Ancestral Causal Inference

Magliacane, S.; Claassen, T.; Mooij, J.

Publication date
2017

Document Version
Author accepted manuscript

Published in
30th Annual Conference on Neural Information Processing Systems 2016

License
Unspecified

Link to publication

Citation for published version (APA):
Magliacane, S., Claassen, T., & Mooij, J. (2017). Ancestral Causal Inference. In D. D. Lee, U. von Luxburg, R. Garnett, M. Sugiyama, & I. Guyon (Eds.), *30th Annual Conference on Neural Information Processing Systems 2016: Barcelona, Spain, 5-10 December 2016* (Vol. 7, pp. 4473-4481). (Advances in Neural Information Processing Systems; Vol. 29). Curran Associates. http://papers.nips.cc/paper/6266-ancestral-causal-inference

Ancestral Causal Inference

Sara Magliacane
VU Amsterdam & University of Amsterdam
sara.magliacane@gmail.com

Tom Claassen
Radboud University Nijmegen
tomc@cs.ru.nl

Joris M. Mooij
University of Amsterdam
j.m.mooij@uva.nl

Abstract

Constraint-based causal discovery from limited data is a notoriously difficult challenge due to the many borderline independence test decisions. Several approaches to improve the reliability of the predictions by exploiting redundancy in the independence information have been proposed recently. Though promising, existing approaches can still be greatly improved in terms of accuracy and scalability. We present a novel method that reduces the combinatorial explosion of the search space by using a more coarse-grained representation of causal information, drastically reducing computation time. Additionally, we propose a method to score causal predictions based on their confidence. Crucially, our implementation also allows one to easily combine observational and interventional data and to incorporate various types of available background knowledge. We prove soundness and asymptotic consistency of our method and demonstrate that it can outperform the state-of-the-art on synthetic data, achieving a speedup of several orders of magnitude. We illustrate its practical feasibility by applying it to a challenging protein data set.

1 Introduction

Discovering causal relations from data is at the foundation of the scientific method. Traditionally, cause-effect relations have been recovered from experimental data in which the variable of interest is perturbed, but seminal work like the *do*-calculus [18] and the PC/FCI algorithms [25, 28] demonstrate that, under certain assumptions (e.g., the well-known *Causal Markov* and *Faithfulness* assumptions [25]), it is already possible to obtain substantial causal information by using only observational data.

Recently, there have been several proposals for combining observational and experimental data to discover causal relations. These causal discovery methods are usually divided into two categories: constraint-based and score-based methods. Score-based methods typically evaluate models using a penalized likelihood score, while constraint-based methods use statistical independences to express constraints over possible causal models. The advantages of constraint-based over score-based methods are the ability to handle latent confounders and selection bias naturally, and that there is no need for parametric modeling assumptions. Additionally, constraint-based methods expressed in *logic* [2, 3, 27, 9] allow for an easy integration of background knowledge, which is not trivial even for simple cases in approaches that are not based on logic [1].

Two major disadvantages of traditional constraint-based methods are: (i) vulnerability to errors in statistical independence test results, which are quite common in real-world applications, (ii) no ranking or estimation of the confidence in the causal predictions. Several approaches address the first issue and improve the reliability of constraint-based methods by exploiting redundancy in the independence information [3, 9, 27]. The idea is to assign weights to the input statements that reflect

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

their reliability, and then use a reasoning scheme that takes these weights into account. Several weighting schemes can be defined, from simple ways to attach weights to single independence statements [9], to more complicated schemes to obtain weights for combinations of independence statements [27, 3]. Unfortunately, these approaches have to sacrifice either accuracy by using a greedy method [3, 27], or scalability by formulating a discrete optimization problem on a super-exponentially large search space [9]. Additionally, the confidence estimation issue is addressed only in limited cases [19].

