Causal Discovery with Mixed Linear and Nonlinear Additive Noise Models: A Scalable Approach

Wenqin Liu WENQINL@STUDENT.UNIMELB.EDU.AU
The University of Melbourne

Biwei Huang BIH007@UCSD.EDU
University of California, San Diego

Erdun Gao ERDUN.GAO@STUDENT.UNIMELB.EDU.AU
The University of Melbourne

Qiuhong Ke QIUHONG.KE@MONASH.EDU
Monash University

Howard Bondell HOWARD.BONDELL@UNIMELB.EDU.AU
The University of Melbourne

Mingming Gong MINGMING.GONG@UNIMELB.EDU.AU
The University of Melbourne

Editors: Francesco Locatello and Vanessa Didelez

Abstract

Estimating the structure of directed acyclic graphs (DAGs) from observational data is challenging due to the super-exponential growth of the search space with the number of nodes. Previous research primarily focuses on identifying a unique DAG under specific model constraints in linear or nonlinear scenarios. However, real-world scenarios often involve causal mechanisms with a mixture of linear and nonlinear characteristics, which has received limited attention in existing literature. Due to unidentifiability, existing algorithms relying on fully identifiable conditions may produce erroneous results. Although traditional methods like the PC algorithm can be employed to uncover such graphs, they typically yield only a Markov equivalence class. This paper introduces a novel causal discovery approach that extends beyond the Markov equivalence class, aiming to uncover as many edge directions as possible when the causal graph is not fully identifiable. Our approach exploits the second derivative of the log-likelihood in observational data, harnessing scalable machine learning approaches to approximate the score function. Overall, our approach demonstrates competitive accuracy comparable to current state-of-the-art techniques while offering a significant improvement in computational speed.

Keywords: Causal Discovery; Additive Noise Models; Directed Acyclic Graph; Score Matching; Identifiability; Scalability

1. Introduction

Discovering the causal relationships among concerned events is a fundamental task and contributes across diverse scientific disciplines including finance, genetics, neuroscience, and artificial intelligence (Pearl et al., 2000). To achieve this, the golden rule is conducting randomized controlled trials (RCTs), which, while highly effective, also pose practical challenges arising from their substantial costs and ethical considerations. To address this issue, in this work, we focus on estimating the causal relationships from purely observational data, i.e., identifying the structure of the causal directed acyclic graph (DAG) that underlies a given dataset.

In essence, uncovering causality from observational data presents a challenging task and it is ill-posed: multiple generative models featuring diverse causal structures can yield the same data distribution. Consequently, traditional solutions such as the PC algorithm (Spirtes et al., 2000), do not yield a unique DAG but rather a Markov equivalence class. To make the problem well-posed, additional assumptions in the generative process become essential. These restricted models behave differently and typically lead to a setting where every DAG defines a unique model for observational data. This has been demonstrated, for instance, in the linear non-gaussian acyclic model (LiNGAM) (Shimizu et al., 2006), linear additive noise models (ANMs) with known or equal variance (Peters and Bühlmann, 2014; Ghoshal and Honorio, 2018; Loh and Bühlmann, 2014), nonlinear ANMs (Hoyer et al., 2008; Peters et al., 2014), and post-nonlinear models (Zhang and Hyvarinen, 2012). Although these studies have demonstrated full identifiability under certain restrictions, they typically constrain the model to a single scenario, such as nonlinear or linear. Yet, real-world scenarios often involve causal mechanisms with a mixture of linear and nonlinear characteristics. Applying the above mentioned linear or nonlinear ANMs to such data may produce erroneous results due to model misspecification. Although classic methods like the PC algorithm can be employed to uncover such graphs, they typically yield only a Markov equivalence class, leaving many edge orientations undetermined.

In this paper, we consider the more practical scenario with mixed linear and nonlinear ANMs, and propose a novel causal discovery approach that extends beyond the Markov equivalence class, aiming to uncover as many edge directions as possible when the causal graph is not fully identifiable. This is achieved by exploiting the second derivative of the log-likelihood in observational data and harnessing scalable machine learning approaches to approximate the score function. We demonstrate that in a causal system involving a combination of linear and nonlinear ANMs, it is possible to identify most directions by analyzing the associated observational score. In addition, our approach enables parallel processing to enhance the overall scalability.

Our contributions can be summarised as follows:

- We examine a more practical setting involving data from an underlying causal model with mixed linear and nonlinear causal mechanisms. In this context, our approach identifies more directions compared to traditional algorithms like the PC algorithm, which is limited to identifying up to the Markov Equivalence class.
- We design an algorithm for inferring the causal graph underlying an ANM comprising both non-identifiable and identifiable components. By eliminating common hypotheses (e.g. equal variance) assumed in the identifiable linear Gaussian additive model, our approach broadens the scope of applicability for causal discovery. This provides a comparable method to current state-of-the-art techniques with theoretical guarantees, especially in critical settings where validating the variances of Gaussian noise assumption is challenging.
- Our algorithm avoids the need for exhaustive combinatorial searches and mitigates concerns related to multiple testing. Simultaneously, it facilitates parallel processing effectively reducing the computational complexity with a larger number of variables.

2. Related Work

In traditional causality research, algorithms for discovering causal relationships are categorized into three classes (Glymour et al., 2019; Schölkopf et al., 2021). Constraint-based approaches, exemplified by PC (Spirtes and Glymour, 1991), fast causal inference (FCI) (Spirtes, 2001), and SGS (Spirtes et al., 2000), evaluate the conditional independence between variables and search for graph structures that satisfy these conditions under a faithfulness assumption. However, these approaches do not produce a unique DAG but rather an equivalence class that may include more than one DAG. The main bottleneck of these approaches lies in the difficulty of conditional independence testing (Shah and Peters, 2020). Score-based methods define a suitable score function and search for the graph that best fits the data within an extensive graph space. Greedy approaches like greedy equivalence search (GES) (Chickering, 2002; Huang et al., 2018) are employed for this exploration, but the scalability is limited due to the super-exponential growth of the space with the number of nodes. The inference of causal relations from observational data typically encounters non-identifiability issues, necessitating additional assumptions.

Causal discovery for non-linear ANMs. Hence, previous studies have investigated identifiable classes of DAG models by placing constraints on distributions. Models are identifiable when link functions are assumed to be twice continuously differentiable, and each variable is determined by a nonlinear function of its parents and an error term (Hoyer et al., 2008; Mooij et al., 2009; Peters et al., 2012). The causal additive model (CAM) (Bühlmann et al., 2014) assumes an additive structure for the link functions. They estimate a topological order by greedily maximizing data likelihood and subsequently pruning the DAG using sparse regression techniques. Recently, Rolland et al. (2022) introduced an order-based approach using score matching to identify causal graph leaves with linear time complexity in the number of nodes. However, the final graph is obtained through classical pruning techniques used in CAM with high time complexity. Building on this, Montagna et al. (2023b) eliminates the time-consuming pruning step. Yet, both methods prove ineffective in identifying the causal graph when assuming a linear additive model with Gaussian noise.

Causal discovery for linear ANMs. Linear non-Gaussian ANMs are proven identifiable, where each variable is determined by a linear function of its parents plus an independent error term (Shimizu et al., 2006; Zhang and Hyvarinen, 2012). Specifically, models are identifiable if one of its parents or error terms belongs to a set of some non-Gaussian distributions. In terms of linear Gaussian additive noise models, Peters and Bühlmann (2014) proves its identifiability under equal or unknown error variances. Recent work by Ghoshal and Honorio (2018) demonstrates the identifiability of linear Gaussian Structural Equation Models (SEMs) with unknown heterogeneous error variances under certain assumptions. However, assumptions about error variances may be unrealistic in real-world data, and the assumptions about all non-Gaussian error distributions and all nonlinear dependency functions might be similarly impractical.

Score matching in causal discovery. To approximate the score of the data distribution, Rolland et al. (2022) extends recent work on score matching and density gradient estimation over an RBF kernel (Li and Turner, 2017). The score function is generally learned by fitting a neural network that minimizes the empirical Fisher divergence (Hyvärinen and Dayan, 2005; Song and Ermon, 2019; Zheng et al., 2023). While effective, such a method is computationally expensive and requires tuning multiple training parameters. Similar to Rolland et al. (2022), we choose to instead minimize the kernelized Stein discrepancy which provides a closed-form solution and allows fast estimation at all observations. This method performs similarly to score matching in practice while being significantly faster. The asymptotic consistency of the Stein gradient estimator and its relation to score matching was thoroughly analyzed in Barp et al. (2019).

The remainder of the paper is organized as follows: Section 3 introduces notations and defines the problem. Section 4 presents LNMIX (Linear and Nonlinear Mixture model), an algorithm for causal graph inference based on the model introduced in Section 3. Section 5 details experimental performance against benchmarks, and Section 6 concludes the work and future directions.

3. Preliminaries

In this section, we introduce our notations and formalize the problem of learning a mixture of linear and nonlinear SEMs from observational data.

Let $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ represent a DAG, consisting of a set of vertices $\mathbf{V} = \{1, \ldots, d\}$ and a set of directed edges $\mathbf{E} \in \mathbf{V} \times \mathbf{V}$. $(i, j) \in \mathbf{E}$ indicates that there exists a directed edge from vertex i to vertex j, i.e. $i \rightarrow j$. In $\mathcal{G}$, we use PA_i and CH_i to denote the set of parents and the set of children of the i-th vertex, respectively. The spouse set of the i-th vertex is defined as $\mathrm{SP}_i = \{k \in \mathrm{PA}_j \mid j \in \mathrm{CH}_i\}$. The Markov blanket of the i-th vertex is defined as $\mathrm{MB}(i) = \mathrm{PA}_i \cup \mathrm{CH}_i \cup \mathrm{SP}_i$. Let $\mathcal{G}^m$ represent a moral graph of $\mathcal{G}$, including undirected edges connecting variables if there exists a directed edge between them, and if they are parents of the same node in $\mathcal{G}$. In other words, the moral graph of a DAG takes the form of an undirected graph where each node is now linked to its Markov Blanket. A vertex $i \in \mathbf{V}$ is a terminal vertex in $\mathcal{G}$ if $\mathrm{CH}_i = \emptyset$. A vertex j is a descendant of i if there exists a directed path from i to j in $\mathcal{G}$ and the set of descendants of the i-th vertex is denoted by DE_i. For each $i \in \mathbf{V}$, there is a random variable $X_i \in \mathbb{R}$ and $\mathbf{X} \in \mathbb{R}^{n \times d}$ is a data matrix consisting of n i.i.d. observations of the random vector $X = (X_1, \ldots, X_d)$. For any set $\mathbf{Z} \subset \mathbf{V}$, we denote $X_{\mathbf{Z}} = \{X_i : i \in \mathbf{Z}\}$. For any matrix $\mathbf{M}$, we denote its support set as $\mathrm{Supp}(\mathbf{M}) = \{(i, j) \in \mathbf{V} \times \mathbf{V} \mid \mathbf{M}_{i,j} \neq 0\}$. Additionally, we denote the set $-i \overset{\text{def}}{=} \mathbf{V} \setminus \{i\}$.

