Scalable Causal Discovery with Score Matching

Francesco Montagna[*] FRANCESCO.MONTAGNA@EDU.UNIGE.IT
MaLGa-DIBRIS, Università di Genova

Nicoletta Noceti NICOLETTA.NOCETI@UNIGE.IT
MaLGa-DIBRIS, Università di Genova

Lorenzo Rosasco LROSASCO@MIT.EDU
MaLGa-DIBRIS, Università di Genova
MIT, CBMM
Istituto Italiano di Tecnologia

Kun Zhang KUNZ1@CMU.EDU
Carnegie Mellon University
MBZUAI

Francesco Locatello LOCATELF@AMAZON.COM
AWS

Editors: Mihaela van der Schaar, Dominik Janzing and Cheng Zhang

Abstract

This paper demonstrates how to discover the whole causal graph from the second derivative of the log-likelihood in observational nonlinear additive Gaussian noise models. Leveraging scalable machine learning approaches to approximate the score function $\nabla \log p(\mathbf{X})$, we extend the work of Rolland et al. (2022) that only recovers the topological order from the score and requires an expensive pruning step removing spurious edges among those admitted by the ordering. Our analysis leads to DAS (acronym for Discovery At Scale), a practical algorithm that reduces the complexity of the pruning by a factor proportional to the graph size. In practice, DAS achieves competitive accuracy with current state-of-the-art while being over an order of magnitude faster. Overall, our approach enables principled and scalable causal discovery, significantly lowering the compute bar.

Keywords: High dimensional causal discovery; Score matching; Scalability

1. Introduction

Causal discovery from observational data is a central problem affecting virtually all scientific domains, such as biology, genetics, economics, and machine learning (Sachs et al. (2005); Koller and Friedman (2009); Pearl (2009); Peters et al. (2017)). Given a causal model one can predict the effect of interventions on the system's variables without the need of accessing interventional data which might be costly, unfeasible or unethical to collect. On the other hand, inferring causal relations from observational data is generally non-identifiable and requires additional assumptions. In traditional causality research, algorithms to discover causal relationships from observations can be divided in three classes (Glymour et al. (2019); Schölkopf et al. (2021)). *Constraint-based* approaches like PC (Spirtes and Glymour (1991)), FCI and SGS (Spirtes et al. (2000)) test the conditional independence between the variables and search for graphs structures that satisfies them under a faithfulness

[*] Work partially done during an internship at Amazon Web Services, Tuebingen

assumption. Usually they do not output a unique graph but an equivalence class. The main bottleneck of these approaches is that conditional independence testing is notoriously difficult (Shah and Peters (2020)). *Score-based* methods define a suitable score function, and search for the graph that best fits the data. Greedy approaches, such as GES (Chickering (2002)), are used to search in this large space, which size grows super-exponentially with the number of nodes (thus limiting scalability). Finally, a *restricted model* class assumption, e.g., nonlinear relations and additive Gaussian noise, allows to identify the Directed Acyclic Graph (DAG) underlying the observations (Peters et al. (2017); Bühlmann et al. (2014); Lachapelle et al. (2020); Zheng et al. (2018); Shimizu et al. (2006)).

One main challenge affecting the discovery of the causal graph is that enforcing the DAG constraint has a cubic per-iteration cost in the number of variables, making the optimization the computational bottleneck. One approach to reduce the computational requirements, is to decouple the causal discovery task in two steps: first, a topological ordering is found, such that a node can be a parent only of its successors in the ordering, thus enforcing the acyclicity constraint. Then, a pruning step selects the correct subset of edges among those admitted by the inferred ordering, removing all spurious connections in the graph. In this setting, a step towards better scalability is the work of Rolland et al. (2022) that recently proposed the SCORE algorithm: first they efficiently estimate the score function $\nabla \log p(\mathbf{X})$, then they recover the topological order from the Jacobian of the score, and finally they prune the fully connected DAG by the method proposed in CAM (Bühlmann et al. (2014)). The pruning step is the bottleneck of SCORE, amounting to 95% of the runtime on graphs with 50 nodes and scaling cubic in the number of nodes.

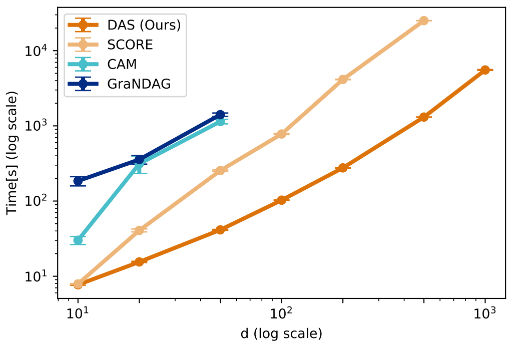


Figure 1: Execution time of different methods versus d number of nodes for dense graphs (ER4 data).

In this work, we show that the second derivative of the log-likelihood allows to recover both the skeleton and the direction of the edges in the causal DAG. Theoretically, this implies that we can get rid of the costly pruning step in SCORE (Rolland et al. (2022)) as all information about the causal structure is already contained in the Jacobian of the score. While our analysis yields a practical algorithm, we found it beneficial to first identify few candidate edges and still retain a final cheap pruning step. This is now much more efficient as most of the edges have already been detected and it is only needed to correct mistakes from the finite samples approximation of the score, reducing the complexity by a factor proportional to the number of nodes in the graph. This advantage is clearly visible in practice, reducing the runtime of SCORE by over an order of magnitude as shown in Figure 1.
Our contributions can be summarized as follows:

- We demonstrate how to theoretically recover the full causal DAG from the score of the data distribution. This extends prior work showing that the topological order can be recovered from the score (Rolland et al. (2022)).

- We introduce DAS[1] (acronym for Discovery At Scale), an algorithm for efficient and scalable causal discovery. As we lower the computational bar to apply causal discovery techniques on large numbers of variables, we also present clear examples and highlight when our algorithm is expected to fail. We expect these to be useful for practitioners interested in using DAS on their data as well as researchers working on scalable causal discovery.
- Our analysis yields a practical approach to filter the candidate edges in the final pruning step. While our method is marginally less accurate than (Rolland et al. (2022)) it improves the runtime by an order of magnitude in the graph size. We demonstrate the speedup improvement on synthetic graphs with up to a thousand nodes.

2. Background knowledge

We introduce the background needed for our analysis starting from the formalism of structural causal models.

2.1. Structural Causal Models

One way to formalize causal relationships between variables is with an additive Structural Causal Model (SCM). Consider a set $\mathbf{X} = \{X_i\}_{i=1}^d$ of observable vertices of a DAG. We assume that the structure of the graph can be expressed in the functional relationship

$$X_i = f_i(\mathrm{pa}_i(\mathbf{X})) + \epsilon_i, \qquad \epsilon_i \sim \mathcal{N}(0, \sigma_i), \qquad \forall i = 1, \dots, d \ , \tag{1}$$

with $\mathrm{pa}_i(\mathbf{X})$ set of parent nodes of X_i in the directed network. We will assume $X_i \in \mathbb{R}$, additive and independently drawn Gaussian noise elements ϵ_i, as well as f_i to be twice continuously differentiable and nonlinear in every component.

Recursive application of (1) allows to derive the joint probability distribution $p(X_1, X_2, \dots, X_d)$. As this probability is over vertices of a directed acyclic graph, the following factorization holds (Pearl (2009); Lauritzen (1996)):

$$p(\mathbf{X}) = \prod_{i=1}^{d} p(X_i | \, \mathrm{pa}_i(\mathbf{X})) \ . \tag{2}$$

The additive noise model (1) is known to be identifiable under *mild* assumptions (Hoyer et al. (2009); Zhang and Hyvärinen (2009)), meaning that we can recover uniquely the causal graph from observational data generated according to the joint distribution over $\mathbf{X}$.

