Scalable Causal Discovery with Score Matching

Anonymous Author(s)
Affiliation
Address
email

Abstract

This paper demonstrates how to discover the whole causal graph from the second derivative of the log-likelihood in observational non-linear 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. [1] that only recovers the topological order from the score and requires an expensive pruning step to discover the edges. Our analysis leads to DAS, 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.

1 Introduction

Causal discovery from observational data is a central problem affecting virtually all scientific domains, such as biology, genetics, economics, and machine learning [2, 3, 4, 5]. Given a causal model, under suitable assumptions one can predict the effect of interventions on the system's variables having access only to observational data. In traditional causality research, algorithms to discover causal relationships from observations can be divided in three classes [6, 7]. *Constraint based* approaches like PC [8], FCI and SGS [9] test conditional independence between the variables, which is notoriously difficult [10]. *Score-based* methods like GES [11] define a suitable score function, and search for the graph that best fits the data. Usually these classes of approaches do not output a unique graph but an equivalence class. Finally, a *restricted model* class assumption, e.g. non-linear relations and additive Gaussian noise, allows to identify the Directed Acyclic Graph (DAG) underlying the observations [5, 12, 13, 14, 15]. The main challenge is that enforcing the DAG constraint has a cubic per-iteration cost in the number of variables, making the optimization the computational bottleneck.

A step towards better scalability is Rolland et al. [1] 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 prune the fully connected DAG by the method proposed in CAM [12]. The pruning step is the bottleneck of SCORE, scaling cubic in the number of nodes. In this work, we show that the Jacobian of the score 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 [1] 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.

Under review at the NeurIPS 2022 Workshop on Score-Based Methods. Do not distribute.

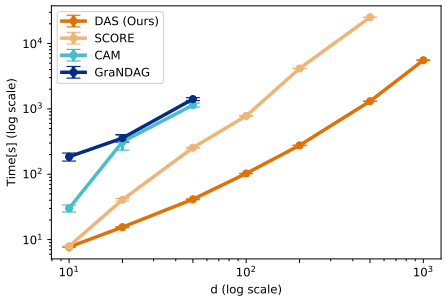

Figure 1: Execution time of different methods versus d number of nodes for dense graphs (ER4 dataset).

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 [1].
- We introduce DAS (acronym for Discovery At Scale), an algorithm for efficient and scalable causal discovery. While our approach is marginally less accurate than the state of the art ([1], [12], [13]), it improves the runtime by at least an order of magnitude in the graph size, as shown in Figure 1. 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.

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 non-linear in every component. Additionally we assume that f_i restricted to any interval is still non-linear.

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 [4, 16]:

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

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. As mentioned before 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 sorting. In our approach we select edges that satisfy constraints derived from the score function.

Topological ordering definition Let $\mathcal{G} = (\mathbf{X}, \mathcal{E})$ be a graph. An ordering of the nodes $\mathbf{X}^\pi = X_{\pi_1}, \dots, X_{\pi_d}$ is a topological ordering relative to $\mathcal{G}$ if, whenever we have $X_{\pi_i} \to X_{\pi_j} \in \mathcal{E}$, then $i < j$ [17].

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. In particular we discuss the key concepts introduced in [1], and we slightly revisit their Lemma 1 to give a more precise statement that holds *almost surely* and avoids some (trivial) corner cases.

3.1 Deriving constraints

Authors of [1] show how to efficiently estimate the score function $s(\mathbf{X}) = \nabla \log p(\mathbf{X})$ and its Jacobian exploiting the Stein identity. Then 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 the following part we show how additional constraints on the off-diagonal elements of the Jacobian matrix itself can be defined to identify directed edges in the graph.