Model definition. The random vector X follows an ANM and the SEM is as follows

$$X_i = f_i(X_{\mathrm{PA}_i}) + N_i \qquad \forall i \in \mathbf{V}, \tag{1}$$

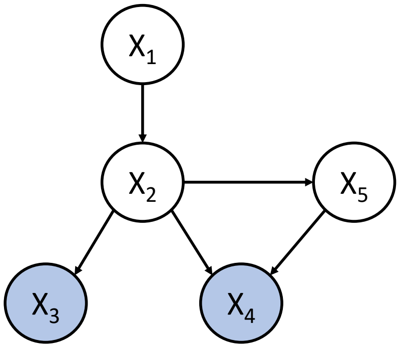
where noise variables $N_i \sim \mathcal{N}(0, \sigma_i^2)$, and $N_i \perp\!\!\!\perp X_1, \ldots, X_{i-1}$. The noise variables are independent such that the joint distribution is $p(N) = \prod_{i=1}^{d} p_i(N_i)$. The link function $f_i : \mathbb{R}^{|\mathrm{PA}_i|} \rightarrow \mathbb{R}$ can be either a linear or nonlinear function of the variables in its parent set. If the link function f_i is nonlinear, we assume that it is continuously twice differentiable. Figure 1 illustrates an example of the model involving a combination of linear and nonlinear additive Gaussian noise SEMs.

The work relies on several fundamental assumptions: Causal Sufficiency, the Causal Markov Condition, and Faithfulness. Causal Sufficiency implies that all confounders of the relevant variables are observed in the given dataset. The Causal Markov Condition states that a variable X is independent of every other variable (except X's effects) conditional on all its direct causes. Let P represent the probability distribution over the vertices in $\mathbf{V}$ generated by the causal structure represented by $\mathcal{G}$. $\mathcal{G}$ and P satisfy the faithfulness if and only if every conditional independence relation true in P is entailed by the Causal Markov Condition applied to $\mathcal{G}$.

Given the SEM, the joint distribution $p(x)$ is fully determined and factorized according to the DAG structure $\mathcal{G}$ as follows:

$$p(x) = \prod_{i=1}^{d} p_i(x_i \mid x_{\mathrm{PA}_i}), \tag{2}$$

where p_i is the probability density function of X_i.



$$\begin{aligned}
X_1 &= f_1(N_1) \overset{\text{def}}{=} N_1,\\
X_2 &= f_2(X_1, N_2) \overset{\text{def}}{=} \sin(4X_1) + N_2,\\
X_3 &= f_3(X_2, N_3) \overset{\text{def}}{=} -1.1X_2 + N_3,\\
X_4 &= f_4(X_2, X_5, N_4) \overset{\text{def}}{=} 0.7X_2 + 1.4X_5 + N_4,\\
X_5 &= f_5(X_2, N_5) \overset{\text{def}}{=} 0.2{X_2}^3 + N_5.
\end{aligned}$$

Figure 1: An illustrative example of the mixture model comprises a combination of linear and nonlinear additive Gaussian noise models. The equations represent the structural equation model governing the data, and the ground truth Directed Acyclic Graph (DAG) is depicted on the left. Nodes represented in blue denote a causal mechanism where these nodes are linear functions of their respective parent nodes. In contrast, unfilled nodes indicate a causal mechanism characterized by nonlinear functions concerning their parent nodes.

4. Causal discovery via Jacobian of the score function

In this section, we demonstrate the process of recovering the causal graph from the score function, considering a combination of linear and nonlinear causal mechanisms defined in Equation (1). The proposed methodology comprises three steps. It first identifies the moral graph from observational data. Then the algorithm addresses the orientation of edges in the moral graph, distinguishing between parents and children in the context of linear and nonlinear ANMs. Finally, V-structures are identified within fully connected triangles.

4.1. Identifying the Moral Graph

According to the model defined in Equation (1), the log-likelihood of the joint distribution function is as follows:

$$\log p(\mathbf{x}) = \log \prod_{i=1}^{d} p_i(x_i \,|\, x_{\mathrm{PA}_i}) = \sum_{i=1}^{d} \log p_i(x_i \,|\, x_{\mathrm{PA}_i}). \tag{3}$$

The i-th component of the score function $s(x) \equiv \nabla_x \log p(\mathbf{x})$ with respect to the data point x is

$$\begin{aligned}
s_i(x) &= \frac{\partial}{\partial x_i}\left[\log p_i(x_i \,|\, x_{\mathrm{PA}_i}) + \sum_{j \in \mathrm{CH}_i} \log p_j(x_j \,|\, x_{\mathrm{PA}_j})\right]\\
&= \frac{\partial}{\partial x_i} \log p_i(x_i \,|\, x_{\mathrm{PA}_i}) + \sum_{j \in \mathrm{CH}_i} \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_i} \frac{\partial}{\partial x_j} \log p_j(x_j \,|\, x_{\mathrm{PA}_j}).
\end{aligned} \tag{4}$$

With conditioning on parents, the marginal of X_i is equivalent to the distribution of N_i shifted by the value of the mechanism $f_i(x_{\mathrm{PA}_i})$. Alternatively, $p_i(x_i \,|\, x_{\mathrm{PA}_i})$ can be replaced by $p(n_i = x_i - f_i(x_{\mathrm{PA}_i}) \,|\, x_{\mathrm{PA}_i})$, which allows to rewrite the score function as:

$$s_i(x) = \frac{\partial}{\partial n_i} \log p_i(n_i) + \sum_{j \in \mathrm{CH}_i} \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_i} \frac{\partial}{\partial n_j} \log p_j(n_j). \tag{5}$$

When $N_i \sim \mathcal{N}(0, \sigma_i^2)$ with a probability density function of $p(n_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{n_i}{\sigma}\right)^2}$, the i-th component of the score function $s(x) \equiv \nabla \log p(x)$ is:

$$\begin{aligned} s_i(x) &= -\frac{\partial}{\partial n_i} \frac{1}{2} \left(\frac{n_i}{\sigma_i}\right)^2 + \sum_{j \in \mathrm{CH}_i} \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_i} \frac{\partial}{\partial n_j} \frac{1}{2} \left(\frac{n_j}{\sigma_j}\right)^2 \\ &= -\frac{x_i - f_i(x_{\mathrm{PA}_i})}{\sigma_i^2} + \sum_{j \in \mathrm{CH}_i} \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_i} \frac{x_j - f_j(x_{\mathrm{PA}_j})}{\sigma_j^2}. \end{aligned} \tag{6}$$

An observation from Equation (6) is that if node j is part of the Markov blanket of node i, then $\frac{\partial s_i(x)}{\partial x_j} \neq 0$. The following lemma demonstrates that this condition offers a reliable way to identify the moral graph using the Jacobian of the score function.

Lemma 1. *Let p be the probability density function of a random variable X defined via an additive Gaussian noise model, and let $s(x) \equiv \nabla \log p(x)$ be the associated score function. Then, $\forall j \in \{1, \ldots, d\}$,*

$$\frac{\partial s_i(x)}{\partial x_j} = 0 \iff j \notin MB(i),$$

and the Markov Blanket is unique under the faithfulness assumption.

Detailed proof is given in Appendix A. By employing Lemma 1, we can derive the moral graph from observational data. To clarify the derivation of the moral graph, if the Jacobian of the score between two variables is zero, there is no edge between them, and vice versa. Therefore, this method only requires computing the support of the Jacobian of the score function, i.e. Supp($\frac{\partial^2 \log p(x)}{\partial x_i x_j}$). This eliminates the need for the extensive conditional independence testing involved in the well-known PC algorithm (Spirtes and Glymour, 1991). As a result, this substantially speeds up the computation process.

4.2. Edge Orientation from the Jacobian of the Score Function

The method outlined above determines the moral graph under observational data and returns a set of edges without identifying the parents, children, or spouses. This graph closely resembles the original causal graph as it contains all arcs as undirected edges, and additionally links spouses together. The subsequent step is to transform it into a partially oriented causal DAG by performing arc orientation and removing spouse links.

To convert the moral graph into a partially oriented causal DAG based on observational data from mixed linear and nonlinear models, we introduce the following lemmas to identify the orientation of undirected edges within the moral graph.

Lemma 2. *(Partially linear additive Gaussian noise model) Let p be the joint probability density function of random variables $X \in \mathbb{R}^d$ defined via a model involving a combination of linear and nonlinear additive Gaussian noise ANMs. Suppose X_j's causal model f_j is nonlinear, and all children of node j have linear causal models. Then the following three statements hold:*

(i) $\forall x_j$, $\frac{\partial s_j(x)}{\partial x_j} = c$, *with* $c \in \mathbb{R}$ *independent of* x, *and hence,* $Var_x[\frac{\partial s_j(x)}{\partial x_j}] = 0$.

(ii) $\forall k \in PA_j$, $s_j(x)$ *depends on* x_k, *and hence,* $Var_x[\frac{\partial s_j(x)}{\partial x_k}] \neq 0$.

(iii) $\forall k \in CH_j$, $\frac{\partial s_j(x)}{\partial x_k} = c$, *with* $c \in \mathbb{R}$ *independent of* x, *and hence,* $Var_x[\frac{\partial s_j(x)}{\partial x_k}] = 0$.

The detailed proof of Lemma 2 is available in Appendix B. Lemma 2 establishes the feasibility of distinguishing between parents and children in a model that includes a mixture of linear SEMs and nonlinear SEMs. According to the lemma, when the variance of the Jacobian of the score function in the diagonal term is a constant, non-constant variances of the Jacobian for the off-diagonal entries indicate corresponding parents of the node. Conversely, if the variance of the Jacobian of the score function is constant, the node must be a child. Examining the structure of the variance of the Jacobian of score function allows us to identify some directions when a mixture of linear SEMs and nonlinear SEMs is present in the local graph. Furthermore, in instances where local causal structure involves only nonlinear functions, the Jacobian of the score function remains a valuable tool for identifying the nonlinear terminal nodes along with their associated parents, as demonstrated in Lemma 3 and Lemma 4.

Lemma 3. *(Adopted from Rolland et al. (2022)) Let* p *be the probability density function of a random variable* X *defined via a nonlinear additive Gaussian noise model, and let* $s(x) = \nabla_x \log p(x)$ *be the associated score function. Then,* $\forall j \in \{1, \ldots, d\}$, *we have:*

(i) *j is a leaf* $\Leftrightarrow \forall x$, $\frac{\partial s_j(x)}{\partial x_j} = c$, *with* $c \in \mathbb{R}$ *independent of* x, *i.e.,* $Var_x[\frac{\partial s_j(x)}{\partial x_j}] = 0$.

(ii) *If j is a leaf, i is a parent of j* $\Leftrightarrow s_j(x)$ *depends on* x_i, *i.e.,* $Var_x[\frac{\partial s_j(x)}{\partial x_i}] \neq 0$.

Lemma 4. *(Adopted from Montagna et al. (2023b)) Let* p *be the probability density function of a random variable* X *defined via a nonlinear additive Gaussian noise model. Let* $s(x) = \nabla_x \log p(x)$ *be the associated score function. Then,* $\forall j \in \{1, \ldots, d\}$, *for a given leaf l, we have,*

$$\mathbb{E}\left[\left|\frac{\partial s_l(x)}{\partial x_j}\right|\right] \neq 0 \iff j \in PA_l.$$

Lemma 3 illustrates that in nonlinear additive Gaussian noise models, only leaf nodes exhibit the characteristic of having a constant diagonal element in the score's Jacobian. This feature offers a way to identify a leaf in the causal graph by utilizing knowledge of the variance of the score's Jacobian diagonal elements. The second criterion in Lemma 3 serves as a tool to identify the parents of these leaf nodes, and Lemma 4 presents a theoretically equivalent but more practically robust formulation. Relying on the sample mean of the absolute value of the score's Jacobian entries is a more robust practical choice for identifying the parents of leaf nodes. Estimating a lower moment results in a lower error, as estimating variance necessitates estimating the mean first, and any statistical error in the mean estimator affects the variance estimator. Thus, it becomes the preferable choice. By iteratively applying this method and removing the identified leaves, we can identify the directions of nonlinear components in the graph.