Problem definition Usually the form of the f_i in the model (1) is not known and neither is the probability in (2), while we can only access a set of observations from the joint distribution. Given these observations the task is to identify the causal structure of the graph underlying the SCM. This problem is known as causal discovery. One solution is to use data to estimate a topological ordering of the variables in $\mathbf{X}$, and then to choose edges of the DAG between those admitted by such ordering. In our approach we select edges that satisfy constraints derived from the Jacobian of the score.

[1] The code for DAS is available as part of the DoDiscover library https://www.pywhy.org/dodiscover/dev/index.html

2.2. Topological ordering of a graph

Given a set of nodes $\mathbf{X} = \{X_i\}_{i=1}^d$, let $\mathcal{G} = (\mathbf{X}, \mathcal{E})$ be a DAG. A topological order relative to $\mathcal{G}$ is a permutation $\mathbf{X}^\pi$ of the nodes such that an edge (i, j) in $\mathcal{G}$ implies X_i appearing before X_j in $\mathbf{X}^\pi$, denoted as $X_j \succ X_i$. Intuitively, a node can be a parent only of its successors in the ordering. According to this definition more than one topological ordering is allowed for a given DAG. On the other hand, there is a one to one correspondence between a given permutation $\mathbf{X}^\pi$ and a fully connected graph $\mathcal{G}^\pi$ where we draw edges (i, j) for each $i, j = 1, \ldots, d$ such that $X_j \succ X_i$ in $\mathbf{X}^\pi$. If $\mathbf{X}^\pi$ is equal to $\mathbf{X}^*$, one of the correct permutations of the real DAG $\mathcal{G}$, then $\mathcal{G}^\pi = (\mathbf{X}, \mathcal{E}^\pi)$ is a supergraph of $\mathcal{G}$ meaning that its set of edges satisfies $\mathcal{E} \subseteq \mathcal{E}^\pi$. In the next section we provide an overview of CAM edges selection procedure, that allows to correctly identify $\mathcal{E}$ given a topological ordering $\mathbf{X}^\pi$ and its corresponding dense set $\mathcal{E}^\pi$.

2.3. Preliminary Neighbours Search (PNS) and CAM-pruning

Now we briefly describe the two-steps pruning method of CAM (Bühlmann et al. (2014)), namely *Preliminary Neighbours Search* (PNS) followed by *CAM-pruning*, which is used to remove *spurious* edges of $\mathcal{G}^\pi$, the dense graph induced by a permutation $\mathbf{X}^\pi$.

PNS PNS is a neighbours selection method for nonlinear additive Gaussian noise models, following the idea of variable selection in graphs presented in Meinshausen and Bühlmann (2006) for the linear Gaussian case. In particular, given an order $\mathbf{X}^\pi$, variable selection is performed by fitting for each $j = 1, \ldots, d$ an additive model of X_j versus all the other variables $\{X_i : X_j \succ X_i \text{ in } \mathbf{X}^\pi\}$, and choosing the K most *important* predictor variables as possible parents of X_j. This restricts the number of possible incoming edges of X_j to an arbitrary fixed amount K, reducing the total complexity of the edges search procedure and the workload of CAM-pruning step. PNS is implemented by the authors with a boosting method for additive models fitting (Bühlmann and Hothorn (2007); Bühlmann and Yu (2003)). The total complexity of PNS is therefore $\mathcal{O}(dt\,\mathrm{r}(n, d))$ as for each node $1, \ldots, d$ the boosting algorithm fits t models of complexity $\mathrm{r}(n, d)$, with n the number of samples in the dataset. The complexity term $\mathrm{r}(n, d)$ depends on the choice of the additive model fitting technique, amounting to $\mathcal{O}(nd^2)$ using Iteratively Reweighted Least Squares (Minka (2003)).

CAM-pruning After *Preliminary Neighbours Search*, a final pruning step is performed by applying significance testing of covariates to remove *superfluous* edges and thus reducing the number of false positives. In this case the computational complexity is negligible being bounded by the fixed parameter K decided in PNS.

Our work aims at replacing PNS with a novel approach making CAM-pruning application scalable to high dimensional graphs. In particular we reduce asymptotic complexity of the edge search procedure from $\mathcal{O}(nd^3)$ to $\mathcal{O}(d^2)$.

3. Deducing causal structure from the score

For the causal discovery problem under analysis we consider an observable $\mathbf{X} \in \mathbb{R}^d$ whose entries X_i are vertices of a graph generated according to the model in (1). In the next section, we show how the score function is in principle sufficient to solve this task. First we illustrate the ideas behind the SCORE algorithm, which can estimate the score's Jacobian and find a topological ordering of the variables of interest. Then we derive additional constraints on the Jacobian matrix of the score that allows to identify the edges of the causal graph.

3.1. SCORE overview

Rolland et al. (2022) introduces a method for efficiently estimating the score function $s(\mathbf{X})$ and its Jacobian exploiting the Stein identity (Stein (1972)). Complementary to this, they propose a method to identify leaf nodes in a causal graph generated according to (1) by inspection of the diagonal elements of the Jacobian of the score.
In order to derive the leaf identification procedure, first we need to find a closed form of $s(\mathbf{X}) = \nabla \log p(\mathbf{X})$. Starting from Equation (2), we have:

$$\begin{aligned} \log p(\mathbf{X}) &= \sum_{i=1}^{d} \log p(X_i \mid \mathrm{pa}_i(\mathbf{X})) = \\ &= -\frac{1}{2}\sum_{i=1}^{d}\left(\frac{X_i - f_i(\mathrm{pa}_i(\mathbf{X}))}{\sigma_i^2}\right)^2 - \frac{1}{2}\sum_{i=1}^{d}\log(2\pi\sigma_i)^2 \ . \end{aligned} \tag{3}$$

The j-th entry of $\nabla \log p(\mathbf{X})$ therefore is

$$s_j(\mathbf{X}) = -\frac{X_j - f_j(\mathrm{pa}_j(\mathbf{X}))}{\sigma_j^2} + \sum_{i \in \mathrm{ch}_j(\mathbf{X})} \frac{\partial f_i}{\partial x_j}(\mathrm{pa}_i(\mathbf{X}))\ \frac{X_i - f_i(\mathrm{pa}_i(\mathbf{X}))}{\sigma_i^2} \quad , \tag{4}$$

with $ch_j(\mathbf{X})$ denoting the set of children of node X_j. Now, let X_j be a leaf node: being the set of children nodes $ch_j(\mathbf{X}) = \emptyset$, from Equation (4) we notice that the score simplifies as follow:

$$s_j(\mathbf{X}) = -\frac{X_j - f_j(\mathrm{pa}_j(\mathbf{X}))}{\sigma_j^2}. \tag{5}$$

Moreover, it is easy to verify that $\frac{\partial s_j(\mathbf{X})}{\partial x_j} = -\frac{1}{\sigma_j^2}$, such that the diagonal entry of the score's Jacobian associated to a leaf node is a constant. Based on this relation, Lemma 1 of Rolland et al. (2022) provides a formal criterion to identify leaves in a causal graph. Given a node X_i, the following holds:

$$\frac{\partial s_i(\mathbf{X})}{\partial x_i} = c \iff X_i \text{ is a leaf}, \ \forall i = 1, \dots, d, \tag{6}$$

where $c \in \mathbb{R}$ is a constant scalar value. This relation directly implies that

$$\mathrm{Var}\left[\frac{\partial s_i(\mathbf{X})}{\partial x_i}\right] = 0 \iff X_i \text{ is a leaf}, \ \forall i = 1, \dots, d. \tag{7}$$

In order to find the complete topological ordering, SCORE algorithm of Rolland et al. (2022) is designed as follow: first it estimates the Jacobian of the score $\hat{J}(s(\mathbf{X}))$, that is used to identify a leaf in the graph by (7). Then, it removes the leaf from the graph, assigning it a position in the order vector. By iteratively repeating this two steps procedure up to the source node, all variables in $\mathbf{X}$ end up being assigned a position in the causal ordering.