Starting from Equation (2) we can derive $s(\mathbf{X})$ in closed form from $\log p(\mathbf{X})$ as follows:

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

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

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

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:

$$\frac{\partial s_l(\mathbf{X})}{\partial X_j} = \begin{cases} \frac{1}{\sigma_l^2} \frac{\partial f_l}{\partial X_j}(\operatorname{pa}_l(\mathbf{X})) \neq 0 & \text{if } X_j \in \operatorname{pa}_l(\mathbf{X}) \\ 0 & \text{else} \end{cases} \quad . \tag{5}$$

It is worth to notice that $\frac{1}{\sigma_l^2} \frac{\partial f_l}{\partial X_j}(\operatorname{pa}_l(\mathbf{X}))$ might still be vanishing for some values of $\operatorname{pa}_l(\mathbf{X})$ even if $X_j \in \operatorname{pa}_l(\mathbf{X})$, for instance if the function has a maximum or a minimum: given the assumption on f_l non-linear even when considered on a restricted interval, these events happen with probability zero, such that $\frac{1}{\sigma_l^2} \frac{\partial f_l}{\partial X_j}(\operatorname{pa}_l(\mathbf{X})) \neq 0$ holds *almost surely*. We prove that the condition in Equation (5) allows to derive a criterion to identify parents of a given leaf node.

Lemma 1 (Adapted from [1])**.** *Let p be the probability density function of a random variable $\mathbf{X} \in \mathbb{R}^d$ defined via non-linear 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, $\mathbf{E}\left[\left|\frac{\partial s_l(\mathbf{X})}{\partial X_j}\right|\right] \neq 0 \Leftrightarrow X_j \in \operatorname{pa}_l(\mathbf{X}), \ \forall j \in \{1, \dots, l-1\}$.*

The proof is provided in the Appendix A.2.

Novelties of Lemma 1 We now provide a discussion of the key differences of our Lemma 1 with Lemma 1 in [1]. Their formulation requires $\operatorname{Var}\left[\frac{1}{\sigma_l^2} \frac{\partial f_l}{\partial X_j}(\operatorname{pa}_l(\mathbf{X}))\right] \neq 0 \Leftrightarrow X_j \in \operatorname{pa}_l$, 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 brake, 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 this 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 $\frac{1}{\sigma_l^2} \frac{\partial f_l}{\partial X_j}(\operatorname{pa}_l(\mathbf{X}))$ for statistical error due to finite set estimates, discarding an existing edge. In practice, we rely on the sample mean of the absolute value of the Jacobian entries $\frac{1}{\sigma_l^2} \frac{\partial f_l}{\partial X_j}(\operatorname{pa}_l(\mathbf{X}))$ 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 is a lower moment thus yielding lower error (estimating variance requires estimating the mean first, such that any statistical error in the mean estimator affects the variance estimator), making it a preferable choice.

In practice we can exploit Equation (5) to reconstruct the entire graph only if a sorting $\mathbf{X}^\pi$ 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 in $\mathbf{X}^\pi \setminus \{X_l\}$ 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.

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

3.2 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. In practice, we rely on SCORE for estimating both $\mathbf{X}^\pi$ and the Jacobian matrix $J(s(\mathbf{X}))$ from a set of n observations, generated according to the model assumptions in Equation (1). Given a topological ordering, we filter edges for each node by inspecting the averages of the absolute values of the off-diagonal elements of the estimator $\hat{J}(s(\mathbf{X}))$ according to the criterion of Lemma 1. Finally, we obtain the output graph by running CAM pruning method on the resulting adjacency matrix. Note that since we use an approximation of $J(s(\mathbf{X}))$ its entries are never precisely equal to zero. To account for that we define a threshold as follow: given a leaf node X_l whose parents we are trying to identify, we consider the estimates of the l-th row of the score's Jacobian, and take the largest entry among the averages of the absolute values, call it h. We multiply h by an hyperparameter $\delta = 0.01$, such that the resulting threshold is the product $h{\cdot}\delta$, below which we consider an entry to be zero. The implementation details of DAS are illustrated in the pseudo-code of Algorithm 1 in Appendix A.1.

Experiments Figure 2 summarizes experimental results of DAS in comparison with other state of the art algorithms for causal discovery on non-linear Gaussian additive noise models: in particular we select CAM [12], SCORE [1] and GranDAG [13] as benchmarks. The causal graphs are synthetically generated using the Erdös-Renyi model [18]. 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). Figure 2 illustrates how performance is retained with respect to the competitors, while Figure 1 shows how this results are achieved with significantly lower computational time. A comprehensive presentation of all the experimental results is provided in the Appendix A.5.