Definition 1. ***(Linear block)****. Let* p *be the joint probability density function of a random variable* X *defined via a model involving a combination of linear and nonlinear additive Gaussian noise SEMs. The linear block is defined as follows:*

(i) j is a root vertex and it has only one child k and X_k is a linear function of X_j plus an independent error term.

(ii) j is a terminal vertex and it has only one parent k and X_j a linear function of X_k plus an independent error term.

(iii) vertexes $\{i, j, k\}$ form a V-structure and there exists only one node l in $\mathcal{G}$ connected to one of the vertexes $\{i, j, k\}$ e.g., x_j and x_l is a linear function of x_j plus an independent error term or x_j is a linear function of x_l plus an independent error term.

Regarding the criteria (i) and (ii) in the definition 1, we can identify the vertexes using the score's Jacobian. In the score's Jacobian, $\frac{\partial s_j(x)}{\partial x_i} \neq 0$ for $i \in \{j, k\}$. $\forall x$, $\frac{\partial s_j(x)}{\partial x_j} = c$ and $\frac{\partial s_j(x)}{\partial x_k} = c$, with $c \in \mathbb{R}$ independent of x, and thus, $\mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_j}] = \mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_k}] = 0$. Regarding criterion (iii) in definition 1, we can identify them again using the score's Jacobian. In the fully connected triangle in the moral graph, two of the nodes with $\mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_j}] = \mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_i}] = \mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_k}] = \mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_l}] = 0$. The linear blocks are the ones where we cannot identify the directions due to non-identifiability in the linear Gaussian additive model. Therefore, it is necessary to remove them to further orient more directions. The terminal vertices and linear blocks are removed at the end of each iteration. In the next iteration, it will orient more edges involving the combination of linear and nonlinear causal mechanisms with Lemma 2. Removing the nonlinear terminal vertexes results in the children becoming purely linear and satisfying the condition in Lemma 2. Additionally, removing the terminal vertexes or roots does not affect the calculation of the score's Jacobian. By repeating this process, we can identify most of the directions in the graph.

4.3. Identifying V-Structure from the Jacobian of the Score function

A V-Structure represents a local probabilistic model involving three variables i, j, and k in the form of $i \rightarrow k \leftarrow j$. In this structure, vertices i and j are considered spouses as they share a common child k. Within the moral graph, only triangles can hide spouse links or V-structure. Therefore, assuming the accuracy of the moral graph identification in Section 4.1, the search can be focused solely on fully connected triangles, i.e., cliques of three nodes, by leveraging the following definition.

Definition 2. ***(Collider set)*** *Suppose that $\mathcal{G}^m = (\mathbf{V}, \mathbf{E})$ is the moral graph of the DAG representing the causal structure of a faithful dataset. Let $\mathbf{Tri}(i - j)$ represent the set of vertices forming a triangle with vertices i and j, where $i, j \in \mathbf{V}$ and $(i, j) \in \mathbf{E}$. We use $\mathbf{Tri}(X_i - X_j)$ to denote the corresponding variables for $\mathbf{Tri}(i - j)$.*

$$\mathbf{Tri}(X_i - X_j) = \{X_k \,|\, k \in \mathbf{V}, (i, k) \in \mathbf{E}, (j, k) \in \mathbf{E}\}.$$

A set of vertices $\mathbf{Z} \subseteq \mathbf{Tri}(i - j)$ then has the Collider Set property for the pair (i, j) if it is the largest set that fulfills

$$\exists \mathbf{S} \subseteq \mathbf{V} \backslash \{i, j\} \backslash \mathbf{Z} : (X_i \perp\!\!\!\perp X_j \,|\, X_{\mathbf{S}}),$$

and

$$\forall Z_i \in \mathbf{Z} : (X_i \not\!\perp\!\!\!\perp X_j \,|\, X_{\mathbf{S} \cup \{Z_i\}}).$$

The set $\mathbf{Z}$ is then a collider set for vertex pair i and j, indicating that the vertices in set $\mathbf{Z}$ are common children of the vertex pair i and j.

Lemma 5. *Let p be the joint probability density function of random variables $X \in \mathbb{R}^d$. Consider the local distribution of X_i, X_j, and $X_{\mathbf{Z}}$. Assume it is part of a larger network that $\{i, j, \mathbf{Z}\} \subseteq \mathbf{V}$, we have*

$$X_i \perp\!\!\!\perp X_j \,|\, X_{\mathbf{Z}} \quad \iff \quad \frac{\partial \log p(x_i, x_j, x_{\mathbf{Z}})}{\partial x_i \partial x_j} = 0. \tag{7}$$

The detailed proof of Lemma 5 is available in Appendix C. Utilizing Lemma 5, we can employ the Jacobian of the core function to identify collider sets between each pair of nodes and remove spouse links accordingly in the moral graph. Two considerations must be taken into account during the search for collider sets. First, there might be other active paths between vertices i and j that do not pass through any node of $\mathbf{Tri}(i-j)$, leading to a dependency between X_i and X_j. Second, the included nodes should not contain any descendant of potential colliders. Otherwise, this would result in a dependency between X_i and X_j, causing the score's Jacobian to be non-zero even when excluding the collider. Given these considerations, we propose Algorithm 5 to identify the collider sets.

Assuming consistent estimation of the Jacobian of score functions, Algorithm 5 correctly identifies all V-structures and all spouse links for faithful and causally sufficient datasets in the large sample limit.

A detailed proof is given in Appendix E. Instead of searching for the separating set as that in Pellet and Elisseeff (2008), we directly search for the collider set. Assuming the moral graph has been correctly estimated, only triangles can involve spouse links and V-structures. Hence, Algorithm 5 only iterates through fully connected triangles. For instance, considering a link $i - j$ within these triangles, the collider set can be inferred by including nodes on a path with length greater than 2 between vertices i and j and excluding nodes in $\mathbf{Tri}(i-j)$ and their descendants during the calculation of the score's Jacobian of X_i and X_j. If it is not a spouse link, the collider set $\mathbf{C}_{ij}$ remains as its default value, **null**. Otherwise, $\mathbf{C}_{ij}$ becomes the collider set for vertrices i and j. The search process offers three main benefits. Firstly, it only searches the fully connected triangles in the moral graph. Secondly, the procedure can be parallelized by allocating triangles into multiple cores in a processor or set of processors. Lastly, for each connected pair $X_i - X_j$ in a triangle, decisions regarding potential spouse links and arc orientation are made simultaneously, resulting in a more efficient process. A detailed example elucidating the identification of a collider set in Figure 4 is available in Appendix D.

After orienting all directions that can be identified with the above procedure, Meek rules (Meek, 2013) are then applied as the final step to further complete the edge orientation. The overall algorithm of the proposed methodology is provided in Appendix G.

4.4. Estimation of Score's Jacobian

In this section, we discuss the estimation for the score function $s(x) \equiv \nabla \text{log} p(x)$ and the Jacobian of the score function $\frac{\partial \log p(\mathbf{x})}{\partial x_i x_j}$ of distribution with density $p(x)$ given an i.i.d. sample $\{x^k\}_{k=1,\ldots,n}$. Our approach aligns with methods proposed in Rolland et al. (2022) and Li and Turner (2017). First, we estimate the first-order derivative of $\log p(x)$ using Stein's Identity (Stein, 1972):

$$\mathbb{E}\left[\mathbf{h}(\mathbf{x}) \nabla \log p(\mathbf{x})^T + \nabla \mathbf{h}(\mathbf{x})\right] = 0, \tag{8}$$

where $\mathbf{h} : \mathbb{R}^d \to \mathbb{R}^{d'}$ is any test function such that $\lim_{x\to\infty} \mathbf{h}(\mathbf{x}) p(\mathbf{x}) = 0$.

Following Li and Turner (2017), we present the estimator for the point-wise first-order partial derivative corresponding to Equation (8), denoted as $\mathbf{G} \equiv (\nabla \log p(x^1), \ldots, \nabla \log p(x^n))^T \in \mathbb{R}^{n \times d}$. The estimator of $\mathbf{G}$ is:

$$\hat{\mathbf{G}} = -(\mathbf{K} + \eta \mathbf{I})^{-1} \langle \nabla, \mathbf{K} \rangle ,$$

where $\mathbf{H} = (\mathbf{h}(\mathbf{x}^1), \ldots, \mathbf{h}(\mathbf{x}^{d'})) \in \mathbb{R}^{d' \times n}$, $\overline{\nabla \mathbf{h}} = \frac{1}{n} \sum_{k=1}^{n} \nabla \mathbf{h}(\mathbf{x^k})$, $\mathbf{K} = \mathbf{H}^T \mathbf{H}$, $\mathbf{K}_{ij} = \kappa(\mathbf{x^i}, \mathbf{x^j}) = \mathbf{h}(\mathbf{x}^i)^T \mathbf{h}(\mathbf{x}^j)$, $\langle \nabla, \mathbf{K} \rangle = n \mathbf{H}^T \overline{\nabla \mathbf{h}}$, $\langle \nabla, \mathbf{K} \rangle_{ij} = \sum_{k=1}^{n} \nabla_{x_j^k} \kappa(\mathbf{x^i}, \mathbf{x^j})$, where $\kappa(\cdot, \cdot)$ can be any kernel satisfying Stein's identity and $\eta \geq 0$ is a regularization parameter.

Once the first-order derivative $\nabla \log p(x)$ is estimated, we proceed to estimate the Hessian using the second-order Stein's identity:

$$\mathbb{E}\left[\mathbf{h}(\mathbf{x}) \nabla^2 \log p(\mathbf{x})^T\right] = \mathbb{E}\left[\nabla^2 \mathbf{h}(\mathbf{x}) - \mathbf{h}(\mathbf{x}) \nabla \log p(\mathbf{x}) \nabla \log p(\mathbf{x})^T\right] . \tag{9}$$

Following Rolland et al. (2022), we now present the estimator for the point-wise second-order partial derivative corresponding to Equation (9), denoted as $\mathbf{J} \equiv (\nabla^2 \log p(x^1), \ldots, \nabla^2 \log p(x^n))^T \in \mathbb{R}^{d \times n \times d}$. The estimator of $\mathbf{J}$ is:

$$\hat{\mathbf{J}} = -\hat{\mathbf{G}} \hat{\mathbf{G}}^T + (\mathbf{K} + \eta \mathbf{I})^{-1} \left\langle \nabla^2, \mathbf{K} \right\rangle ,$$

where $\mathbf{H} = (\mathbf{h}(\mathbf{x}^1), \ldots, \mathbf{h}(\mathbf{x}^n)) \in \mathbb{R}^{d' \times n}$, $\overline{\nabla^2 \mathbf{h}} = \frac{1}{n} \sum_{k=1}^{n} \nabla^2 \mathbf{h}(\mathbf{x^k})$, $(\nabla^2 \mathbf{h}(\mathbf{x}))_{ij} = \frac{\partial^2 h_i(x)}{\partial x_j^2}$, $\mathbf{K} = \mathbf{H}^T \mathbf{H}$, $\mathbf{K}_{ij} = \kappa(\mathbf{x}^i, \mathbf{x}^j) = \mathbf{h}(\mathbf{x}^i)^T \mathbf{h}(\mathbf{x}^j)$, $\left\langle \nabla^2, \mathbf{K} \right\rangle = n \mathbf{H}^T \overline{\nabla \mathbf{h}}$, $\left\langle \nabla^2, \mathbf{K} \right\rangle_{ij} = \sum_{k=1}^{n} \nabla_{x_j^k}^2 \frac{\partial^2 \kappa(\mathbf{x^i}, \mathbf{x^j})}{(\partial x_j^k)^2}$, where $\kappa(\cdot, \cdot)$ can be any kernel satisfying Stein's identity and $\eta \geq 0$ is the regularization parameter.