In the following section we show that given that a topological ordering is known, we can derive additional constraints on the off-diagonal elements of the score's Jacobian that identify directed edges in the graph.

3.2. Deriving constraints for edge selection

We observe that for a leaf node l, $X_l \in \mathbf{X}$, the partial derivative of (4) over X_j with $j \neq l$ is:

$$\frac{\partial s_l(\mathbf{X})}{\partial X_j} = \begin{cases} \frac{1}{\sigma_l^2}\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X})) \neq 0 & \text{if } X_j \in \mathrm{pa}_l(\mathbf{X}) \\ 0 & \text{else} \end{cases} . \tag{8}$$

It is worth to notice that $\frac{1}{\sigma_l^2}\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X}))$ might still be vanishing for some values of $\mathrm{pa}_l(\mathbf{X})$ even if $X_j \in \mathrm{pa}_l(\mathbf{X})$, for instance if the function has a maximum or a minimum: given the assumption on f_l nonlinear even when considered on a restricted interval, these events happen with probability zero, such that $\frac{1}{\sigma_l^2}\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X})) \neq 0$ holds *almost surely*. We prove that the condition in Equation (8) allows to derive a criterion to identify parents of a given leaf node by slightly adapting the result of Rolland et al. (2022).

Lemma 1 (Adapted from Rolland et al. (2022)) *Let p be the probability density function of a random variable $\mathbf{X} \in \mathbb{R}^d$ defined via nonlinear additive Gaussian noise model (1). Let also $s(\mathbf{X}) = \nabla \log p(\mathbf{X})$ be the associated score function. Without loss of generality, assume a topological ordering $\mathbf{X}^\pi = (X_1, \dots, X_d)$. Then given a leaf l:*

$$\mathbf{E}\left[\left|\frac{\partial s_l(\mathbf{X})}{\partial X_j}\right|\right] \neq 0 \iff X_j \in \mathrm{pa}_l(\mathbf{X}), \;\; \forall j \in \{1, \dots, l-1\} . \tag{9}$$

The proof is provided for completeness in the Appendix A.

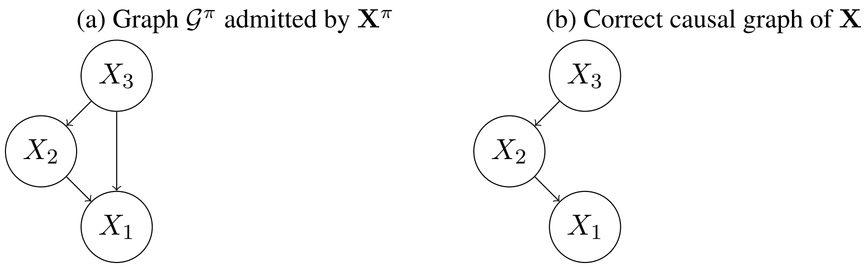
Difference of Lemma 1 with Rolland et al. (2022) The formulation in Rolland et al. (2022) requires $\mathrm{Var}\left[\frac{1}{\sigma_l^2}\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X}))\right] \neq 0 \Leftrightarrow X_j \in \mathrm{pa}_l(\mathbf{X})$, where X_l is a leaf node. We illustrate the problem with this considering a simple two variables case with graph $X_1 \longrightarrow X_2$: if parent node X_1 has zero variance, their selection condition would break, predicting a graph with X_1 and X_2 independent. While this case would be ruled out by the assumption of variance larger than zero for every node, in practice it can be a problem. Given a finite sample $X \in \mathbb{R}^{n \times d}$ and its topological ordering $\mathbf{X}^\pi$, if parents of a leaf X_l show small variance in the sample, we might still mistake the oscillation observed in $\frac{1}{\sigma_l^2}\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X}))$ for statistical error due to finite set estimates, discarding an existing edge. To estimate the edges we need higher statistical accuracy compared to the topological order since in the former case we do not know how many parents a node has while in the latter we know there is always at least one leaf. This is why we adapted the Lemma of Rolland et al. (2022) to a theoretically equivalent but practically more robust formulation. We rely on the sample mean of the absolute value of the score's Jacobian entries $\frac{1}{\sigma_l^2}\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X}))$ for the implementation of Lemma 1: this estimator is potentially subject to the same issues, but shows better robustness properties than the sample variance (due to the absolute value) and estimating a lower moment yields lower error (estimating variance requires estimating the mean first, so any statistical error in the mean estimator affects the variance estimator), making it a preferable choice.

In practice we can exploit Lemma 1 to reconstruct the entire graph only if an ordering $\mathbf{X}^\pi$ is provided. To see why, consider the last entry X_l of $\mathbf{X}^\pi$: by definition of topological ordering X_l is a leaf. Then we can apply Lemma 1 doing partial derivatives of $s_l(\mathbf{X})$ over all nodes $\{X_j : X_j \prec X_l \text{ in } \mathbf{X}^\pi\}$ and identify as parents those that satisfy the required constraint. At this point, we remove

X_l from the ordering $\mathbf{X}^\pi$ and repeat the procedure on the pruned graph with vertices $\mathbf{X} \setminus X_l$. By iterating these steps over each node in the ordering from last to source we can identify the exact graph.

Example 1 *To clarify these ideas we discuss a simple three variables example illustrating how the results in Lemma 1 enables edges discovery.*

Let $\mathbf{X} = (X_1, X_2, X_3)$ with $X_i \in \mathbb{R}$ generated according to model (1). Consider the topological ordering $\mathbf{X}^\pi = (X_3, X_2, X_1)$ to be given. The goal is to recover the real causal graph with X_3 as source node, $pa_1(\mathbf{X}) = \{X_2\}$ and $pa_2(\mathbf{X}) = \{X_3\}$.


(a) Graph $\mathcal{G}^\pi$ admitted by $\mathbf{X}^\pi$

(b) Correct causal graph of $\mathbf{X}$

We proceed analyzing the Jacobian of the score function for X_1 and X_2, while we easily see from the ordering $\mathbf{X}^\pi$ that X_3 has no parents.

- *X_1: according to (5) its score is simply $s_1(\mathbf{X}) = -\frac{X_1 - f_1(X_2)}{\sigma_1^2}$. Its off diagonal partial derivatives are*

$$\frac{\partial s_1}{\partial X_2}(\mathbf{X}) = \frac{1}{\sigma_1^2}\frac{\partial f_1}{\partial X_2}(X_2) \ ,$$
$$\frac{\partial s_1}{\partial X_3}(\mathbf{X}) = 0 \ .$$

  Applying Lemma 1 we correctly conclude that the set of parents of X_1 is $\mathrm{pa}_1(\mathbf{X}) = \{X_2\}$.

