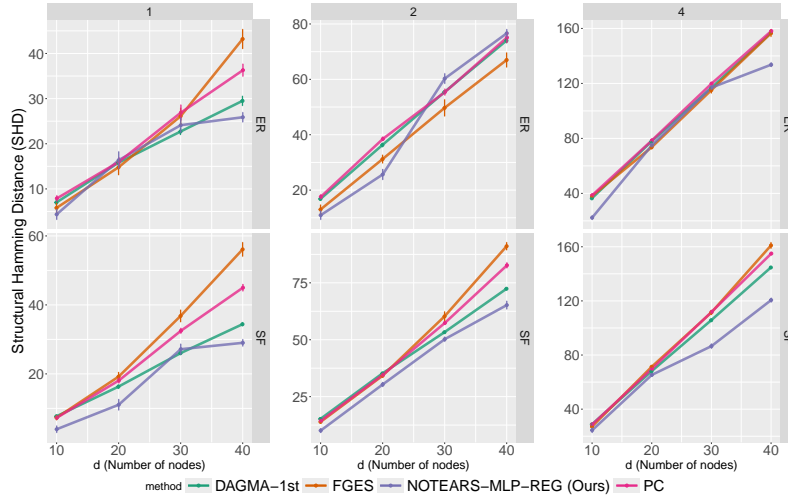


Figure 6: Results in terms of SHD between MECs of estimated graph and ground truth. Lower is better. Data are generated using extended feature map Φ^{pth} in (99). Column: $k = \{1, 2, 4\}$. Row: random graph types. $\{\text{ER}, \text{SF}\} \cdot k = \{\text{Scale-Free}, \text{Erdős-Rényi}\}$ graphs with kd expected edges. Here $p = \{10, 20, 30, 40\}$. NOTEARS-MLP-REG is our two stage approach. DAGMA [4] is renamed as “DAGMA-1st”, to emphasize only linear term is used.