Choice of Kernel Estimating the score's Jacobian requires the selection of a kernel κ. The Radial Basis Function (RBF) kernel, $\kappa_s(x, y) = e^{-\frac{\|x-y\|_2^2}{2s^s}}$, is commonly used, with the bandwidth parameter s playing an important role.This bandwidth influences the smoothness of the kernel, determining how rapidly it changes and controlling the range over which a data point affects others. Typically, the median heuristic, i.e., the median of pairwise distances between vectors in X, is chosen as the bandwidth (Garreau et al., 2017). However, to mitigate the influence of data variance, we use the inverse of the standard deviation for each variable as its corresponding bandwidth. It's important to note that when using Algorithm 1 for causal discovery, the kernel lengthscale is recalculated each time nodes are removed from the data matrix. The regularization parameters are set to $1e^{-6}$, and the coefficient c is chosen to be the value that induces a kernel matrix rank almost equivalent to the number of samples.

Algorithm 1 Kernel computation in estimating the Jacobian of the score

```
Input: Observational data: $\mathbf{X} \in \mathbb{R}^{n \times d}$ , regularizer parameter $\eta > 0$, coefficient $c$
Output: Kernel matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$
for $i = 1$ to $d$ do
    $s \leftarrow \frac{c}{\text{std}(\mathbf{X}[:, i])}$                          ▷ Compute local scale
    local_kernel $\leftarrow$ RBF($\mathbf{X}[:, i]$) with local scale $s$     ▷ Compute local RBF kernel
    $\mathbf{K} \leftarrow \mathbf{K} \odot$ local_kernel                      ▷ Element-wise multiplication
end for
return K
```

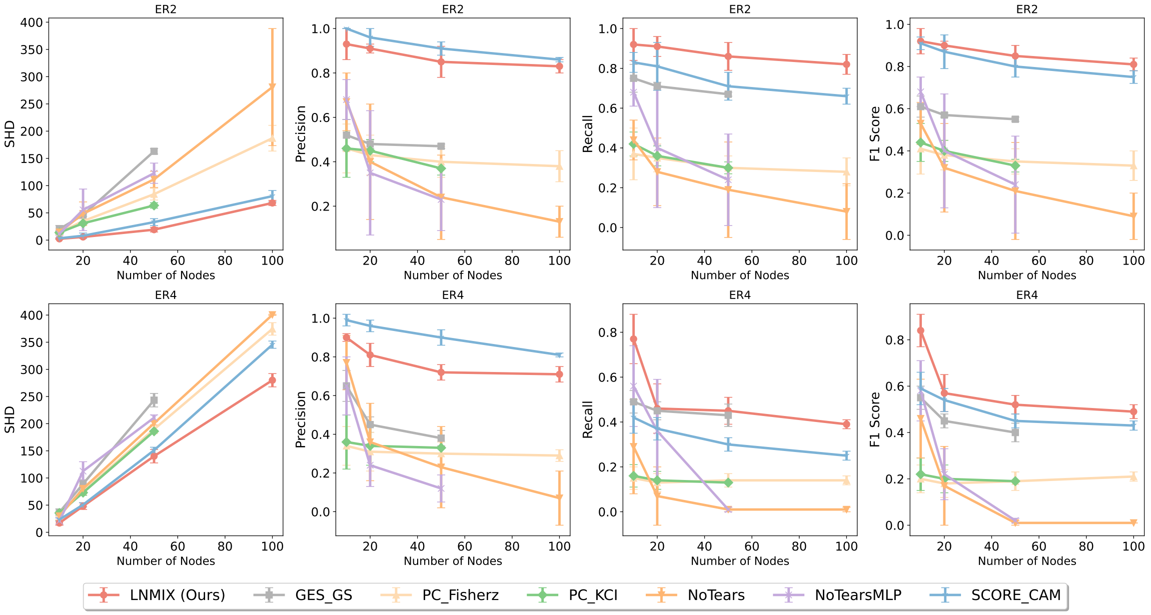


Figure 2: SHD, Precision, Recall, F1 Score v.s. number of nodes $d \in \{10, 20, 50, 100\}$ for different methods on sparse ER2 (upper) and dense ER4 (lower) graphs. For higher values of d, some methods are missing as they were too much time expensive to run. Number of observations is 1000.

5. Experiments

Setup. We generate synthetic data from SEMs based on Equation (1), employing random DAGs generated through the Erdős-Rễnyi (ER) graphical model (Erdős et al., 1960). The number of nodes d is selected from the set $\{10, 20, 50, 100\}$. Two scenarios are considered: sparse graphs ER2 and dense graphs ER4, representing graphs with $2d$ and $4d$ directed edges, respectively. The choice of the link function f_i between linear and nonlinear components is made randomly with a probability of 0.5 in the data-generating process. For linear components, edge weights in the generated DAG follow a Uniform distribution $\mathcal{U}([-1.5, -0.5], [0.5, 1.5])$, and noises are simulated from $\mathcal{N}(0, \sigma_i^2)$, where $\sigma \in [0.2, 0.5]$. Nonlinear components use link functions randomly chosen from `sin`, `cos`, `tanh`, `sigmoid`, `polynomial`, and their combinations.

Methods Considered. We compare several methods for learning DAGs: **(PC-KCI)** PC algorithm (Spirtes et al., 2000) with Kernel-based Conditional Independence test. (Zhang et al., 2012) **(PC-Fisherz)** PC algorithm with Fisher-z conditional independence test (Pearson, 1913) **(GES-GS)** Greedy Equivalence Search (GES) (Chickering, 2002) with Generalized Score (GS). (Huang et al., 2018). (**SCORE**) Topological-based algorithm (Rolland et al., 2022) with CAM pruning (Bühlmann et al., 2014). (**NoTears**) Gradient-based algorithm for linear data models with least-squares loss (Zheng et al., 2018). (**NoTears-MLP**) Gradient-based algorithm using neural network modeling for non-linear causal relationships (Zheng et al., 2020).

Results. We evaluate performance using Structural Hamming Distance (SHD), precision, recall, F1 score (Figure 2), along with run time in seconds (Figure 3). The results are averaged over 10 trials with different random seeds. As the number of variables increases, some methods experience failure due to timeouts (>1 day) or memory issues. Comparative results with varying numbers of variables and a sample size of 1000 show that our approach, LNMIX, along with SCORE, consistently outperforms other methods. An illustrative example is provided in Figure 1, LNMIX successfully recovers the ground truth DAG by leveraging the score's Jacobian structure. In contrast, constraint-based, score-based, and optimization-based methods only manage to recover up to their equivalence class. LNMIX consistently performs better than SCORE across different numbers of variables in data generated through both ER2 and ER4 graphs. Although the performance gap between LNMIX and SCORE in ER2 diminishes compared to ER4, potentially due to the reduced chance of a mixture model when fewer edges are presented. Furthermore, experimental results suggest some robustness in SCORE, allowing it to handle linear nodes with children, contrary to theoretical expectations of a zero variance in the score's Jacobian for linear nodes with children. Nonetheless, LNMIX has a better recall in both ER2 and ER4 graphs, which means that our algorithm is able to identify a greater number of edges presented in the ground truth DAG. The robust performance of LNMIX not only underscores the efficacy of our moral graph but also establishes a solid foundation for subsequent steps.

The overall time complexity of our algorithm is $\mathcal{O}(d^2n^3 + 3^2m(d + e))$, considering the estimation of the score's Jacobian involving inverting d^2 kernel matrices of size $n \times n$. Nonetheless, the computation of the score's Jacobian for each node can be parallelized across multiple processors. This parallelization brings about a substantial reduction in the overall processing time, making it applicable even to larger graphs. The complexity of iterating over all triangles is $\mathcal{O}(3^2m)$, where m is the number of triangles in the moral graph and the worst case time complexity for finding all paths between two vertices is $\mathcal{O}(d+e)$ using Depth First Search (DFS) traversal.

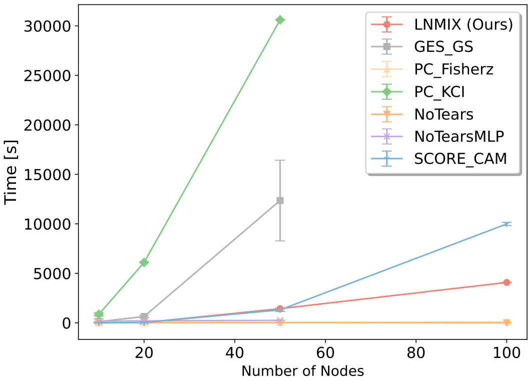


Figure 3: Run-time comparison on ER4.

However, the search for v-structures hidden in these triangles can also be parallelized by allocating triangles to multiple processors. Moreover, we orient most of the edges in section 4.2, and thus make this more efficient. The constraint-based and score-based approaches raise an NP-hard problem that the complexity of the PC algorithm and GES is exponential to the number of nodes. Continuous-based algorithms NoTears and NoTears-MLP require less time than constraint-based and search-based methods, but their performance degrades with an increasing number of nodes. SCORE is a more scalable causal discovery approach, but its final complexity is $\mathcal{O}(dn^3 + dr(n, d))$ due to the bottleneck pruning step where $r(n, d)$ is the complexity of fitting a generalized additive model using n data points in d dimensions and amounting to $\mathcal{O}(nd^2)$ using Iteratively Reweighted Least Squares (Minka, 2003).

6. Conclusion

In our study, we demonstrate the theoretical recovery of an accurate causal graph under an ANM with Gaussian noise that incorporates both linear and nonlinear causal mechanisms from observational data. Our approach utilizes the score function's Jacobian to initially identify the moral graph of variables and reveal the direction of a significant portion of edges. The methodology proves effective in scenarios characterized by a mixture of linear and nonlinear components, as well as involving purely nonlinear relationships. We also identify V-structures by examining the Jacobian of the score on local distribution. However, scalability challenges arise, especially in dense graphs with numerous spouse links. This issue is addressed by parallelizing the resolution of V-structures. Our analysis resulted in an algorithm that significantly accelerates practical applications compared to established causal discovery algorithms, maintaining a comparable level of accuracy. It's important to note, however, that our scalability experiments were limited to synthetic data, and our model currently assumes additive noise. Consequently, we aim to broaden our methodology to include input variables beyond our current model's scope, emphasizing the exploration of nonparametric models and addressing latent confounders.