- *X_2: in order to apply Lemma 1 also to X_2 we need to discard X_1 from the graph, such that X_2 is a leaf node. Given the set of variables $\tilde{\mathbf{X}} = \mathbf{X} \setminus \{X_1\}$ we get the graph of Figure 3. Then $s_2(\tilde{\mathbf{X}}) = -\frac{X_2 - f_2(X_3)}{\sigma_2^2}$ and the partial derivative is*

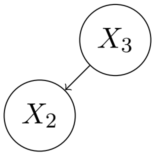
$$\frac{\partial s_2}{\partial X_3}(\tilde{\mathbf{X}}) = \frac{1}{\sigma_2^2}\frac{\partial f_2(X_3)}{\partial X_3} \ .$$

  By Lemma 1 we see that $\mathbf{E}\left[\left|\frac{1}{\sigma_2^2}\frac{\partial f_2(X_3)}{\partial X_3}\right|\right] \neq 0$, such that X_3 is a parent of X_2.

The algorithm recovers the exact structure.

Discussion These findings on identifiability of the causal structure from the score function are not completely surprising in the light of previous results on Markov networks (Spantini et al. (2018); Morrison et al. (2017)). Given a collection of random variables $\mathbf{X} = (X_1, X_2, \dots, X_d)$ with joint

Figure 3: Causal graph of $\tilde{\mathbf{X}} = (X_2, X_3)$.



density $p(\mathbf{X})$, the information of conditional independencies between the variables of $\mathbf{X}$ can be embedded in a simple undirected Markov network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where edges (i, j) encode some sort of probabilistic interaction between the pairs of random variables X_i, X_j. In particular Spantini et al. (2018) proved how to construct a Markov graph reading the conditional independence of pairs of random variables as follow:

$$X_j \perp\!\!\!\perp X_i \,|\, \mathbf{X}_{\mathcal{V}\setminus\{i,j\}} \iff \partial_{ij} \log p(\mathbf{X}) = 0 \ , \tag{10}$$

where $\partial_{ij}(\cdot)$ denotes the ij-th mixed partial derivative and $\partial_{ij} \log p(\mathbf{X})$ is an entry of the Jacobian of the score. By adding edges between each couple of nodes that appears not to satisfy Equation (10), we obtain an undirected graph encoding all and only the existing conditional independencies between the variables of $\mathbf{X}$.

Equation (8) of our work discovers the same constraint in a slightly different setting: rather than evaluating $\partial_{ij} \log p(\mathbf{X})$ for each node against every other, we follow an iterative approach where first we identify a leaf X_l and then we test its mixed derivatives only against nodes coming before in the topological ordering. By the time we find an edge we know its direction as we know that X_l is a leaf, which breaks the symmetry in the relation. Moreover Lemma 1 ensures correct identification of directed *v-structures* like $i \rightarrow j \leftarrow k$ that instead in the conditional independence map are *moralized* with an additional link (i, k), thus allowing to recover all edges and their direction.

Next we derive an algorithm for causal discovery based on Lemma 1, and show how it retains performance with respect to other state of the art methods, while showing better scalability in the number of nodes.

3.3. DAS: an algorithm for causal Discovery At Scale

We want to use the constraint of Lemma 1 on the score function to derive an algorithm for causal discovery which is faster and exhibits better scaling properties in the number of nodes than any other technique to our knowledge. Given a set $X \in \mathbb{R}^{n\times d}$ of n observations generated according to model (1), first we estimate a topological ordering $\hat{\mathbf{X}}^\pi$ via the SCORE algorithm. Then, we iterate over each node X_l of such ordering starting from the last, which we know to be a leaf, and identify incoming edges of X_l as follow: to begin we estimate the score's Jacobian $\hat{J}$ via the SCORE algorithm; we have now n estimates of the Jacobian of the score, one for each of the n data points. According to Lemma 1, we consider the absolute value of the l-th row of the n Jacobian matrices, and look for entries with non-zero mean: this can be achieved by statistical hypothesis testing, where the idea is to test for the mean of a sample to be different from zero. In practice, we perform hypothesis testing according to the following heuristic method: we compute the average of the absolute value of the entries in the l-th row, and select as potential parents of X_l the $K + 1$ nodes associated to largest averages. This parameter K is the same that we find in PNS algorithm, which limits the maximum number of nodes fed to the pruning algorithm. Between these $K+1$ entries, we find a reference node X_{ref} whose samples of $\left|\hat{J}_{l,ref}\right|$ have the average closest to zero: at this point, for each of the remaining K nodes, we perform Welchs's t-test with the null hypothesis of equality of the population means $H_0 : \mathbf{E}[|J_{l,j}|] = \mathbf{E}[|J_{l,ref}|]$ and the alternative $H_1 : \mathbf{E}[|J_{l,j}|] > \mathbf{E}[|J_{l,ref}|]$, with X_j potential parent of X_l. If we reject the null with *p-value* ≤ 0.01, then X_j is added to the parents of X_l in an adjacency matrix representing the inferred graph.

Eventually, X_l column is removed from the matrix of the data X, and the procedure is repeated for another leaf node found in the ordering. Once every node in $\mathbf{X}^{\pi}$ is considered, we prune the resulting adjacency matrix via *CAM-pruning* (Section 2.3), simply with the goal of reducing the number of false positives.

The implementation details of DAS are illustrated in the pseudo-code of Algorithm 1.

3.4. Algorithmic complexity

Considering an input matrix $n \times d$ with n the number of samples and d the number of nodes, the overall complexity of DAS is $\mathcal{O}(dn^3 + d^2)$. Indeed estimating the topological order with SCORE involves inverting a $n \times n$ matrix for d times, one for each iteration necessary to identify a leaf node: hence, the $\mathcal{O}(dn^3)$ contribute. Additionally the edge search step requires iterating over the d elements of the ordering, each time selecting the K largest entries on a list of size $\leq d$ (see Algorithm 1) yielding a $\mathcal{O}(d^2)$ contribution.

On the other hand SCORE, arguably the most scalable state-of-the-art algorithm for causal discovery, uses PNS and CAM-pruning to select edges while shares the same ordering of DAS. The bottleneck in SCORE execution is the preliminary neighbours search step, whose complexity has been studied in detail in Section 2.3 and amounts to $\mathcal{O}(dt\,\mathrm{r}(n,d))$ with $\mathrm{r}(n,d)$ the number of operations to fit a generalized additive model ($\mathcal{O}(nd^2)$ (Minka (2003)) with Iteratively Reweighted Least Squares). Therefore our use of the score function for candidate edges selection dramatically improves the execution time allowing to scale causal discovery in high dimensions by a factor of $\mathcal{O}(d)$.

Non identifiability of the linear model Next we further highlight the consistency of our algorithm by showing how it fails in identifying the causal graph under the assumption of linear f_j in model (1), $\forall j = 1, \ldots, d$. Indeed it has been proven that observational data generated according to a linear additive Gaussian noise model do not allow for recovery of the underlying causal structure (Peters et al. (2017); Comon (1994)) unless additional assumptions are made (Peters and Bühlmann (2013)). In the simplest setting of two variables X, Y linked in a causal graph, unidentifiability amounts to the impossibility of choosing the edge direction, i.e., it is not possible to decide whether X is the cause or the effect of Y.