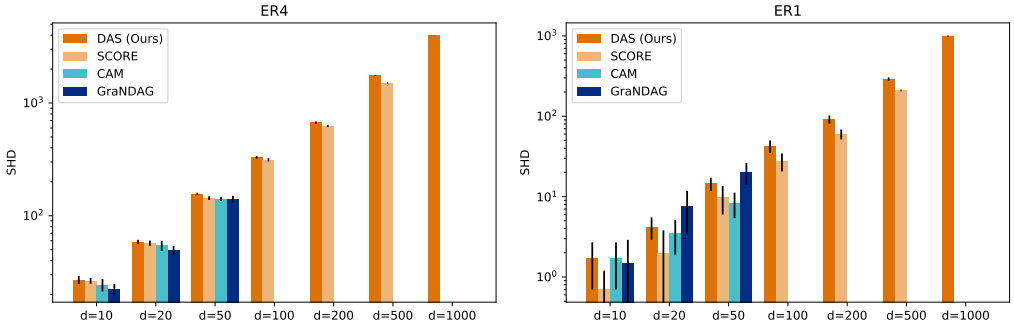

Figure 2: SHD (Structural Hamming Distance) versus d number of nodes for different methods on dense (left) and sparse (right) graphs. For higher values of d some methods are missing as they were too much time expensive to run. Number of samples is $n = 1000$.

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 sorting with SCORE involves inverting d times a $n \times n$ matrix, with a complexity $\mathcal{O}(dn^3)$. Additionally the edge search step of DAS requires iterating over the d elements of the ordering, each time inspecting a list of size $\leq d$ (see Algorithm 1) yielding a $\mathcal{O}(d^2)$ contribution. On the other hand the bottleneck of SCORE, arguably the most scalable state-of-the-art algorithm for causal discovery, is the first step in the pruning approach, namely Preliminary Neighbours Search (PNS): this procedure acts as an edge selection preliminary to CAM pruning, amounting to complexity $\mathcal{O}(nd^3)$. Therefore our use of the score for preliminary edges selection in place of PNS dramatically improves the execution time allowing to scale causal discovery in high dimensions by a factor of $\mathcal{O}(d)$.

References

[1] 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.

[2] 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.

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

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

[5] 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.

[6] 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.

[7] 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.

[8] 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.

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

[10] 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.

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

[12] 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.

[13] 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.

[14] 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.

[15] 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.

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

[17] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. Adaptive computation and machine learning. MIT Press, 2009. ISBN 9780262013192. URL https://books.google.co.in/books?id=7dzpHCHzNQ4C.

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

[19] 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.

[20] 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.

[21] 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.

A Appendix

A.1 DAS pseudo-code

The pseudo-code of Algorithm 1 provides an overview of the implementation details of DAS.

Algorithm 1 DAS

Input: data matrix $X \in \mathbb{R}^{n \times d}$
$X^\pi \leftarrow \mathrm{SCORE}(X)$ (such that $X^\pi[d]$ source node)
$A \leftarrow d \times d$ zeros adjacency matrix
remaining nodes $= 1, \dots, d$
$\delta = 0.01$ (hyperparameter)
for X_l in X^π **do**
  $\hat{J} \leftarrow \mathrm{Average}\left[\left|\frac{\partial s_l}{\partial X_j}(X)\right|\right]_{X_j \in X \setminus \{X_l\}}$ (estimate from SCORE)
  $threshold = \delta \cdot \max(\hat{J})$
  for j in *remaining nodes* **do**
    if $\hat{J}[j] > threshold$ **then**
      $A[j, l] = 1$
    end if
  end for
  Remove l-th column from X
  Remove l from *remaining nodes*
end for
Graph $\mathcal{G} \leftarrow \mathrm{CAM\text{-}pruning}(A)$

A.2 Proof of Lemma 1

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

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

$$\frac{\partial s_l(\mathbf{X})}{\partial X_j} = \frac{1}{\sigma_l^2} \frac{\partial f_l}{\partial X_j}(\mathrm{pa}_l(\mathbf{X})) \tag{6}$$

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 (5) 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 $\mathbf{E}\left[\left|\frac{\partial s_l(\mathbf{X})}{\partial X_j}\right|\right] > 0$.

□

A.3 Lemma 1 and Markov networks

In this section we provide a more detailed analysis about Lemma 1, in particular we want to show how it relates to prior work on Markov networks.