We propose Ancestral Causal Inference (ACI), a logic-based method that provides comparable accuracy to the best state-of-the-art constraint-based methods (e.g., [9]) for causal systems with latent variables without feedback, but improves on their scalability by using a more coarse-grained representation of causal information. Instead of representing all possible direct causal relations, in ACI we represent and reason only with ancestral relations ("indirect" causal relations), developing specialised ancestral reasoning rules. This representation, though still super-exponentially large, drastically reduces computation time. Moreover, it turns out to be very convenient, because in real-world applications the distinction between direct causal relations and ancestral relations is not always clear or necessary. Given the estimated ancestral relations, the estimation can be refined to direct causal relations by constraining standard methods to a smaller search space, if necessary.

Furthermore, we propose a method to score predictions according to their confidence. The confidence score can be thought of as an approximation to the marginal probability of an ancestral relation. Scoring predictions enables one to rank them according to their reliability, allowing for higher accuracy. This is very important for practical applications, as the low reliability of the predictions of constraint-based methods has been a major impediment to their wide-spread use.

We prove soundness and asymptotic consistency under mild conditions on the statistical tests for ACI and our scoring method. We show that ACI outperforms standard methods, like bootstrapped FCI and CFCI, in terms of accuracy, and achieves a speedup of several orders of magnitude over [9] on a synthetic dataset. We illustrate its practical feasibility by applying it to a challenging protein data set [23] that so far had only been addressed with score-based methods and observe that it successfully recovers from faithfulness violations. In this context, we showcase the flexibility of logic-based approaches by introducing weighted ancestral relation constraints that we obtain from a combination of observational and interventional data, and show that they substantially increase the reliability of the predictions. Finally, we provide an open-source version of our algorithms and the evaluation framework, which can be easily extended, at `http://github.com/caus-am/aci`.

2 Preliminaries and related work

Preliminaries We assume that the data generating process can be modeled by a causal Directed Acyclic Graph (DAG) that may contain latent variables. For simplicity we also assume that there is no selection bias. Finally, we assume that the *Causal Markov Assumption* and the *Causal Faithfulness Assumption* [25] both hold. In other words, the conditional independences in the observational distribution correspond one-to-one with the d-separations in the causal DAG. Throughout the paper we represent variables with uppercase letters, while sets of variables are denoted by boldface. All proofs are provided in the Supplementary Material.

A directed edge $X \to Y$ in the causal DAG represents a *direct causal relation* between cause X on effect Y. Intuitively, in this framework this indicates that manipulating X will produce a change in Y, while manipulating Y will have no effect on X. A more detailed discussion can be found in [25]. A sequence of directed edges $X_1 \to X_2 \to \cdots \to X_n$ is a *directed path*. If there exists a directed path from X to Y (or $X = Y$), then X is an *ancestor* of Y (denoted as $X \dashrightarrow Y$). Otherwise, X is not an ancestor of Y (denoted as $X \not\dashrightarrow Y$). For a set of variables $\boldsymbol{W}$, we write:

$$\begin{aligned} X \dashrightarrow \boldsymbol{W} &:= \exists Y \in \boldsymbol{W} : X \dashrightarrow Y, \\ X \not\dashrightarrow \boldsymbol{W} &:= \forall Y \in \boldsymbol{W} : X \not\dashrightarrow Y. \end{aligned} \tag{1}$$

We define an *ancestral structure* as any non-strict partial order on the observed variables of the DAG, i.e., any relation that satisfies the following axioms:

$$(\textit{reflexivity}) : X \dashrightarrow X, \tag{2}$$

$$(\textit{transitivity}) : X \dashrightarrow Y \ \land \ Y \dashrightarrow Z \implies X \dashrightarrow Z, \tag{3}$$

$$(\textit{antisymmetry}) : X \dashrightarrow Y \land Y \dashrightarrow X \implies X = Y. \tag{4}$$

The underlying causal DAG induces a *unique* "true" ancestral structure, which represents the transitive closure of the direct causal relations projected on the observed variables.