Given that the topological ordering between variables itself disambiguates the direction of edges in the DAG, we have to show that our topological ordering method on a set $\mathbf{X} \in \mathbb{R}^d$ fails in the linear setting. SCORE algorithm identifies leaf nodes by finding terms with zero variance in the diagonal of the score's Jacobian, as specified in Equation (7). In case of an SCM as defined in (1) but with linear functions f_j, we find that $\frac{\partial s_j}{\partial X_j}(\mathbf{X}) = -\frac{1}{\sigma_j^2} + \text{constant}$ for every node j in the graph. Thus it can be easily seen that

$$\mathrm{Var}\left[\frac{\partial s_j}{\partial X_j}(\mathbf{X})\right] = 0,\ \forall j = 1, \ldots, d\ . \tag{11}$$

```
Algorithm 1: DAS
Input: data matrix $X \in \mathbb{R}^{n \times d}$, $K \in \mathbb{R}$
$X^\pi \leftarrow \mathrm{SCORE}(X)$                ($X^\pi[d]$ leaf node)
$X^\pi \leftarrow reverse(X^\pi)$               ($X^\pi[d]$ source node)
$A \leftarrow d \times d$ zeros adjacency matrix
for $X_l$ in $X^\pi$ do
    $\hat{J} \leftarrow \left[\left|\frac{\partial s_l}{\partial X_j}\right|\right]_{X_j \prec X_l \text{ in } X^\pi}$        (estimate from SCORE)
    $K \leftarrow \min(K, \mathrm{length}(\hat{J}))$
    $topK = topK(\mathrm{Average}(\hat{J}))$      ( topK$(\cdot)$ : return indices of the K largest values of the input)
    $j_{ref} = topK[\mathrm{argmin}\ \hat{J}[topK]]$
    for $k$ in $topK$ do
        $p =$ p-value for the test $H_0 : \mu_{J_k} = \mu_{J_{ref}}, H_1 : \mu_{J_k} > \mu_{J_{ref}}$        ($\mu$ = mean)
        if $p < 0.01$ then
            $A[k, l] = 1$
        end
    end
    Remove $l$-th column from $X$
end
$\hat{\mathcal{G}} \leftarrow$ CAM-pruning$(A)$
return $\hat{\mathcal{G}}$
```

Since the variance in (11) vanishes for each node j rather than for leaves only, then the criterion of Equation (7) does not hold anymore. This implies failure of the topological ordering method for the linear case in accordance with our claim.

In the next section we study the algorithmic complexity of DAS and we highlight the better efficiency with respect to SCORE.

4. Experiments

Now we summarize experimental outcomes of DAS method in comparison with several state of the art algorithms for causal discovery. We report results of SCORE-ordering with PNS and CAM-pruning steps (named simply SCORE in the table) (Rolland et al. (2022)), CAM [2] (Bühlmann et al. (2014)) and GraNDAG (Lachapelle et al. (2020)). Other algorithms such as PC and FCI are omitted as they perform much worse (Bühlmann et al. (2014); Lachapelle et al. (2020)). Up to 200 nodes we ran experiments on a machine with 16GB RAM and 8 processors Intel(R) Core(TM) i5-8265U CPU at 1.60GHz. For 500 or more nodes we used a machine with 256GB RAM and 64 processors AMD EPYC 7301 16-Core Processor at 2.20GHz.

The metrics used are precision, recall, Structural Hamming Distance (SHD) – which is computed as the sum of false positive, false negative and wrongly directed edges – and Structural Intervention Distance (SID) (Peters and Bühlmann (2015)) – accounting for the number of miscalculated inter-

[2] CAM refers to both topological ordering and pruning steps introduced in the original paper. PNS is applied only for $d \geq 50$.

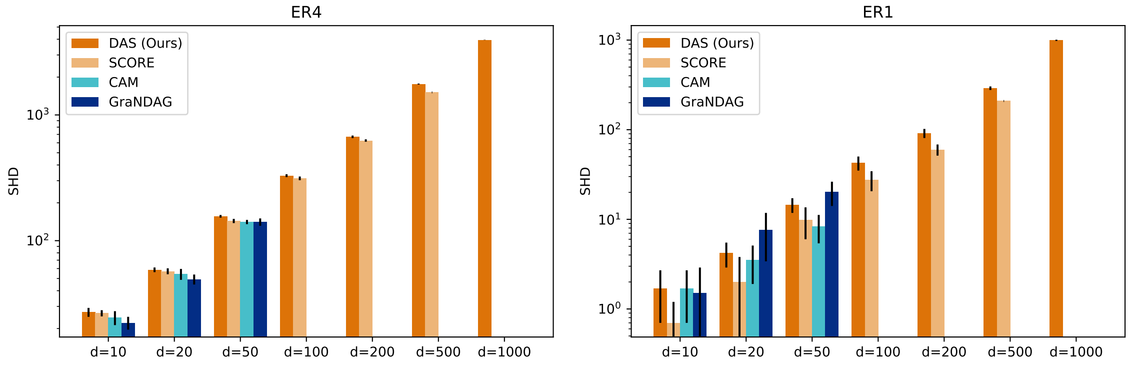


Figure 4: SHD versus d number of nodes for different methods on dense (left) and sparse (right) graphs. For higher values of d some methods are missing as they were too much time expensive to run. Number of samples is $n = 1000$.

ventional distributions that would result from the inferred graph.

We focus our experiments on synthetic data to show the scalability properties of DAS with increasing number of nodes. In order to sample data from the nonlinear additive Gaussian noise model of Equation (1) we mimic the experimental setting of Rolland et al. (2022). The causal graphs are generated using the Erdös-Renyi model (Erdos and Renyi (1960)). We run experiments fixing the number of nodes d as well as the sparsity of the graph by setting the expected amount of edges to be equal to d (ER1) or $4d$ (ER4). For $d > 200$ we drop the SID metric as it is too slow to compute. Whenever results for some method are not appearing in the table this means we could not perform these runs in a reasonable time. We repeat the experiments for 10 times and report empirical mean and standard deviation over the metrics. The number of samples is maintained fixed at $n = 1000$ and we set $K = 20$ (same value found SCORE and CAM experimental settings).

From Table 1 we can see that on denser graphs (ER4) our method maintains similar performance with respect to the other three for nodes up to 50, while being considerably faster in particular with respect to GraNDAG and CAM. As d increases, the gap with SCORE reduces up to the point that for 200 nodes we observe better SID for our algorithm. At $d \geq 500$ it becomes arguably impossible to run SCORE on a personal computer in a finite amount of time, whereas DAS is the only reasonable option.

Similarly, the performances across the different methods are comparable when running inference on sparser graphs (ER1), as reported in Table 2. These results are directly observable in Figure 4: each algorithm shows a similar degrade in performance with the number of nodes increasing, and bars set to close SHD values. Nevertheless, in Figure 1 it clearly appears that DAS achieves these metrics in a significantly smaller amount of time, supporting the claim of better efficiency in terms of velocity and scalability of our approach.

In Appendix B we provide additional empirical results, focused on the stability of DAS with respect to changes in the threshold for hypothesis testing of mean larger than zero. Moreover, we significantly extend our experiments testing DAS on Scale-free synthetic graphs (Barabasi and Albert (1999), Appendix C), and on Sachs real data (Sachs et al. (2005)) and semi-synthetic data sampled from SynTReN generator (Van den Bulcke et al. (2006))), Appendix D).

Table 1: Experiments on ER4 data. For CAM and GraNDAG we report results found in Rolland et al. (2022). For higher values of d some methods are missing as they were too much time expensive to run.