Acknowledgments

This research was undertaken using the LIEF HPC-GPGPU Facility hosted at the University of Melbourne. This Facility was established with the assistance of LIEF Grant LE170100200. WL was supported by Melbourne Research Scholarship from the University of Melbourne. EG was supported by an Australian Government Research Training Program (RTP) Scholarship. MG was supported by ARC DE210101624. HB was supported by ARC FT190100374.

References

Lazar Atanackovic, Alexander Tong, Jason Hartford, Leo J Lee, Bo Wang, and Yoshua Bengio. Dyngfn: Bayesian dynamic causal discovery using generative flow networks. *arXiv preprint arXiv:2302.04178*, 2023.

Alessandro Barp, Francois-Xavier Briol, Andrew Duncan, Mark Girolami, and Lester Mackey. Minimum stein discrepancy estimators. *Advances in Neural Information Processing Systems*, 32, 2019.

Peter Bühlmann, Jonas Peters, and Jan Ernest. Cam: Causal additive models, high-dimensional order search and penalized regression. 2014.

David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of machine learning research*, 3(Nov):507–554, 2002.

Paul Erdős, Alfréd Rényi, et al. On the evolution of random graphs. *Publ. math. inst. hung. acad. sci*, 5(1):17–60, 1960.

Damien Garreau, Wittawat Jitkrittum, and Motonobu Kanagawa. Large sample analysis of the median heuristic. *arXiv preprint arXiv:1707.07269*, 2017.

Asish Ghoshal and Jean Honorio. Learning linear structural equation models in polynomial time and sample complexity. In *International Conference on Artificial Intelligence and Statistics*, pages 1466–1475. PMLR, 2018.

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524, 2019.

Luigi Gresele, Julius Von Kügelgen, Vincent Stimper, Bernhard Schölkopf, and Michel Besserve. Independent mechanism analysis, a new concept? *Advances in neural information processing systems*, 34:28233–28248, 2021.

Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. *Advances in neural information processing systems*, 21, 2008.

Biwei Huang, Kun Zhang, Yizhu Lin, Bernhard Schölkopf, and Clark Glymour. Generalized score functions for causal discovery. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1551–1560, 2018.

Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

Sébastien Lachapelle, Philippe Brouillard, Tristan Deleu, and Simon Lacoste-Julien. Gradient-based neural dag learning. *arXiv preprint arXiv:1906.02226*, 2019.

Yingzhen Li and Richard E Turner. Gradient estimators for implicit models. *arXiv preprint arXiv:1705.07107*, 2017.

Po-Ling Loh and Peter Bühlmann. High-dimensional learning of linear causal networks via inverse covariance estimation. *The Journal of Machine Learning Research*, 15(1):3065–3105, 2014.

Christopher Meek. Causal inference and causal explanation with background knowledge. *arXiv preprint arXiv:1302.4972*, 2013.

Thomas P Minka. A comparison of numerical optimizers for logistic regression. *Unpublished draft*, pages 1–18, 2003.

Francesco Montagna, Nicoletta Noceti, Lorenzo Rosasco, Kun Zhang, and Francesco Locatello. Causal discovery with score matching on additive models with arbitrary noise. *arXiv preprint arXiv:2304.03265*, 2023a.

Francesco Montagna, Nicoletta Noceti, Lorenzo Rosasco, Kun Zhang, and Francesco Locatello. Scalable causal discovery with score matching. *arXiv preprint arXiv:2304.03382*, 2023b.

Joris Mooij, Dominik Janzing, Jonas Peters, and Bernhard Schölkopf. Regression by dependence minimization and its application to causal inference in additive noise models. In *Proceedings of the 26th annual international conference on machine learning*, pages 745–752, 2009.

Judea Pearl et al. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19(2):3, 2000.

Karl Pearson. On the probable error of a coefficient of correlation as found from a fourfold table. *Biometrika*, 9(1/2):22–33, 1913. ISSN 00063444. URL http://www.jstor.org/stable/2331798.

Jean-Philippe Pellet and André Elisseeff. Using markov blankets for causal structure learning. *Journal of Machine Learning Research*, 9(7), 2008.

Jonas Peters and Peter Bühlmann. Identifiability of gaussian structural equation models with equal error variances. *Biometrika*, 101(1):219–228, 2014.

Jonas Peters, Joris Mooij, Dominik Janzing, and Bernhard Schölkopf. Identifiability of causal graphs using functional models. *arXiv preprint arXiv:1202.3757*, 2012.

Jonas Peters, Joris M Mooij, Dominik Janzing, and Bernhard Schölkopf. Causal discovery with continuous additive noise models. *Journal of Machine Learning Research*, 15(58):2009–2053, 2014.

Patrik Reizinger, Yash Sharma, Matthias Bethge, Bernhard Schölkopf, Ferenc Huszár, and Wieland Brendel. Jacobian-based causal discovery with nonlinear ica. *Transactions on Machine Learning Research*, 2022.

Paul Rolland, Volkan Cevher, Matthäus Kleindessner, Chris Russell, Dominik Janzing, Bernhard Schölkopf, and Francesco Locatello. Score matching enables causal discovery of nonlinear additive noise models. In *International Conference on Machine Learning*, pages 18741–18753. PMLR, 2022.

Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721): 523–529, 2005.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

Rajen D Shah and Jonas Peters. The hardness of conditional independence testing and the generalised covariance measure. 2020.

Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, Antti Kerminen, and Michael Jordan. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10), 2006.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Peter Spirtes. An anytime algorithm for causal inference. In *International Workshop on Artificial Intelligence and Statistics*, pages 278–285. PMLR, 2001.

Peter Spirtes and Clark Glymour. An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, 9(1):62–72, 1991.

Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2000.

Charles Stein. A bound for the error in the normal approximation to the distribution of a sum of dependent random variables. In *Proc. Sixth Berkeley Symp. Math. Stat. Prob.*, pages 583–602, 1972.

Kun Zhang and Aapo Hyvarinen. On the identifiability of the post-nonlinear causal model. *arXiv preprint arXiv:1205.2599*, 2012.

Kun Zhang, Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. Kernel-based conditional independence test and application in causal discovery. *arXiv preprint arXiv:1202.3775*, 2012.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31, 2018.

Xun Zheng, Chen Dan, Bryon Aragam, Pradeep Ravikumar, and Eric Xing. Learning sparse nonparametric dags. In *International Conference on Artificial Intelligence and Statistics*, pages 3414–3425. PMLR, 2020.

Yujia Zheng, Ignavier Ng, Yewen Fan, and Kun Zhang. Generalized precision matrix for scalable estimation of nonparametric markov networks. *arXiv preprint arXiv:2305.11379*, 2023.

Appendix A. Proof of Lemma 1

Proof. **(Sufficient Condition).** Consider a random variable X in the observational data. The score function can be expressed as:

$$s_i(x) = -\frac{x_i - f_i(x_{\mathrm{PA}_i})}{\sigma_i^2} + \sum_{j \in \mathrm{CH}_i} \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_i} \left(\frac{x_j - f_j(x_{\mathrm{PA}_j})}{\sigma_j^2} \right). \tag{10}$$

Here, when $k \in \mathrm{MB}(i)$, k can be a parent, a child, or a spouse node of node i.
When $k \in \mathrm{PA}_i$, Equation (10) becomes

$$\frac{\partial s_i(x)}{\partial x_k} = \frac{1}{{\sigma_i}^2} \frac{\partial f_i(x_{\mathrm{PA}_i})}{\partial x_k} + \sum_{j \in \{\mathrm{CH}_i \cap \mathrm{CH}_k\}} \frac{\partial}{\partial x_k} \left\{ \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_j} \frac{x_j - f_j(x_{\mathrm{PA}_j})}{\sigma_j^2} \right\}.$$

When $k \in \mathrm{CH}_i$, Equation (10) becomes

$$\frac{\partial s_i(x)}{\partial x_k} = \frac{1}{\sigma_k^2} \frac{\partial f_k(x_{\mathrm{PA}_k})}{\partial x_i}.$$

When $k \in \mathrm{SP}_i$, Equation (10) becomes

$$\frac{\partial s_i(x)}{\partial x_k} = -\sum_{j \in \{\mathrm{CH}_i \cap \mathrm{CH}_k\}} \frac{1}{\sigma_j^2} \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_i} \left(\frac{\partial f_j(x_{\{k\} \cup \mathrm{PA}_j \setminus \{k\}})}{\partial x_k} \right).$$

Hence, $\frac{\partial s_i(x)}{\partial x_k} \neq 0$ if $k \in \mathrm{MB}(i)$. When $\frac{\partial s_i(x)}{\partial x_j} = 0$, we have $j \notin \mathrm{MB}(i)$.
(Necessary Condition). Suppose we have $j \notin \mathrm{MB}(i)$. Then

$$X_i \perp\!\!\!\perp X_j \mid X_{\mathrm{MB}(i)}. \tag{11}$$

This follows that the joint probability density of $\mathbf{V}$ with respect to a tensor product Lebesgue measure must factor as

$$p(x) = \prod_{j \in \mathbf{V} \setminus \{i\} \setminus \mathrm{MB}(i)} p(x_i \mid x_{\mathrm{MB}(i)}) p(x_j \mid x_{\mathrm{MB}(i)}) p(x_{\mathrm{MB}(i)}). \tag{12}$$

Therefore, it follows from Eq.(12) that $\frac{\partial^2 \log p(x)}{\partial x_i x_j} = \frac{\partial s_i(x)}{\partial x_j} = 0$. □

Appendix B. Proof of Lemma 2

Proof. **(Sufficient Condition)**. Considering the SEMs outlined above, the score function of X_j is as follows,

$$s_j(x) = -\frac{x_j - f_j(x_{\mathrm{PA}_j})}{\sigma_j^2} + \sum_{i \in \mathrm{CH}_j} \frac{\partial f_i(x_{\mathrm{PA}_i})}{\partial x_j} \frac{x_i - f_i(x_{\mathrm{PA}_i})}{\sigma_i^2}. \tag{13}$$

Since f_j is a nonlinear function of the parents of node j. $\forall k \in \mathrm{PA}_j$,

$$\frac{\partial s_j(x)}{\partial x_k} = \frac{1}{{\sigma_j}^2} \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_k} - \sum_{i \in \{\mathrm{CH}_j \cap \mathrm{CH}_k\}} \frac{\partial}{\partial x_k} \left\{ \frac{\partial f_i(x_{\mathrm{PA}_i})}{\partial x_j} \frac{f_i(x_{\mathrm{PA}_i})}{\sigma_i^2} \right\}.$$

Note that, we do not exclude the case that node k can also be a spouse of node i. Since f_j is a nonlinear function and all children of node j are linear functions of X_j and their associated parents which means that f_i are linear functions, we have

$$\frac{\partial s_j(x)}{\partial x_k} = \frac{1}{{\sigma_j}^2} \frac{\partial f_j(x_{\mathrm{PA}_j})}{\partial x_k} + c, \tag{14}$$

where the second term on the right-hand side of the equation (14) is a constant that does not depend on any variable. Since $f_j(x_{PAj})$ is a nonlinear function of x_k, $\frac{\partial s_j(x)}{\partial x_k}$ depends on x_k and hence, $\mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_k}] \neq 0$.