For disjoint sets $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{W}$ we denote conditional independence of $\boldsymbol{X}$ and $\boldsymbol{Y}$ given $\boldsymbol{W}$ as $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \mid \boldsymbol{W}$, and conditional dependence as $\boldsymbol{X} \not\perp\!\!\!\perp \boldsymbol{Y} \mid \boldsymbol{W}$. We call the cardinality $|\boldsymbol{W}|$ the *order* of the conditional (in)dependence relation. Following [2] we define a *minimal conditional independence* by:

$$X \perp\!\!\!\perp Y \,|\, \boldsymbol{W} \cup [Z] := (X \perp\!\!\!\perp Y \,|\, \boldsymbol{W} \cup Z) \wedge (X \not\perp\!\!\!\perp Y \,|\, \boldsymbol{W}),$$

and similarly, a *minimal conditional dependence* by:

$$X \not\perp\!\!\!\perp Y \,|\, \boldsymbol{W} \cup [Z] := (X \not\perp\!\!\!\perp Y \,|\, \boldsymbol{W} \cup Z) \wedge (X \perp\!\!\!\perp Y \,|\, \boldsymbol{W}).$$

The square brackets indicate that Z is needed for the (in)dependence to hold in the context of $\boldsymbol{W}$. Note that the negation of a minimal conditional independence is not a minimal conditional dependence. Minimal conditional (in)dependences are closely related to ancestral relations, as pointed out in [2]:

Lemma 1. *For disjoint (sets of) variables $X, Y, Z, \boldsymbol{W}$:*

$$X \perp\!\!\!\perp Y \,|\, \boldsymbol{W} \cup [Z] \implies Z \dashrightarrow (\{X, Y\} \cup \boldsymbol{W}), \tag{5}$$

$$X \not\perp\!\!\!\perp Y \,|\, \boldsymbol{W} \cup [Z] \implies Z \not\dashrightarrow (\{X, Y\} \cup \boldsymbol{W}). \tag{6}$$

Exploiting these rules (as well as others that will be introduced in Section 3) to deduce ancestral relations directly from (in)dependences is key to the greatly improved scalability of our method.

Related work on conflict resolution One of the earliest algorithms to deal with conflicting inputs in constraint-based causal discovery is Conservative PC [20], which adds "redundant" checks to the PC algorithm that allow it to detect inconsistencies in the inputs, and then makes only predictions that do not rely on the ambiguous inputs. The same idea can be applied to FCI, yielding Conservative FCI (CFCI) [4, 11]. BCCD (Bayesian Constraint-based Causal Discovery) [3] uses Bayesian confidence estimates to process information in decreasing order of reliability, discarding contradictory inputs as they arise. COmbINE (Causal discovery from Overlapping INtErventions) [27] is an algorithm that combines the output of FCI on several overlapping observational and experimental datasets into a single causal model by first pooling and recalibrating the independence test p-values, and then adding each constraint incrementally in order of reliability to a SAT instance. Any constraint that makes the problem unsatisfiable is discarded.

Our approach is inspired by a method presented by Hyttinen, Eberhardt and Järvisalo [9] (that we will refer to as HEJ in this paper), in which causal discovery is formulated as a constrained discrete minimization problem. Given a list of weighted independence statements, HEJ searches for the optimal causal graph $\mathcal{G}$ (an acyclic directed mixed graph, or ADMG) that minimizes the sum of the weights of the independence statements that are violated according to $\mathcal{G}$. In order to test whether a causal graph $\mathcal{G}$ induces a certain independence, the method creates an *encoding DAG of d-connection graphs*. D-connection graphs are graphs that can be obtained from a causal graph through a series of operations (conditioning, marginalization and interventions). An encoding DAG of d-connection graphs is a complex structure encoding all possible d-connection graphs and the sequence of operations that generated them from a given causal graph. This approach has been shown to correct errors in the inputs, but is computationally demanding because of the huge search space.