	Method	SHD	SID	Prec.	Rec.	Time [s]
d=10	DAS (Ours)	27.0 ± 2.2	43.6 ± 5.8	1.00 ± 0.00	0.33 ± 0.02	$\mathbf{7.7 \pm 0.1}$
	SCORE	26.5 ± 1.5	42.3 ± 2.9	0.99 ± 0.00	0.33 ± 0.02	7.9 ± 0.1
	CAM	24.4 ± 3.1	45.2 ± 10.2	–	–	30.1 ± 3.7
	GraNDAG	$\mathbf{22.2 \pm 2.6}$	$\mathbf{42.0 \pm 6.2}$	–	–	185 ± 26
d=20	DAS (Ours)	56.4 ± 2.5	213 ± 28	0.99 ± 0.00	0.27 ± 0.04	$\mathbf{16.1 \pm 0.3}$
	SCORE	57.17 ± 3.1	229 ± 23	0.99 ± 0.01	0.30 ± 0.04	40.7 ± 1.8
	CAM	54.2 ± 5.4	202 ± 29	–	–	313 ± 80
	GraNDAG	$\mathbf{49.3 \pm 4.5}$	$\mathbf{211 \pm 37}$	–	–	357 ± 47
d=50	DAS (Ours)	156 ± 4	1460 ± 67	0.96 ± 0.02	0.24 ± 0.03	$\mathbf{48.3 \pm 1.1}$
	SCORE	144 ± 6	1346 ± 57	0.97 ± 0.01	0.30 ± 0.03	245 ± 5
	CAM	$\mathbf{141 \pm 6}$	$\mathbf{1337 \pm 94}$	–	–	1143 ± 79
	GraNDAG	$\mathbf{141 \pm 10}$	1432 ± 110	–	–	1410 ± 73
d=100	DAS (Ours)	335 ± 5	6695 ± 224	0.91 ± 0.03	0.21 ± 0.04	$\mathbf{114 \pm 2}$
	SCORE	$\mathbf{313 \pm 11}$	$\mathbf{5965 \pm 273}$	0.91 ± 0.03	0.27 ± 0.06	779 ± 13
d=200	DAS (Ours)	691 ± 13	$\mathbf{25147 \pm 835}$	0.89 ± 0.06	0.21 ± 0.05	$\mathbf{279 \pm 4}$
	SCORE	$\mathbf{626 \pm 14}$	25707 ± 891	0.88 ± 0.04	0.30 ± 0.05	4142 ± 35
d=500	DAS (Ours)	1761 ± 15	–	0.80 ± 0.04	0.19 ± 0.03	$\mathbf{1308 \pm 7}$
	SCORE[3]	$\mathbf{1642}$	–	0.82	0.27	25307
d=1000	DAS (Ours)	$\mathbf{3951 \pm 9}$	–	0.76 ± 0.05	0.08 ± 0.00	$\mathbf{5539 \pm 81}$

[2] For $d = 500$ and SCORE method no standard deviation appears because experiments could not be repeated in a reasonable time. The values in the table refer to a single run.

5. Conclusion

Under the assumption of nonlinear additive Gaussian noise model of the data we showed how to theoretically recover the exact causal graph from the Jacobian of the score function. Our finding extends the work of Rolland et al. (2022), using the score to learn the topological ordering of the variables from the data. In addition to this, we showed that all edges can be discovered and oriented from the score. Based on our analysis, we designed an algorithm that is more efficient by a factor of $\mathcal{O}(d)$ compared to Rolland et al. (2022), yielding significant speed up in practice while retaining comparable accuracy.

While we already obtained a significant speed up, further improvement could be achieved by amortizing the score computation over subgraphs or even multiple graphs (Löwe et al. (2020)) or using more scalable estimators. However, the main blocker for future work (and biggest limitation of our work) is the lack of a public, highly curated, and large benchmark for causal discovery. In fact, our scalability experiments were limited to inference on synthetic data. Without real data it is hard to evaluate algorithms' performance outside of a controlled synthetic setting. As huge and fully annotated causal graphs may never become available, we identify three important directions for future work. First, extending our approach to input variables outside of the model assumptions, in partic-

Table 2: Experiments on ER1 data. For CAM and GraNDAG we report results found in Rolland et al. (2022). For higher values of d some methods are missing as they were too much time expensive to run.

	Method	SHD	SID	Prec.	Rec.	Time [s]
d=10	DAS (Ours)	1.0 ± 0.8	$\mathbf{3.3} \pm \mathbf{3.7}$	0.97 ± 0.01	0.84 ± 0.05	$\mathbf{7.8} \pm \mathbf{0.1}$
	SCORE	$\mathbf{0.7} \pm \mathbf{0.5}$	4.5 ± 4.3	0.98 ± 0.01	0.98 ± 0.01	8.0 ± 0.2
	CAM	1.7 ± 1.0	6.4 ± 4.2	–	–	30.1 ± 3.7
	GraNDAG	1.5 ± 1.4	6.5 ± 7.2	–	–	185 ± 26
d=20	DAS (Ours)	3.2 ± 1.4	18.1 ± 10.1	0.98 ± 0.02	0.85 ± 0.03	$\mathbf{16.7} \pm \mathbf{0.4}$
	SCORE	$\mathbf{2.0} \pm \mathbf{1.8}$	$\mathbf{8.3} \pm \mathbf{9.9}$	0.99 ± 0.01	0.91 ± 0.03	36.4 ± 1.8
	CAM	3.5 ± 1.6	14.3 ± 9.8	–	–	313 ± 80
	GraNDAG	7.6 ± 4.2	31.6 ± 22.7	–	–	357 ± 47
d=50	DAS (Ours)	17.3 ± 3.4	99.6 ± 42.1	0.96 ± 0.04	0.77 ± 0.04	$\mathbf{49.2} \pm \mathbf{2.1}$
	SCORE	9.8 ± 3.8	69.6 ± 41.3	0.98 ± 0.01	0.87 ± 0.03	251 ± 7
	CAM	$\mathbf{8.3} \pm \mathbf{2.9}$	$\mathbf{53.7} \pm \mathbf{31.9}$	–	–	1143 ± 79
	GraNDAG	20.2 ± 6.1	135 ± 456	–	–	1410 ± 73
d=100	DAS (Ours)	46.6 ± 9.1	327 ± 47	0.92 ± 0.06	0.68 ± 0.04	$\mathbf{111} \pm \mathbf{6}$
	SCORE	$\mathbf{27.5} \pm \mathbf{6.9}$	$\mathbf{288} \pm \mathbf{115}$	0.97 ± 0.02	0.83 ± 0.05	776 ± 12
d=200	DAS (Ours)	110 ± 6.6	899 ± 191	0.88 ± 0.07	0.68 ± 0.06	$\mathbf{282} \pm \mathbf{4}$
	SCORE	$\mathbf{59.9} \pm \mathbf{8.5}$	$\mathbf{495} \pm \mathbf{161}$	0.95 ± 0.03	0.85 ± 0.07	4237 ± 22
d=500	DAS (Ours)	291 ± 13	–	0.78 ± 0.07	0.65 ± 0.05	$\mathbf{1329} \pm \mathbf{7}$
	SCORE[4]	$\mathbf{209}$	–	0.8	0.85	25115
d=1000	DAS (Ours)	$\mathbf{994} \pm \mathbf{15}$	–	0.59 ± 0.02	0.09 ± 0.00	$\mathbf{5544} \pm \mathbf{73}$

[3] For $d = 500$ and SCORE method no standard deviation appears because experiments could not be repeated in a reasonable time. The values in the table refer to a single run.

ular considering data coming from interventions and with latent confounders. Second, evaluation protocols with partially annotated causal graphs and externally collected interventional distributions. Third, merging many smaller and annotated data sets that share similar but non-overlapping variables (Mejia et al. (2022)).