The findings of Lemma 1 on identification of the causal structure from the score function are not completely surprising in the light of previous results on Markov networks [19, 20]. Given a collection of random variables $\mathbf{X} = (X_1, X_2, \dots, X_d)$ with joint density $p(\mathbf{X})$, the information of conditional independencies bewteen 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. [19] proved how to construct a Markov graph reading the conditional independence of pairs of random variables as follow:

$$X_j \perp\!\!\!\perp X_i \mid \mathbf{X}_{\mathcal{V}\setminus\{i,j\}} \iff \partial_{ij} \log p(\mathbf{X}) = 0 \ , \tag{7}$$

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 (7), we obtain an undirected graph encoding all and only the existing conditional independencies between the variables of $\mathbf{X}$.

Equation (5) 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.

The additional pieces of information we have access to - namely that X_l has no children in the considered graph and that the sorting is given - allows to identify the direction of a detected link, since a leaf can only be the effect in a causal relation. Moreover Lemma 1 ensures correct identification of directed *v-structures* like $i \to j \leftarrow k$ that instead in the conditional independence map are *moralized* with an additional link (i, k).

A.4 Sachs dataset experiments

Table 1 reports experimental outcomes on Sachs [2], a real-world dataset with 11 variables popular for causal discovery. We see that with $\delta = 0.01$ our method matches SCORE.

Table 1: Experiments on Sachs dataset

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

A.5 Synthetic data experiments

In this section we summarize experimental outcomes of DAS on synthetic data generated with Erdös-Renyi [18] models and on Scale-free graphs. For both type of data we provide experiments on sparser (Table 4 and Table 5) and denser (Table 2 and Table 3) graphs.
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) [21] – accounting for the number of miscalculated interventional distributions that would result from the inferred graph.

From Table 2 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 accuracy 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 4. These results are directly observable in Figure 2: 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.

Table 2: Experiments on ER4 data. For CAM and GraNDAG we report results found in [1].

	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	7.7 ± 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	22.2 ± 2.6	42.0 ± 6.2	–	–	185 ± 26
d=20	DAS (Ours)	58.7 ± 2.5	214 ± 20	0.99 ± 0.00	0.27 ± 0.04	18.5 ± 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	49.3 ± 4.5	211 ± 37	–	–	357 ± 47
d=50	DAS (Ours)	156 ± 4	1460 ± 67	0.96 ± 0.02	0.24 ± 0.03	61.4 ± 0.6
	SCORE	144 ± 6	1346 ± 57	0.97 ± 0.01	0.30 ± 0.03	245 ± 5
	CAM	141 ± 6	1337 ± 94	–	–	1143 ± 79
	GraNDAG	141 ± 10	1432 ± 110	–	–	1410 ± 73
d=100	DAS (Ours)	329 ± 9	6342 ± 330	0.91 ± 0.03	0.21 ± 0.04	133 ± 1
	SCORE	313 ± 11	5965 ± 273	0.91 ± 0.03	0.27 ± 0.06	779 ± 13
	CAM	–	–	–	–	–
	GraNDAG	–	–	–	–	–
d=200	DAS (Ours)	690 ± 15	24221 ± 746	0.89 ± 0.06	0.21 ± 0.05	367 ± 6
	SCORE	626 ± 14	25707 ± 891	0.88 ± 0.04	0.30 ± 0.05	4142 ± 35
	CAM	–	–	–	–	–
	GraNDAG	–	–	–	–	–
d=500	DAS (Ours)	1761 ± 15	–	0.80 ± 0.04	0.19 ± 0.03	1608 ± 7
	SCORE[1]	1642	–	0.82	0.27	25307
	CAM	–	–	–	–	–
	GraNDAG	–	–	–	–	–
d=1000	DAS (Ours)	3951 ± 9	–	0.76 ± 0.05	0.08 ± 0.00	6539 ± 81
	SCORE	–	–	–	–	–
	CAM	–	–	–	–	–
	GraNDAG	–	–	–	–	–

[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.

Table 3: Experiments on SF4 data. For SCORE, CAM and GraNDAG we report results found in [1].