3 ACI: Ancestral Causal Inference

We propose Ancestral Causal Inference (ACI), a causal discovery method that accurately reconstructs ancestral structures, also in the presence of latent variables and statistical errors. ACI builds on HEJ [9], but rather than optimizing over encoding DAGs, ACI optimizes over the much simpler (but still very expressive) ancestral structures.

For n variables, the number of possible ancestral structures is the number of partial orders (`http://oeis.org/A001035`), which grows as $2^{n^2/4+o(n^2)}$ [12], while the number of DAGs can be computed with a well-known super-exponential recurrence formula (`http://oeis.org/A003024`). The number of ADMGs is $|\,\mathrm{DAG}(n)| \times 2^{n(n-1)/2}$. Although still super-exponential, the number of ancestral structures grows asymptotically much slower than the number of DAGs and even more so, ADMGs. For example, for 7 variables, there are 6×10^6 ancestral structures but already 2.3×10^{15} ADMGs, which lower bound the number of encoding DAGs of d-connection graphs used by HEJ.

New rules The rules in HEJ explicitly encode marginalization and conditioning operations on d-connection graphs, so they cannot be easily adapted to work directly with ancestral relations. Instead, ACI encodes the ancestral reasoning rules (2)–(6) and five novel causal reasoning rules:

Lemma 2. *For disjoint (sets) of variables $X, Y, U, Z, \boldsymbol{W}$:*

$$(X \perp\!\!\!\perp Y \mid \boldsymbol{Z}) \wedge (X \not\dashrightarrow \boldsymbol{Z}) \implies X \not\dashrightarrow Y, \tag{7}$$

$$X \not\perp\!\!\!\perp Y \mid \boldsymbol{W} \cup [Z] \implies X \not\perp\!\!\!\perp Z \mid \boldsymbol{W}, \tag{8}$$

$$X \perp\!\!\!\perp Y \mid \boldsymbol{W} \cup [Z] \implies X \not\perp\!\!\!\perp Z \mid \boldsymbol{W}, \tag{9}$$

$$(X \perp\!\!\!\perp Y \mid \boldsymbol{W} \cup [Z]) \wedge (X \perp\!\!\!\perp Z \mid \boldsymbol{W} \cup U) \implies (X \perp\!\!\!\perp Y \mid \boldsymbol{W} \cup U), \tag{10}$$

$$(Z \not\perp\!\!\!\perp X \mid \boldsymbol{W}) \wedge (Z \not\perp\!\!\!\perp Y \mid \boldsymbol{W}) \wedge (X \perp\!\!\!\perp Y \mid \boldsymbol{W}) \implies X \not\perp\!\!\!\perp Y \mid \boldsymbol{W} \cup Z. \tag{11}$$

We prove the soundness of the rules in the Supplementary Material. We elaborate some conjectures about their completeness in the discussion after Theorem 1 in the next Section.

Optimization of loss function We formulate causal discovery as an optimization problem where a loss function is optimized over possible causal structures. Intuitively, the loss function sums the weights of all the inputs that are violated in a candidate causal structure.

Given a list I of weighted input statements (i_j, w_j), where i_j is the input statement and w_j is the associated weight, we define the loss function as the sum of the weights of the input statements that are not satisfied in a given possible structure $W \in \mathcal{W}$, where $\mathcal{W}$ denotes the set of all possible causal structures. Causal discovery is formulated as a discrete optimization problem:

$$W^* = \arg\min_{W \in \mathcal{W}} \mathcal{L}(W; I), \tag{12}$$

$$\mathcal{L}(W; I) := \sum_{(i_j, w_j) \in I:\ W \cup \mathcal{R} \models \neg i_j} w_j, \tag{13}$$

where $W \cup \mathcal{R} \models \neg i_j$ means that input i_j is not satisfied in structure W according to the rules $\mathcal{R}$.