Since all the children are linear functions of their parents, $\forall k \in \mathrm{PA}_j$,

$$\frac{\partial s_j(x)}{\partial x_k} = \frac{1}{\sigma_k^2} \frac{\partial f_k(x_{\mathrm{PA}_k})}{\partial x_j} - \sum_{i \in \{\mathrm{CH}_j \cap \mathrm{CH}_k\}} \frac{\partial}{\partial x_k} \frac{\partial f_i(x_{\mathrm{PA}_i})}{\partial x_j} \frac{f_i(x_{\mathrm{PA}_i})}{\sigma_i^2}. \tag{15}$$

Note that, we do not exclude the case that node k can also be a spouse of node i. Since all children are linear functions of their parents, we have

$$\frac{\partial s_j(x)}{\partial x_k} = c, \tag{16}$$

where $c \in \mathbb{R}$ is a constant that does not depend on any variable and hence, $\mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_k}] = 0$.

For the node itself, since all children are linear functions of X_j and their parents,

$$\frac{\partial s_j(x)}{\partial x_j} = -\frac{1}{\sigma_j^2} - \sum_{i \in \mathrm{CH}_j} \frac{1}{\sigma_i^2} \left(\frac{\partial f_i(x_{\mathrm{PA}_i})}{\partial x_j} \right)^2. \tag{17}$$

Since $f_i(x_{PAi})$ is a linear function of its parents, the derivative with respect to x_j will be a constant that does not depend on any variable and hence, $\mathrm{Var}_x[\frac{\partial s_j(x)}{\partial x_k}] = 0$. If f_j is a nonlinear function of all parents of node j, and all children of node j are linear functions of X_j and their associated parents, then for the non-zero entries in the j-th row of the score matrix, in the j-th row of the score's Jacobian matrix, the variance of j-th entry and the entries corresponding to children of node j will be zero while the variance of the entries corresponding to parents of node j will be nonzero.

(Necessary Condition). We prove this by contradiction. Suppose that the children of X_j are nonlinear functions of their parents and that $\forall x$, $\frac{\partial s_j(x)}{\partial x_j} = c$.

$$s_j(x) = cx_j + h(x_{-j}), \tag{18}$$

where $h(x_{-j})$ can depend on any variables but x_j. Let j_c be a child of j and f_{j_c} be a nonlinear function of x_j. Equation (13) can be written as

$$\begin{aligned} s_j(x) = &-\frac{x_j - f_j(x_{PA_j})}{\sigma_j^2} + \frac{\partial f_{i_c}(x_{\mathrm{PA}_{i_c}})}{\partial x_j}\frac{x_{i_c} - f_{i_c}(x_{\mathrm{PA}_{i_c}})}{\sigma_{i_c}^2} \\ &+ \sum_{i \in ch(j), i \neq i_c} \frac{\partial f_i(x_{\mathrm{PA}_i})}{\partial x_j}\frac{x_i - f_i(x_{\mathrm{PA}_i})}{\sigma_i^2}. \end{aligned} \tag{19}$$

Since equation (17) and equation (19) are equivalent, we can get,

$$\begin{aligned} \frac{\partial f_{i_c}(x_{\mathrm{PA}_{i_c}})}{\partial x_j}\frac{x_{i_c} - f_{i_c}(x_{\mathrm{PA}_{i_c}})}{\sigma_{i_c}^2} - h(x_{-j}) = &(c + \frac{1}{\sigma_j^2})x_j + \frac{f_j(x_{\mathrm{PA}_j})}{\sigma_j^2} \\ &- \sum_{i \in \mathrm{CH}_j, i \neq i_c} \frac{\partial f_i(x_{\mathrm{PA}_i})}{\partial x_j}\frac{x_i - f_i(x_{\mathrm{PA}_i})}{\sigma_i^2}. \end{aligned} \tag{20}$$

Since the right hand side of equation (20) does not depend on variable x_{i_c}, taking the differentiation with respect to x_{i_c}, we get

$$\frac{\partial}{\partial x_{i_c}}\left[\frac{\partial f_{i_c}(x_{\mathrm{PA}_{i_c}})}{\partial x_j}\frac{x_{i_c} - f_{i_c}(x_{\mathrm{PA}_{i_c}})}{\sigma_{i_c}^2} - h(x_{-j})\right] = 0.$$

$$\frac{\partial f_{i_c}(x_{\mathrm{PA}_{i_c}})}{\partial x_j} = \sigma_{i_c}^2 \frac{h(x_{-j})}{\partial x_{i_c}}.$$

Since $h(x_{-j})$ does not depend on x_j, $\frac{\partial f_{i_c}(x_{\mathrm{PA}_{i_c}})}{\partial x_j}$ does not depend on x_j. It means that $f_{i_c}(x_{\mathrm{PA}i_c})$ is a linear function of x_j which contradict the assumption that f_{i_c} is a nonlinear function of x_j.

Moreover, to demonstrate that the structure of the Jacobian of the score functions corresponds exclusively to the SCMs outlined in Lemma 2, we consider three scenarios: (1) f_j is a linear function of all parents of node j and all children of node j are linear functions of X_j and their associated parents. For this case, all entries in the j-th row of the score's Jacobian are constant, resulting in zero variances. This is due to the linearity of all functions in $s_j(x)$, and their derivatives of these functions being constant. (2) f_j is a nonlinear function of all parents of node j and all children of node j are nonlinear functions of X_j and their associated parents. As proven in lemma 3, the j-th entry of the score's Jacobian in the j-th row, i.e. diagonal, is constant only if node j is a terminal vertex. However, since it has children that are nonlinear functions of it, the j-th diagonal term of the score's Jacobian is not constant. (3) f_j is a linear function of all parents of node j while all children of node j are nonlinear functions of X_j and their associated parents. For (1), all entries in j-th row of the score's Jacobian are constant and hence the variances are zero. Similar to scenario (2), the score's Jacobian of the j-th diagonal term is non-constant as the children are nonlinear functions of it. We systematically enumerate all scenarios in which a mixture model could be, establishing that only the one outlined in Lemma 2 yields the corresponding score's Jacobian structure. □

Appendix C. Proof of Lemma 5

Proof. **Sufficient Condition**. Suppose we have $X_i \perp\!\!\!\perp X_j \mid X_{\mathbf{Z}}$,

$$\begin{aligned}\frac{\partial \log p(x_i, x_j, x_{\mathbf{z}})}{\partial x_i \partial x_j} &= \frac{\partial \log p(x_i \mid x_{\mathbf{z}}) p(x_j \mid x_{\mathbf{z}}) p(x_{\mathbf{z}})}{\partial x_i \partial x_j} \\ &= \frac{\partial}{\partial x_j}\left\{\frac{\partial}{\partial x_i}\left[\log p(x_i \mid x_{\mathbf{z}}) + \log p(x_j \mid x_{\mathbf{z}}) + \log p(x_{\mathbf{z}})\right]\right\} \\ &= \frac{\partial}{\partial x_j} h(x_{1,\ldots,j-1}, x_{j+1,\ldots,d}) \\ &= 0.\end{aligned}$$

since $h : \mathcal{R}^{n-1} \to \mathcal{R}$ is a function does not depend on x_j.

Necessary Condition. By taking integration on both sides with respect to x_j in the right-hand side of equation (7), we have

$$\int \frac{\partial \log p(x_i, x_j, x_{\mathbf{z}})}{\partial x_i \partial x_j} dx_j = \frac{\partial \log p(x_i, x_j, x_{\mathbf{z}})}{\partial x_i} = f(x_i, x_{\mathbf{z}}), \tag{21}$$

where $f : \mathcal{R}^{(d+1)} \to \mathcal{R}$ is a function that does not depend on x_j. By further taking the integration on both sides of the equation (21) with respect to x_i, we have

$$\begin{aligned}\log p(x_i, x_j, z) &= \int \frac{\partial \log p(x_i, x_j, x_{\mathbf{z}})}{\partial x_i} dx_i \\ &= f(x_i, x_{\mathbf{z}}) + g(x_j, x_{\mathbf{z}}).\end{aligned}$$

where $h, g : \mathcal{R}^{d+1} \to \mathcal{R}$ are functions that do not depend on x_j and x_i, respectively. Therefore, this follows that $X_i \perp\!\!\!\perp X_j | X_{\mathbf{Z}}$ and there is no edge between vertices i and j in the graph. □

Appendix D. An illustrative example of Algorithm 5

This procedure is best illustrated with a graphical example. Consider the sample local structure in Figure 4 and assume it is part of a larger network, We aim to find a collider set for vertices i and j, and $\{z\}$ is such a set in the ground truth DAG in Figure 4a. To see how the algorithm identifies it, the search procedure starts at line 2 in Algorithm 5 given the moral graph in Figure 4b. We have

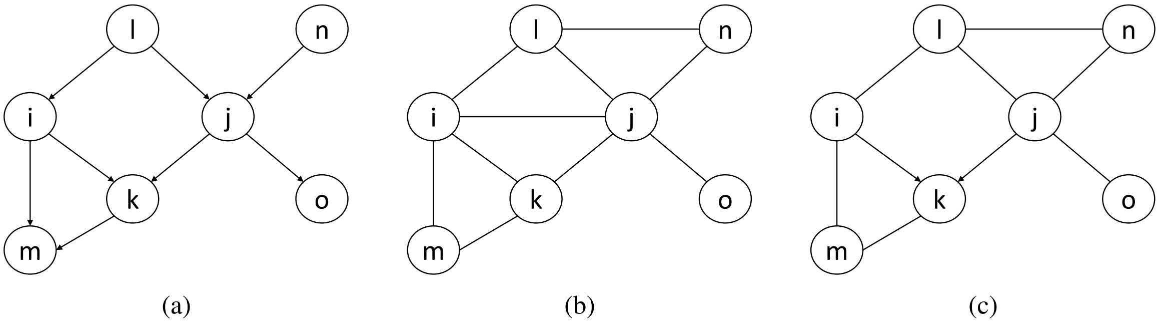


Figure 4: Sample local causal structure (a) and its corresponding moral graph (b). Collider set search in Algorithm 5 for the linked pair $i - j$ yields spouse link and orientation information (c)

$\mathbf{Tri}(i - j) = \{l, k\}$ which is the possible collider set of i and j and paths with length greater than 2 between i and j in the moral graph are $\mathbf{paths}(i - j) = \{i - l - n - j, i - m - k - j\}$. Iterating over subsets in $\mathbf{Tri(i - j)}$, for instance, starting from the potential collider set $\{k\}$, the score of the distribution will be calculated in terms of variables $\{X_i, X_l, X_j, X_m, X_n\}$ which include the set of vertices $\mathbf{path(i - j)}$ but excluding the possible collider set $\{k\}$. However, the score's Jacobian of X_i and X_j will still not be zero because $\{m\}$ is a descendant of the collider k. In the nested loop at lines 11 to 19. As we perform an extra search on the possible descendants of k, eventually we will exclude m in addition to k. This will lead to a zero in the score's Jacobian of X_i and X_j and in turn, allow us to identify the link $i - j$ as a spouse link and orient the arcs in Figure 4c.