	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)	34.3 ± 5.71	237.1 ± 25.7	0.98 ± 0.02	0.59 ± 0.06	19.1 ± 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	DAS (Ours)	115.5 ± 10.8	703.1 ± 87.5	0.97 ± 0.01	0.45 ± 0.06	65.1 ± 0.4
	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)	312.8 ± 11.0	3212 ± 145	0.97 ± 0.01	0.25 ± 0.02	116 ± 1
d=200	DAS (Ours)	725.3 ± 12.5	21314 ± 891	0.95 ± 0.02	0.11 ± 0.01	302 ± 3
d=500	DAS (Ours)	1970 ± 7.1	–	0.97 ± 0.03	0.02 ± 0.01	1512 ± 31
d=1000	DAS (Ours)	3891 ± 19.5	–	0.92 ± 0.04	0.03 ± 0.01	5616 ± 53

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

Table 4: Experiments on ER1 data. For CAM and GraNDAG we report results found in [1].

	Method	SHD	SID	Prec.	Rec.	Time [s]
d=10	DAS (Ours)	1.2 ± 0.9	4.2 ± 4.5	0.97 ± 0.01	0.84 ± 0.05	7.8 ± 0.1
	SCORE	0.7 ± 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 ± 1.3	17.1 ± 9.2	0.98 ± 0.02	0.85 ± 0.03	18.7 ± 0.4
	SCORE	2.0 ± 1.8	8.3 ± 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)	14.5 ± 2.7	95.4 ± 38.5	0.96 ± 0.04	0.77 ± 0.04	62.1 ± 0.4
	SCORE	9.8 ± 3.8	69.6 ± 41.3	0.98 ± 0.01	0.87 ± 0.03	251 ± 7
	CAM	8.3 ± 2.9	53.7 ± 31.9	–	–	1143 ± 79
	GraNDAG	20.2 ± 6.1	135 ± 456	–	–	1410 ± 73
d=100	DAS (Ours)	44.6 ± 7.6	313 ± 74	0.92 ± 0.06	0.68 ± 0.04	134 ± 1
	SCORE	27.5 ± 6.9	288 ± 115	0.97 ± 0.02	0.83 ± 0.05	776 ± 12
	CAM	–	–	–	–	–
	GraNDAG	–	–	–	–	–
d=200	DAS (Ours)	101.4 ± 10.6	833 ± 227	0.88 ± 0.07	0.68 ± 0.06	365 ± 3
	SCORE	59.9 ± 8.5	495 ± 161	0.95 ± 0.03	0.85 ± 0.07	4237 ± 22
	CAM	–	–	–	–	–
	GraNDAG	–	–	–	–	–
d=500	DAS (Ours)	291 ± 13	–	0.78 ± 0.07	0.65 ± 0.05	1629 ± 7
	SCORE[2]	209	–	0.8	0.85	25115
	CAM	–	–	–	–	–
	GraNDAG	–	–	–	–	–
d=1000	DAS (Ours)	994 ± 15	–	0.59 ± 0.02	0.09 ± 0.00	6544 ± 73
	SCORE	–	–	–	–	–
	CAM	–	–	–	–	–
	GraNDAG	–	–	–	–	–

[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.

Table 5: Experiments on SF1 data. For SCORE, CAM and GraNDAG we report results found in [1].

	Method	SHD	SID	Prec.	Rec.	Time [s]
d=10	DAS (Ours)	0.8 ± 0.6	4.2 ± 2.9	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)	3.4 ± 1.7	18.7 ± 8.9	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	DAS (Ours)	13.0 ± 5.1	194.1 ± 41.3	0.96 ± 0.03	0.74 ± 0.08	61.5 ± 0.4
	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)	24.5 ± 7.09	217.9 ± 39.4	0.94 ± 0.02	0.51 ± 0.07	114 ± 2
d=200	DAS (Ours)	94.2 ± 6.2	612.1 ± 78.7	0.97 ± 0.02	0.33 ± 0.03	314 ± 3
d=500	DAS (Ours)	271.1 ± 14.9	–	0.93 ± 0.09	0.68 ± 0.01	1515 ± 9
d=1000	DAS (Ours)	910 ± 12	–	0.59 ± 0.02	0.09 ± 0.00	5842 ± 61

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