This general formulation includes both HEJ and ACI, which differ in the types of possible structures $\mathcal{W}$ and the rules $\mathcal{R}$. In HEJ $\mathcal{W}$ represents all possible causal graphs (specifically, acyclic directed mixed graphs, or ADMGs, in the acyclic case) and $\mathcal{R}$ are operations on d-connection graphs. In ACI $\mathcal{W}$ represent ancestral structures (defined with the rules(2)-(4)) and the rules $\mathcal{R}$ are rules (5)–(11).

Constrained optimization in ASP The constrained optimization problem in (12) can be implemented using a variety of methods. Given the complexity of the rules, a formulation in an expressive logical language that supports optimization, e.g., Answer Set Programming (ASP), is very convenient. ASP is a widely used declarative programming language based on the stable model semantics [13, 8] that has successfully been applied to several NP-hard problems. For ACI we use the state-of-the-art ASP solver `clingo 4` [7]. We provide the encoding in the Supplementary Material.

Weighting schemes ACI supports two types of input statements: conditional independences and ancestral relations. These statements can each be assigned a weight that reflects their confidence. We propose two simple approaches with the desirable properties of making ACI asymptotically consistent under mild assumptions (as described in the end of this Section), and assigning a much smaller weight to independences than to dependences (which agrees with the intuition that one is confident about a measured strong dependence, but not about independence vs. weak dependence). The approaches are:

- a *frequentist* approach, in which for any appropriate frequentist statistical test with independence as null hypothesis (resp. a non-ancestral relation), we define the weight:

  $$w = |\log p - \log \alpha|, \text{where } p = p\text{-value of the test}, \alpha = \text{significance level (e.g., 5\%)}; \tag{14}$$

- a *Bayesian* approach, in which the weight of each input statement i using data set $\mathcal{D}$ is:

  $$w = \log \frac{p(i|\mathcal{D})}{p(\neg i|\mathcal{D})} = \log \frac{p(\mathcal{D}|i)}{p(\mathcal{D}|\neg i)} \frac{p(i)}{p(\neg i)}, \tag{15}$$

  where the prior probability $p(i)$ can be used as a tuning parameter.

Given observational and interventional data, in which each intervention has a single known target (in particular, it is not a *fat-hand* intervention [5]), a simple way to obtain a weighted ancestral statement $X \dashrightarrow Y$ is with a two-sample test that tests whether the distribution of Y changes with respect to its observational distribution when intervening on X. This approach conveniently applies to various types of interventions: perfect interventions [18], soft interventions [15], mechanism changes [26], and activity interventions [17]. The two-sample test can also be implemented as an independence test that tests for the independence of Y and I_X, the indicator variable that has value 0 for observational samples and 1 for samples from the interventional distribution in which X has been intervened upon.

4 Scoring causal predictions

The constrained minimization in (12) may produce several optimal solutions, because the underlying structure may not be identifiable from the inputs. To address this issue, we propose to use the loss function (13) and score the confidence of a feature f (e.g., an ancestral relation $X \dashrightarrow Y$) as:

$$C(f) = \min_{W\in\mathcal{W}} \mathcal{L}(W; I \cup \{(\neg f, \infty)\}) - \min_{W\in\mathcal{W}} \mathcal{L}(W; I \cup \{(f, \infty)\}). \tag{16}$$

Without going into details here, we note that the confidence (16) can be interpreted as a MAP approximation of the log-odds ratio of the probability that feature f is true in a Markov Logic model:

$$\frac{\mathbb{P}(f \,|\, I, \mathcal{R})}{\mathbb{P}(\neg f \,|\, I, \mathcal{R})} = \frac{\sum_{W\in\mathcal{W}} e^{-\mathcal{L}(W;I)} 1_{W\cup\mathcal{R}\models f}}{\sum_{W\in\mathcal{W}} e^{-\mathcal{L}(W;I)} 1_{W\cup\mathcal{R}\models \neg f}} \approx \frac{\max_{W\in\mathcal{W}} e^{-\mathcal{L}(W;I\cup\{(f,\infty)\})}}{\max_{W\in\mathcal{W}} e^{-\mathcal{L}(W;I\cup\{(\neg f,\infty)\})}} = e^{C(f)}.$$

In this paper, we usually consider the features f to be ancestral relations, but the idea is more generally applicable. For example, combined with HEJ it can be used to score direct causal relations.