Appendix E. Proof of Theorem 4.3

In a moral graph, a node X_i is connected to its parents, children, and spouses. For instance, vertices i and j are spouses and they have at least one common child, i.e. $\mathbf{Z}$. The common children $\mathbf{Z}$ is called a collider set of i and j. Furthermore, if vertices i and j are non-adjacent in the full graph, then $\mathbf{Z}$ are unshielded colliders for i and j such that $X_i \not\perp\!\!\!\perp X_j \mid X_{\mathbf{Z}}$ and $X_i \perp\!\!\!\perp X_j$. Considering the local causal structure $\{i, j, \mathbf{Z}\}$, by removing all common children Z, the score's Jacobian of X_i and X_j in the local structure will be zero according to the lemma 5. The spouse link $X_i - X_j$ can then be removed, and for each $Z \in \mathbf{Z}$, we orient the triplet as $i \rightarrow Z \leftarrow j$. This is equivalent to searching for the d-separating set $\mathbf{S}$ for vertices i and j where $X_i \perp\!\!\!\perp X_j \mid X_{\mathbf{S}}$ (Pellet and Elisseeff, 2008; Pearl et al., 2000), whereas we are in a reverse direction, directly searching for the collider set. The proof boils down to providing that the proposed search procedure always identifies the collider set for i and j when there is one.

If all colliders of i and j are unshielded colliders, then the link $i-j$ is a spouse link by definition of a moral graph, which implies that i and j have a non-empty set of common children $\mathbf{Z}$. Each Z is linked to both i and j and is thus in $\mathbf{Tri}(i - j)$ by definition 2. Consider the local causal structure $\{i, j, \mathbf{Z}, \mathbf{S}\}$ where $\mathbf{S}$ are the nodes on a path of length 2 between i and j. Assume that all colliders of i and j are unshielded colliders, including one of the unshielded collider or its descendant will yield non-zero in the score's Jacobian of X_i and X_j in the local distribution. In algorithm 5, all possible colliders and descendants of current conjectured colliders undergo a subset search in line 5 and 13, such that there will always be one iteration where all colliders and their descendants will be left out of the local distribution in calculating the score's Jacobian.

Appendix F. Experiments

F.1. Synthetic data

In this section, we present the results of experiments conducted with LNMIX using synthetic data, as discussed in section 5. The data was generated by creating DAGs through Erdős-Rényi (ER) graphical models. Our experiments include scenarios involving both sparser ER2 graphs (refer to Table 1) as well as denser ER4 graphs (refer to Table 2).

Table 1: Experiments on ER2 data.

	Method	SHD	Precision	Recall	F1	Time [s]
d=10	LNMIX (Ours)	**2.36 ± 1.20**	0.93 ± 0.07	**0.92 ± 0.08**	**0.92 ± 0.06**	4.19 ± 0.23
	GES-GS	20.33 ± 6.60	0.52 ± 0.00	0.75 ± 0.07	0.61 ± 0.05	90.94 ± 8.13
	PC-Fisherz	13.80 ± 2.93	0.43 ± 0.11	0.37 ± 0.13	0.41 ± 0.12	**0.05 ± 0.02**
	PC-KCI	13.80 ± 2.14	0.46 ± 0.13	0.42 ± 0.06	0.40 ± 0.09	226.57 ± 89.00
	NoTears	14.62 ± 2.78	0.67 ± 0.13	0.44 ± 0.10	0.53 ± 0.10	1.33 ± 2.01
	NoTearsMLP	11.75 ± 2.63	0.68 ± 0.09	0.68 ± 0.07	0.68 ± 0.07	28.46 ± 65.38
	SCORE-CAM	3.33 ± 0.94	**0.99 ± 0.04**	0.83 ± 0.05	0.91 ± 0.03	7.96 ± 0.15
d=20	LNMIX (Ours)	**5.79 ± 1.21**	0.91 ± 0.02	**0.91 ± 0.05**	**0.90 ± 0.02**	26.51 ± 15.44
	GES-GS	46.0 ± 2.16	0.48 ± 0.07	0.71 ± 0.13	0.57 ± 0.01	426.25 ± 35.03
	PC-Fisherz	34.9 ± 4.09	0.45 ± 0.09	0.36 ± 0.02	0.38 ± 0.07	**1.18 ± 0.05**
	PC-KCI	45.80 ± 1.72	0.45 ± 0.05	0.30 ± 0.05	0.33 ± 0.05	926.22 ± 299.42
	NoTears	48.38 ± 21.74	0.40 ± 0.26	0.28 ± 0.17	0.32 ± 0.21	11.67 ± 33.84
	NoTearsMLP	55.5 ± 38.24	0.35 ± 0.28	0.40 ± 0.30	0.40 ± 0.27	149.00 ± 279.87
	SCORE-CAM	8.00 ± 4.97	**0.96 ± 0.04**	0.81 ± 0.12	0.87 ± 0.08	32.90 ± 2.34
d=50	LNMIX (Ours)	**22.16 ± 4.3**	0.85 ± 0.02	**0.86 ± 0.04**	**0.85 ± 0.07**	760.4 ± 1.18
	GES-GS	163.0 ± 4.97	0.47 ± 0.01	0.67 ± 0.01	0.55 ± 0.01	5180.56 ± 209.76
	PC-Fisherz	84.03 ± 10.75	0.40 ± 0.07	0.30 ± 0.06	0.35 ± 0.02	**1.69 ± 0.52**
	PC-KCI	63.60 ± 5.00	0.37 ± 0.03	0.30 ± 0.03	0.33 ± 0.03	10017.13 ± 383.47
	NoTears	111.50 ± 14.82	0.24 ± 0.19	0.19 ± 0.24	0.21 ± 0.23	34.76 ± 46.86
	NoTearsMLP	122.75 ± 18.59	0.23 ± 0.14	0.24 ±0.23	0.27 ± 0.17	212.03 ± 340.12
	SCORE-CAM	33.00 ± 6.38	**0.91 ± 0.03**	0.71 ± 0.07	0.80 ± 0.05	1272.88 ± 126.64
d=100	LNMIX (Ours)	**68.16 ± 4.74**	0.83 ± 0.03	**0.81 ± 0.03**	**0.82 ± 0.05**	1405.85 ± 465.28
	GES-GS	–	–	–	–	–
	PC-Fisherz	187.00 ± 23.62	0.33 ± 0.07	0.28 ± 0.07	0.28 ± 0.07	**5.22 ± 1.90**
	PC-KCI	–	–	–	–	–
	NoTears	280.88 ± 107.44	0.13 ± 0.07	0.09 ± 0.11	0.08 ± 0.14	96.73 ± 70.72
	NoTearsMLP	–	–	–	–	–
	SCORE-CAM	80.67 ± 10.40	**0.86 ± 0.01**	0.66 ± 0.04	0.75 ± 0.03	4818.33 ± 231.9

F.2. Real data

We conducted a comparative analysis of algorithms using a well-known real-world dataset provided by Sachs et al. (2005) for causal discovery with 11 nodes, 17 edges, and 853 observations. Our method demonstrated a slightly better performance on the this dataset, achieving a Structural Hamming Distance (SHD) of 11, outperforming SCORE, DAS, and CAM, which has SHD of 12, and GraN-DAG, which has SHD of 13. Examining the resulting DAG, we observed some edges that

Table 2: Experiments on ER4 data.

	Method	SHD	Precision	Recall	F1	Time [s]
d=10	LNMIX (Ours)	**17.01 ± 4.03**	0.90 ± 0.02	**0.77 ± 0.11**	**0.84 ± 0.07**	15.51 ± 7.33
	GES-GS	36.76 ± 6.42	0.65 ± 0.08	0.49 ± 0.05	0.55 ± 0.05	117.78 ± 9.23
	PC-Fisherz	34.96 ± 2.14	0.34 ± 0.10	0.15 ± 0.05	0.20 ± 0.06	**0.17 ± 0.03**
	PC-KCI	35.5 ± 2.08	0.34 ± 0.14	0.13 ± 0.05	0.19 ± 0.07	856.64 ± 176.99
	NoTears	29.94 ± 7.36	0.77 ± 0.14	0.29 ± 0.21	0.46 ± 0.17	19.8 ± 32.02
	NoTearsMLP	20.75 ± 6.32	0.65 ± 0.15	0.56 ± 0.18	0.58 ± 0.13	181.23 ± 223.66
	SCORE-CAM	23.36 ± 2.96	**0.99 ± 0.03**	0.42 ± 0.07	0.59 ± 0.07	6.14 ± 0.27
d=20	LNMIX (Ours)	**48.14 ± 6.15**	0.81 ± 0.06	**0.46 ± 0.11**	**0.57 ± 0.08**	28.45 ± 1.99
	GES-GS	89.10 ± 7.55	0.45 ± 0.04	0.47 ± 0.04	0.45 ± 0.03	641.79 ± 133.03
	PC-Fisherz	75.26 ± 4.80	0.31 ± 0.10	0.13 ± 0.04	0.18 ± 0.06	**1.58 ± 0.67**
	PC-KCI	73.00 ± 5.00	0.33 ± 0.10	0.14 ± 0.04	0.20 ± 0.06	6101.58 ± 19.77
	NoTears	80.44 ± 5.20	0.36 ± 0.20	0.07 ± 0.13	0.17 ± 0.17	22.94 ± 65.59
	NoTearsMLP	112.62 ± 17.33	0.24 ± 0.11	0.36 ± 0.23	0.22 ± 0.11	198.75 ± 337.96
	SCORE-CAM	50.66 ± 3.77	**0.96 ± 0.03**	0.37 ± 0.05	0.54 ± 0.05	32.94 ± 2.31
d=50	LNMIX (Ours)	**142.08 ± 12.35**	0.72 ± 0.04	**0.45 ± 0.06**	**0.52 ± 0.04**	1431.51 ± 21.45
	GES-GS	243.25 ± 12.40	0.38 ± 0.04	0.43 ± 0.05	0.40 ± 0.04	12353.43 ± 4078.12
	PC-Fisherz	189.70 ± 8.03	0.30 ± 0.06	0.14 ± 0.03	0.19 ± 0.04	**15.93 ± 6.08**
	PC-KCI	186.00 ± 0.00	0.36 ± 0.00	0.16 ± 0.00	0.22 ± 0.00	30604.94 ± 0.00
	NoTears	200.60 ± 3.03	0.23 ± 0.21	0.01 ± 0.00	0.01 ± 0.01	28.33 ± 54.51
	NoTearsMLP	251.00 ± 5.20	0.21 ± 0.07	0.01 ± 0.01	0.02 ± 0.01	253.06 ± 216.46
	SCORE-CAM	151.00 ± 5.61	**0.89 ± 0.04**	0.30 ± 0.03	0.44 ± 0.03	1308.91 ± 167.01
d=100	LNMIX (Ours)	**286.00 ± 12.22**	0.71 ± 0.04	**0.39 ± 0.02**	**0.49 ± 0.03**	4080.01 ± 78.12
	GES-GS	–	–	–	–	–
	PC-Fisherz	374.50 ± 11.49	0.16 ± 0.03	0.10 ± 0.02	0.13 ± 0.02	**94.95 ± 24.52**
	PC-KCI	–	–	–	–	–
	NoTears	400.44 ± 0.98	0.07 ± 0.14	0.00 ± 0.00	0.00 ± 0.00	421.69 ± 472.38
	NoTearsMLP	–	–	–	–	–
	SCORE-CAM	325.46 ± 6.68	**0.81 ± 0.01**	0.25 ± 0.02	0.43 ± 0.02	9987.87 ± 160.70

were either identified or missed by different approaches. An illustrative instance is the correct identification of the edge in our method, which was not captured by SCORE, DAS, and CAM. Plotting the data revealed a nonlinear relationship between variable 1 and variable 5, while variable 5 and variable 6 exhibited a clear linear pattern. This accurate representation in the final graph highlights our method's capability of Lemma 2 in mixed linear and nonlinear scenarios. Nonetheless, several edges coincided with SCORE and CAM, indicating that, upon plotting the data pairs, missing edges were consistent with the absence of correlation in the plots. Furthermore, the correct partial graph consistently exhibited clear relationships between pairs of variables in all cases. However, utilizing a more refined real dataset, characterized by stronger associations between variables, could potentially yield improved results.