Acknowledgments

We want to thank Volkan Cevher for the valuable discussions. This work has been supported by AFOSR, grant n. FA8655-20-1-7035. FM is supported by *Programma Operativo Nazionale ricerca e innovazione 2014-2020.*

References

Albert-Laszlo Barabasi and Reka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999. doi: 10.1126/science.286.5439.509. URL http://www.sciencemag.org/cgi/content/abstract/286/5439/509.

Peter Bühlmann and Torsten Hothorn. Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22(4), nov 2007. URL https://doi.org/10.1214%2F07-sts242.

Peter Bühlmann and B. Yu. Boosting with the l 2 loss. *Journal of The American Statistical Association - J AMER STATIST ASSN*, 98:324–339, 06 2003.

Peter Bühlmann, Jonas Peters, and Jan Ernest. CAM: Causal additive models, high-dimensional order search and penalized regression. *The Annals of Statistics*, 42(6), dec 2014. URL https://doi.org/10.1214%2F14-aos1260.

David Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554, 01 2002.

Pierre Comon. Independent component analysis, a new concept? *Signal Processing*, 36(3):287–314, 1994. ISSN 0165-1684. URL https://www.sciencedirect.com/science/article/pii/0165168494900299. Higher Order Statistics.

Paul Erdos and Alfred Renyi. On the evolution of random graphs. *Publ. Math. Inst. Hungary. Acad. Sci.*, 5:17–61, 1960.

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in Genetics*, 10, 2019. ISSN 1664-8021. URL https://www.frontiersin.org/article/10.3389/fgene.2019.00524.

Patrik O Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In *Advances in neural information processing systems*, pages 689–696, 2009.

Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009. ISBN 0262013193.

Sébastien Lachapelle, Philippe Brouillard, Tristan Deleu, and Simon Lacoste-Julien. Gradient-based neural dag learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rklbKA4YDS.

Steffen L. Lauritzen. *Graphical Models*. Oxford University Press, 1996. ISBN 0-19-852219-3.

Sindy Löwe, David Madras, Richard Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. *arXiv preprint arXiv:2006.10833*, 2020.

Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3), jun 2006. URL https://doi.org/10.1214%2F009053606000000281.

Sergio H Garrido Mejia, Elke Kirschbaum, and Dominik Janzing. Obtaining causal information by merging datasets with maxent. In *International Conference on Artificial Intelligence and Statistics*, pages 581–603. PMLR, 2022.

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

Rebecca Morrison, Ricardo Baptista, and Youssef Marzouk. Beyond normality: Learning sparse probabilistic graphical models in the non-gaussian setting. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/ea8fcd92d59581717e06eb187f10666d-Paper.pdf.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, USA, 2nd edition, 2009. ISBN 052189560X.

J. Peters and P. Bühlmann. Identifiability of gaussian structural equation models with equal error variances. *Biometrika*, 101(1):219–228, nov 2013. URL https://doi.org/10.1093%2Fbiomet%2Fast043.

Jonas Peters and Peter Bühlmann. Structural Intervention Distance for Evaluating Causal Graphs. *Neural Computation*, 27(3):771–799, 03 2015. ISSN 0899-7667. doi: 10.1162/NECO_a_00708. URL https://doi.org/10.1162/NECO_a_00708.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 2017. ISBN 978-0-262-03731-0. URL https://mitpress.mit.edu/books/elements-causal-inference.

Paul Rolland, Volkan Cevher, Matthäus Kleindessner, Chris Russel, Bernhard Schölkopf, Dominik Janzing, and Francesco Locatello. Score matching enables causal discovery of nonlinear additive noise models. In *(To appear) International Conference on Machine Learning (ICML)*, 2022. URL https://arxiv.org/abs/2203.04413.

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. URL https://www.science.org/doi/abs/10.1126/science.1105809.

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. *The Annals of Statistics*, 48(3), jun 2020.

Shohei Shimizu, Patrik O. Hoyer, Aapo Hyvarinen, and Antti Kerminen. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(72):2003–2030, 2006. URL http://jmlr.org/papers/v7/shimizu06a.html.

Alessio Spantini, Daniele Bigoni, and Youssef Marzouk. Inference via low-dimensional couplings. *J. Mach. Learn. Res.*, 19(1):2639–2709, jan 2018. ISSN 1532-4435.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT press, 2nd edition, 2000.

Peter Spirtes and Clark Glymour. An algorithm for fast recovery of sparse causal graphs. *Social Science Computer Review*, 9(1):62–72, 1991. URL https://doi.org/10.1177/089443939100900106.

Charles M. Stein. A bound for the error in the normal approximation to the distribution of a sum of dependent random variables. 1972.

Tim Van den Bulcke, Koenraad Leemput, Bart Naudts, Piet Remortel, Hongwu Ma, Alain Verschoren, Bart De Moor, and Kathleen Marchal. Syntren: a generator of synthetic gene expression data for design and analysis of structure learning algorithms. *BMC bioinformatics*, 7:43, 02 2006. doi: 10.1186/1471-2105-7-43.

K. Zhang and A. Hyvärinen. Acyclic causality discovery with additive noise: An information-theoretical perspective. In *Proc. European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD) 2009*, Bled, Slovenia, 2009.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/e347c51419ffb23ca3fd5050202f9c3d-Paper.pdf.

Appendix A. Proof of Lemma 1

In this section we provide a proof of the statement of Lemma 1 for completeness.

Proof For a leaf l the score of Equation (4) becomes $s_l(\mathbf{X}) = -\frac{X_l - f_l(\mathrm{pa}_l(\mathbf{X}))}{\sigma_l^2}$. We compute the partial derivative

$$\frac{\partial s_l(\mathbf{X})}{\partial X_j} = \frac{1}{\sigma_l^2}\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X})) \tag{12}$$

and observe that:

(i) $\mathbf{E}\left[\left|\frac{\partial s_l(\mathbf{X})}{\partial X_j}\right|\right] \neq 0 \Rightarrow X_j \in \mathrm{pa}_l(\mathbf{X})$. By contradiction, consider $X_j \notin \mathrm{pa}_l(\mathbf{X})$: being $f_l(\mathrm{pa}_l(\mathbf{X}))$ constant in X_j, then $\frac{\partial f_l(\mathrm{pa}_l(\mathbf{X}))}{\partial X_j} = 0$ for every $\mathbf{X} \in \mathbb{R}^d$ by definition of derivative. Then, $\mathbf{E}\left[\left|\frac{\partial s_l(\mathbf{X})}{\partial X_j}\right|\right] = 0$, which contradicts the hypothesis.

(ii) $X_j \in \mathrm{pa}_l(\mathbf{X}) \Rightarrow \mathbf{E}\left[\left|\frac{\partial s_l(\mathbf{X})}{\partial X_j}\right|\right] \neq 0$: we observe from Equation (8) that $\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X})) \neq 0$ *almost surely*, such that $\left|\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X}))\right| > 0$ *almost surely*. Being the probability of vanishing $\left|\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X}))\right|$ equals to zero, then the expectation $\mathbf{E}\left[\left|\frac{\partial s_l(\mathbf{X})}{\partial X_j}\right|\right]$ is equivalent to the integral $\int_{\mathcal{X}^+}\left|\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X}))\right| dP(\mathbf{X})$, with $\mathcal{X}^+ \subseteq \mathbb{R}^d$ the subset of values where $\left|\frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X}))\right|$ is strictly positive. Since the integral of a strictly positive function is strictly positive itself, then $\mathbf{E}\left[\left|\frac{\partial s_l(\mathbf{X})}{\partial X_j}\right|\right] > 0$.