Soundness and completeness Our scoring method is sound for oracle inputs:

Theorem 1. *Let $\mathcal{R}$ be sound (not necessarily complete) causal reasoning rules. For any feature f, the confidence score $C(f)$ of (16) is sound for oracle inputs with infinite weights.*

Here, soundness means that $C(f) = \infty$ if f is identifiable from the inputs, $C(f) = -\infty$ if $\neg f$ is identifiable from the inputs, and $C(f) = 0$ otherwise (neither are identifiable). As features, we can consider for example ancestral relations $f = X \dashrightarrow Y$ for variables X, Y. We conjecture that the rules (2)–(11) are "order-1-complete", i.e., they allow one to deduce all (non)ancestral relations that are identifiable from oracle conditional independences of order ≤ 1 in observational data. For higher-order inputs additional rules can be derived. However, our primary interest in this work is improving computation time and accuracy, and we are willing to sacrifice completeness. A more detailed study of the completeness properties is left as future work.

Asymptotic consistency Denote the number of samples by N. For the frequentist weights in (14), we assume that the statistical tests are consistent in the following sense:

$$\log p_N - \log \alpha_N \overset{P}{\to} \begin{cases} -\infty & H_1 \\ +\infty & H_0, \end{cases} \tag{17}$$

as $N \to \infty$, where the null hypothesis H_0 is independence/nonancestral relation and the alternative hypothesis H_1 is dependence/ancestral relation. Note that we need to choose a sample-size dependent threshold α_N such that $\alpha_N \to 0$ at a suitable rate. Kalisch and Bühlmann [10] show how this can be done for partial correlation tests under the assumption that the distribution is multivariate Gaussian.

For the Bayesian weighting scheme in (15), we assume that for $N \to \infty$,

$$w_N \overset{P}{\to} \begin{cases} -\infty & \text{if } i \text{ is true} \\ +\infty & \text{if } i \text{ is false.} \end{cases} \tag{18}$$

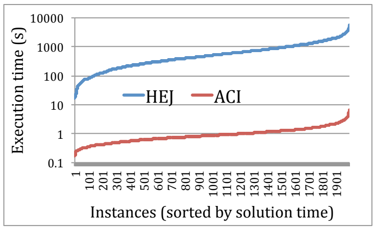
This will hold (as long as there is no model misspecification) under mild technical conditions for finite-dimensional exponential family models. In both cases, the probability of a type I or type II error will converge to 0, and in addition, the corresponding weight will converge to ∞.

Theorem 2. *Let $\mathcal{R}$ be sound (not necessarily complete) causal reasoning rules. For any feature f, the confidence score $C(f)$ of (16) is asymptotically consistent under assumption (17) or (18).*

Here, "asymptotically consistent" means that the confidence score $C(f) \to \infty$ in probability if f is identifiably true, $C(f) \to -\infty$ in probability if f is identifiably false, and $C(f) \to 0$ in probability otherwise.

Average execution time (s)					
n	c	ACI	HEJ	BAFCI	BACFCI
6	1	0.21	12.09	8.39	12.51
6	4	1.66	432.67	11.10	16.36
7	1	1.03	715.74	9.37	15.12
8	1	9.74	≥ 2500	13.71	21.71
9	1	146.66	$\gg 2500$	18.28	28.51

(a)



(b)

Figure 1: Execution time comparison on synthetic data for the frequentist test on 2000 synthetic models: (a) average execution time for different combinations of number of variables n and max. order c; (b) detailed plot of execution times for $n = 7, c = 1$ (logarithmic scale).