Appendix G. Algorithms

In this section, we present the overall algorithm for discovering the underlying causal graph from observational data generated by the model as described in Equation (1).

Algorithm 2 Causal Discovery with score's Jacobian

```
Input: Observational data: $\mathbf{X} \in \mathbb{R}^{n \times d}$
Output: Partially Oriented DAG: $\mathcal{G}$
Compute the moral graph for $\mathbf{X}$ according to Algorithm 3        ▷ Step1: Skeleton Discovery
d ← number of remaining nodes
while $d > 0$ do                                                          ▷ Step2: Edges Oriented
    Orient Edges according to Algorithm 4
    Remove leaves and linear blocks at the tail of the graph
end while
Orient collider sets according to Algorithm 5                             ▷ Step3: Colliders Oriented
Orient edges according to Meek orientation rules                          ▷ Step4: Meek Rules
return $\mathcal{G}$
```

Algorithm 3 Moral Graph Discovery with score's Jacobian

```
Input: Observational data: $\mathbf{X}$, complete undirected graph $\mathcal{G}$, threshold $k$
Output: Moral graph: $\mathcal{G}^m$
$d$ ← number of nodes
Estimate the score function $s_i(x) = \frac{\partial \log p(x)}{\partial x_i}$
Estimate score's Jacobian $h_{ij}(x) = \frac{\partial \log p(x)}{\partial x_i \partial x_j}$
for i in d do
    for j in i do
        if $h_{ij}(x) < k$ then
            Remove link $X_i - X_j$
        end if
    end for
end for
return $\mathcal{G}^m$
```

Appendix H. Additional Discussions

H.1. Comparisons with Jacobian-based methods

The Jacobian matrix is widely utilized in generative and inference models for identifiability and causal discovery. Various approaches also leverage the Jacobian matrix for different aspects of causal inference. In the line of causal discovery, LiNGAM (Shimizu et al., 2006) uses the Jacobian for inferring DAGs in linear scenarios. In contrast, Lachapelle et al. (2019) computes the Jacobian of the inference network for enforcing acyclicity in nonlinear additive models. Similarly, Rolland et al. (2022) focuses on the Jacobian of the score function within the same model class. Additionally, Atanackovic et al. (2023) proposes a Bayesian approach for causal discovery in dynamical

Algorithm 4 Edges orientation in the combination of nonlinear and linear models

```
Input: Moral Graph $\mathcal{G}^m$ or partial DAG $\mathcal{G}$
Output: $\mathcal{G}$ : Partial DAG
d ← number of remaining nodes
Compute $\widetilde{V} : \widetilde{v_{ij}} = \text{var}(\frac{\partial \log p(x)}{\partial x_i \partial x_j})$ with the remaining nodes
for i in d do
    if $\widetilde{v_{ii}} = 0$ then
        for $j \in \text{MB}(i)$ do
            if $\widetilde{v_{ij}} \neq 0$ then
                Orient edge $(i, j)$ as $j \to i$                ▷ Nonlinear leaf node
            else
                Orient edge $(i, j)$ as $i \to j$                ▷ Mixture nodes
            end if
        end for
    end if
end for
return $\mathcal{G}$
```

systems using the Jacobian of the SEMs. Meanwhile, Zheng et al. (2023) establish a Markov structure learning algorithm based on the Jacobian of the data generating process. The use of Jacobian properties extends into identifiability, where Independent Mechanism Analysis (IMA) assumes the generative model's Jacobian has orthogonal columns (Gresele et al., 2021). Reizinger et al. (2022) employs the Jacobian of the inference model for causal models with unconstrained function classes and non-i.i.d. data, providing identifiability guarantees. Our approach builds upon the Jacobian of the score function, extending the work of Rolland et al. (2022) to encompass mixed linear and nonlinear data. For a concise summary, see Table 3, an extension of Table 5 in Reizinger et al. (2022).

H.2. Comparisons with SCORE and DAS

SCORE algorithm (Rolland et al., 2022) initiates the process by efficiently recovering the topological order through the estimation of the Jacobian of the score function (i.e., $\nabla_x \log p(x)$). Leaf nodes are identified by locating terms with zero variance in the diagonal of the score's Jacobian. The fully connected DAG is then pruned using the method proposed in Bühlmann et al. (2014). Recognizing the computational demands of the pruning step, Montagna et al. (2023a) proposed DAS, which eliminates this step by solely relying on the Jacobian of the score, resulting in enhanced efficiency.

In the scenario where all causal mechanisms f_i are purely linear in the model (as described in Equation (1)), $\forall i \in \{1, \cdots, d\}$, both SCORE and DAS algorithms fail to identify the causal graph. The topological ordering methods fail as the topological ordering between variables itself does not disambiguate the direction of edges in the DAG. For example, in the case of an SCM defined in Equation (1) with linear functions f_i, the Jacobian of the score function $\frac{\partial s_i}{\partial x_i} = -\frac{1}{\sigma_i^2} + c$ for every node i rather than for leaves only, the criterion of identifying the leaf node (Lemma 3 and Lemma 4) does not hold anymore, leading to a failure of the topological ordering method for the linear case. However, with purely linear data, the Markov Equivalence class can still be attained in our

Algorithm 5 Orient the Collider Sets

```
Input: $\mathcal{G}$ : Partially Oriented DAG: the Markov blanket information for each node $i \in \mathbf{V}$ and some
    oriented edges from Algorithm 4
Output: $\mathcal{G}$ : Partially Oriented DAG
 1: for each edge $i - j$ part of a fully undirected connected triangle do
 2:     $\mathbf{C}_{ij} \leftarrow$ null
 3:     $\mathbf{B} \leftarrow \{$nodes on a path of length greater than 2 between X and Y$\}$
 4:     for each $\mathbf{S} \subset \mathbf{Tri}(i - j)$ do                ▷ possible collider subset search
 5:         $\mathbf{Z} \leftarrow \mathbf{B} \backslash \mathbf{S}$
 6:         if $\frac{\partial \log p(x_i, x_j, \mathbf{x_z})}{\partial x_i \partial x_j} = 0$ then
 7:             $\mathbf{C}_{ij} \leftarrow \mathbf{S}$
 8:             break to line 20
 9:         end if
10:         $\mathbf{D} \leftarrow \mathbf{B} \cap \{$nodes reachable by $\mathbf{S}\}$
11:         $\mathbf{Z}' \leftarrow \mathbf{Z} \backslash \mathbf{D}$
12:         for each $\mathbf{S}' \subset \mathbf{D}$ do                ▷ search for possible descendants of collider
13:             $\mathbf{Z} \leftarrow \mathbf{Z}' \cup \mathbf{S}'$
14:             if $\frac{\partial \log p(x_i, x_j, \mathbf{x_z})}{\partial x_i \partial x_j} = 0$ then
15:                 $\mathbf{C}_{ij} \leftarrow \mathbf{S}$
16:                 break to line 20
17:             end if
18:         end for
19:     end for
20:     if $\mathbf{C}_{ij}$ is not null then                ▷ orientation directive
21:         remove spouse link of $i - j$ in $\mathcal{G}$
22:         for each $k \in (\mathbf{C}_{ij})$ do
23:             $\mathcal{G} \leftarrow \mathcal{G} \cup \{(i \rightarrow k \leftarrow j)\}$
24:         end for
25:     end if
26: end for
27: return $\mathcal{G}$
```

Table 3: Comparisons with Jacobian-based approaches: The table includes columns denoted as follows: Column f signifies constraints on the function class of the SEMs, the Data column enumerates restrictions on the data distribution, J describes the Jacobian of the employed function, CD indicates its application in causal discovery, and the Id. column denotes whether the method provides identifiability guarantees. For detailed information on Assumption 2 and Proposition 1, please refer to Reizinger et al. (2022) for specifics on Assums.2 and F.1.

Method	f	Data	J	CD	Identifiability
Shimizu et al. (2006)	Linear	Non-Gaussian	$J_{f^{-1}}$	✓	✓
Lachapelle et al. (2019)	Additive	Gaussian	$J_{f^{-1}}$	✓	✗
Gresele et al. (2021)	IMA	All	J_f	✗	✓
Zheng et al. (2023)	Sparse	All	J_f	✗	✓
Rolland et al. (2022)	Additive	Gaussian	$J_{\nabla_x \log p(x)}$	✓	✗
Atanackovic et al. (2023)	Cyclic (ODE)	All	J_f	✓	✗
Reizinger et al. (2022)	All	Assums.2, F.1	$J_{f^{-1}}$	✓	✓
Ours	Mixed Linear & Nonlinear	Gaussian	$J_{\nabla_x \log p(x)}$	✓	Partial

work using Lemma 1 to derive the Markov Network and Lemma 5 to address colliders, similar to the methodologies employed in constraint-based and score-based approaches. Nevertheless, our approach demonstrates superior scalability as the number of nodes increases.

On the other hand, in the context of mixed linear and nonlinear data, both SCORE and DAS again encounter challenges due to the non-identifiability of linear Gaussian data, where the variance of the Jacobian of the score function becomes zero, except for the nonlinear leaf nodes. Consequently, our approach significantly broadens the scope of applicability for causal discovery, providing theoretical guarantees in situations that pose difficulties for other existing methods. Overall, both SCORE and DAS are only suitable for a nonlinear additive Gaussian noise model, whereas our method can handle linear (up to the Markov Equivalence Class), nonlinear, and mixed linear and nonlinear scenarios.

H.3. Comparison with PC algorithm

Lemma 1, used to compute the moral graph, corresponds to the step of determining adjacencies in the PC algorithm. Lemma 5 utilizes the Jacobian of the score function to infer V-structures, equivalent to the orientation rule in the PC algorithm. However, Lemma 2 enhances our ability to ascertain directions, even in non-collider scenarios.

To illustrate this enhancement, consider a simple 3-node example: a chain $i \to j \to k$. Consider the corresponding Structural Causal Model (SCM) as follows, f_j is a nonlinear function of variable X_i, and f_k is a linear function of X_j. Lemma 2 allows us to recover the true underlying DAG in this case. However, the PC algorithm can only recover up to the Markov Equivalence class, resulting in $i - j - k$. This underscores the additional inferential power provided by our approach. Importantly, our method ensures that the set of causal structures compatible with our final partially oriented graph is a subset of the causal structures within the Markov equivalence class.

Furthermore, our approach exhibits superior scalability as the number of nodes increases. Constraint-based methods, such as PC (Spirtes and Glymour, 1991), fast causal inference (FCI) (Spirtes, 2001), and SGS (Spirtes et al., 2000), assess the conditional independence among variables and seek graph structures that satisfy these conditions under a faithfulness assumption. However, the main bottleneck in these methods lies in the challenging nature of conditional independence testing (Shah and Peters, 2020). On the other hand, score-based techniques involve defining a suitable score function and searching for the graph that best fits the data within an extensive graph space. Greedy approaches like greedy equivalence search (GES) (Chickering, 2002; Huang et al., 2018) are employed for this exploration, but their scalability is hampered by the super-exponential growth of the space with the number of nodes.