■

Appendix B. DAS stability with respect to hypothesis testing threshold

In our experiments of Section 4, we fix the α threshold for hypothesis testing of non-zero mean to 0.01. Standard α values are 0.1 or lower. In absence of specific information on the data, most of causal discovery methods based on conditional independence testing employ default threshold of 0.05 (for instance, see FCI and PC implementations on `DoDiscover` and `causal-learn` well known libraries): for comparison, in Table 3 we provide experimental results of DAS with α threshold set to 0.05. We run 10 experiments with different seeds and report empirical mean and standard deviation. Experiments show that $\alpha = 0.05$ doesn't significantly affect the performance of DAS (distance to the average SHD with $\alpha = 0.01$ is always within error bars).

Appendix C. SF Experiments

In this section we present experimental results on Scale Free graphs, both on sparser (Table 4) and denser graphs (Table 5).

Table 3: DAS experiment on ER1 and ER4 data with different α cutoff thresholds for hypothesis testing.

Nodes	Threshold	SHD (ER1)	SHD (ER4)
$d = 10$	$\alpha = 0.01$	1.0 ± 0.8	27.0 ± 2.2
	$\alpha = 0.05$	0.9 ± 0.9	26.7 ± 2.0
$d = 20$	$\alpha = 0.01$	3.2 ± 1.4	56.4 ± 2.5
	$\alpha = 0.05$	3.1 ± 1.6	58.2 ± 2.5
$d = 50$	$\alpha = 0.01$	17.3 ± 3.4	156 ± 4
	$\alpha = 0.05$	17.1 ± 3.7	155 ± 5
$d = 100$	$\alpha = 0.01$	46.6 ± 9.1	355 ± 5
	$\alpha = 0.05$	47.6 ± 5.3	339 ± 6
$d = 200$	$\alpha = 0.01$	110 ± 6	691 ± 13
	$\alpha = 0.05$	114 ± 8	698 ± 9

Table 4: Experiments on SF1 data. For SCORE, CAM and GraNDAG we report results found in Rolland et al. (2022).

	Method	SHD	SID	Prec.	Rec.	Time [s]
d=10	DAS (Ours)	0.4 ± 0.7	2.2 ± 3.4	0.99 ± 0.04	0.84 ± 0.15	7.6 ± 0.1
	SCORE	0.3 ± 0.6	2.7 ± 5.8	–	–	–
	CAM	0.4 ± 0.5	2.8 ± 3.6	–	–	–
	GraNDAG	1.4 ± 1.0	12.5 ± 9.7	–	–	–
d=20	DAS (Ours)	1.9 ± 1.6	19.1 ± 7.4	0.99 ± 0.02	0.84 ± 0.11	16.6 ± 0.4
	SCORE	0.9 ± 0.9	13.8 ± 12.6	–	–	–
	CAM	0.9 ± 0.9	12.9 ± 14.0	–	–	–
	GraNDAG	3.2 ± 1.9	25.5 ± 15.6	–	–	–
d=50[5]	DAS (Ours)	10.9 ± 4.9	225.3 ± 72.1	0.96 ± 0.03	0.74 ± 0.08	53.1 ± 0.9
	SCORE	4.6 ± 2.4	132.6 ± 75.8	–	–	–
	CAM	3.6 ± 1.9	115.4 ± 72.6	–	–	–
	GraNDAG	9.2 ± 3.3	281.8 ± 129.8	–	–	–
d=100	DAS (Ours)	39.4 ± 8.0	214 ± 42	0.94 ± 0.02	0.51 ± 0.07	102 ± 5
d=200	DAS (Ours)	112 ± 11	626 ± 83	0.97 ± 0.02	0.33 ± 0.03	279 ± 6
d=500	DAS (Ours)	271 ± 15	–	0.93 ± 0.09	0.68 ± 0.01	1315 ± 9
d=1000	DAS (Ours)	910 ± 12	–	0.59 ± 0.02	0.09 ± 0.00	5442 ± 61

[4] For $d > 50$ experiments are executed only for DAS.

Appendix D. Real and semi-synthetic data

In addition to experiments on synthetic graphs, we test the empirical performance of DAS on Sachs real data (Sachs et al. (2005), a common benchmark for causal discovery on biological data with 17 edges and 853 observations) and on semi-synthetic data sampled from SynTReN generator of gene

Table 5: Experiments on SF4 data. For SCORE, CAM and GraNDAG we report results found in Rolland et al. (2022).

	Method	SHD	SID	Prec.	Rec.	Time [s]
d=10	DAS (Ours)	10.1 ± 2.39	35.7 ± 9.1	0.99 ± 0.01	0.75 ± 0.06	7.8 ± 0.1
	SCORE	4.6 ± 1.7	21.5 ± 9.6	–	–	–
	CAM	9.6 ± 2.0	40.4 ± 11.4	–	–	–
	GraNDAG	4.7 ± 1.8	23.0 ± 7.3	–	–	–
d=20	DAS (Ours)	30.4 ± 5.78	248.1 ± 20.0	0.98 ± 0.02	0.59 ± 0.06	17.1 ± 0.7
	SCORE	17.5 ± 3.5	179.2 ± 23.8	–	–	–
	CAM	26.4 ± 3.9	253.7 ± 28.8	–	–	–
	GraNDAG	14.7 ± 4.0	168.0 ± 39.2	–	–	–
d=50[6]	DAS (Ours)	115.5 ± 10.8	703.1 ± 87.5	0.97 ± 0.01	0.45 ± 0.06	51.3 ± 1.2
	SCORE	68.3 ± 3.6	1724 ± 109	–	–	–
	CAM	85.3 ± 4.2	1935 ± 99	–	–	–
	GraNDAG	63.8 ± 9.7	1677 ± 118	–	–	–
d=100	DAS (Ours)	295.3 ± 10.8	3212 ± 145	0.97 ± 0.01	0.25 ± 0.02	108 ± 2
d=200	DAS (Ours)	674.2 ± 16.6	21314 ± 891	0.95 ± 0.02	0.11 ± 0.01	283 ± 3
d=500	DAS (Ours)	1890 ± 7.1	–	0.97 ± 0.03	0.02 ± 0.01	1212 ± 31
d=1000	DAS (Ours)	3715 ± 21.3	–	0.92 ± 0.04	0.03 ± 0.01	5016 ± 53

[5] For $d > 50$ experiments are executed only for DAS.

expression records (Van den Bulcke et al. (2006)). In Table 6 we see that DAS matches SCORE performance on Sachs, whereas it is comparable to SCORE, CAM and GraNDAG on SynTReN experiments (Table 7).

Table 6: Experiments on Sachs dataset.

Method	SHD	SID
DAS (Ours)	12	45
SCORE	12	45
CAM	12	55
GraNDAG	13	47

Table 7: Experiments on SynTReN data (20 nodes, 500 samples). Empirical mean and error are computed over 10 runs on different seeds.

Method	SHD	SID
DAS (Ours)	38.0 ± 4.1	188.5 ± 71.2
SCORE	36.8 ± 4.7	193.4 ± 60.2
CAM	40.5 ± 6.8	152.3 ± 48.0
GraNDAG	34.0 ± 8.5	161.7 ± 53.4