5 Evaluation

In this section we report evaluations on synthetically generated data and an application on a real dataset. Crucially, in causal discovery precision is often more important than recall. In many real-world applications, discovering a few high-confidence causal relations is more useful than finding every possible causal relation, as reflected in recently proposed algorithms, e.g., [19].

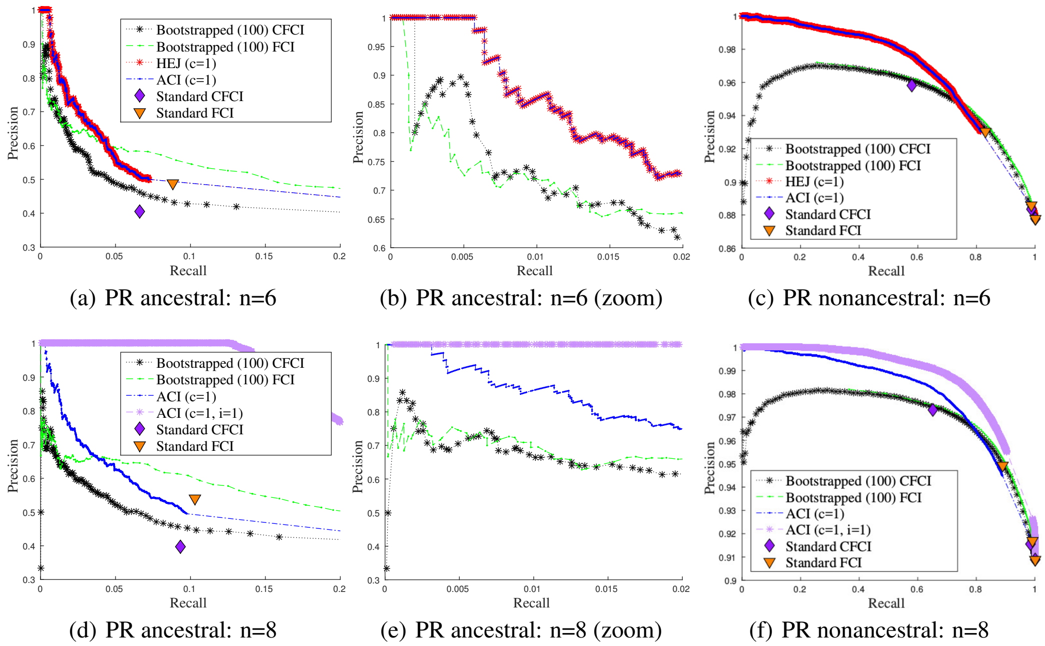
Compared methods We compare the predictions of ACI and of the acyclic causally insufficient version of HEJ [9], when used in combination with our scoring method (16). We also evaluate two standard methods: Anytime FCI [24, 28] and Anytime CFCI [4], as implemented in the `pcalg` R package [11]. We use the anytime versions of (C)FCI because they allow for independence test results up to a certain order. We obtain the ancestral relations from the output PAG using Theorem 3.1 from [22]. (Anytime) FCI and CFCI do not rank their predictions, but only predict the type of relation: ancestral (which we convert to +1), non-ancestral (-1) and unknown (0). To get a scoring of the predictions, we also compare with bootstrapped versions of Anytime FCI and Anytime CFCI. We perform the bootstrap by repeating the following procedure 100 times: sample randomly half of the data, perform the independence tests, run Anytime (C)FCI. From the 100 output PAGs we extract the ancestral predictions and average them. We refer to these methods as BA(C)FCI. For a fair comparison, we use the same independence tests and thresholds for all methods.

Synthetic data We simulate the data using the simulator from HEJ [9]: for each experimental condition (e.g., a given number of variables n and order c), we generate randomly M linear acyclic models with latent variables and Gaussian noise and sample $N = 500$ data points. We then perform independence tests up to order c and weight the (in)dependence statements using the weighting schemes described in Section 3. For the frequentist weights we use tests based on partial correlations and Fisher's z-transform to obtain approximate p-values (see, e.g., [10]) with significance level $\alpha = 0.05$. For the Bayesian weights, we use the Bayesian test for conditional independence presented in [14] as implemented by HEJ with a prior probability of 0.1 for independence.

In Figure 1(a) we show the average execution times on a single core of a 2.80GHz CPU for different combinations of n and c, while in Figure 1(b) we show the execution times for $n = 7, c = 1$, sorting the execution times in ascending order. For 7 variables ACI is almost 3 orders of magnitude faster than HEJ, and the difference grows exponentially as n increases. For 8 variables HEJ can complete only four of the first 40 simulated models before the timeout of 2500s. For reference we add the execution time for bootstrapped anytime FCI and CFCI.

In Figure 2 we show the accuracy of the predictions with precision-recall (PR) curves for both ancestral ($X \dashrightarrow Y$) and nonancestral ($X \not\dashrightarrow Y$) relations, in different settings. In this Figure, for ACI and HEJ all of the results are computed using frequentist weights and, as in all evaluations, our scoring method (16). While for these two methods we use $c = 1$, for (bootstrapped) (C)FCI we use all possible independence test results ($c = n - 2$). In this case, the anytime versions of FCI and CFCI are equivalent to the standard versions of FCI and CFCI. Since the overall results are similar, we report the results with the Bayesian weights in the Supplementary Material.

In the first row of Figure 2, we show the setting with $n = 6$ variables. The performances of HEJ and ACI coincide, performing significantly better for nonancestral predictions and the top ancestral



(a) PR ancestral: n=6

(b) PR ancestral: n=6 (zoom)

(c) PR nonancestral: n=6

(d) PR ancestral: n=8

(e) PR ancestral: n=8 (zoom)

(f) PR nonancestral: n=8

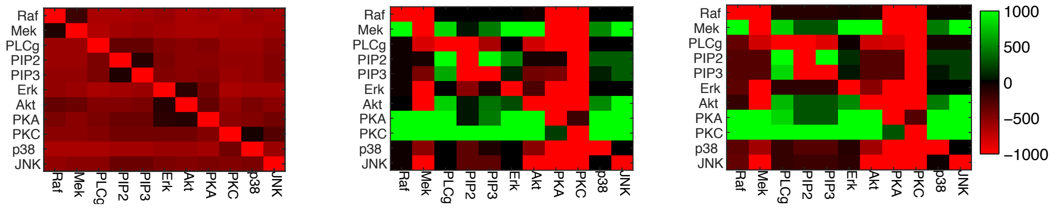
Figure 2: Accuracy on synthetic data for the two prediction tasks (ancestral and nonancestral relations) using the frequentist test with $\alpha = 0.05$. The left column shows the precision-recall curve for ancestral predictions, the middle column shows a zoomed-in version in the interval (0,0.02), while the right column shows the nonancestral predictions.

predictions (see zoomed-in version in Figure 2(b)). This is remarkable, as HEJ and ACI use only independence test results up to order $c = 1$, in contrast with (C)FCI which uses independence test results of all orders. Interestingly, the two discrete optimization algorithms do not seem to benefit much from higher order independence tests, thus we omit them from the plots (although we add the graphs in the Supplementary Material). Instead, bootstrapping traditional methods, oblivious to the (in)dependence weights, seems to produce surprisingly good results. Nevertheless, both ACI and HEJ outperform bootstrapped FCI and CFCI, suggesting these methods achieve nontrivial error-correction.

In the second row of Figure 2, we show the setting with 8 variables. In this setting HEJ is too slow. In addition to the previous plot, we plot the accuracy of ACI when there is oracle background knowledge on the descendants of one variable ($i = 1$). This setting simulates the effect of using interventional data, and we can see that the performance of ACI improves significantly, especially in the ancestral preditions. The performance of (bootstrapped) FCI and CFCI is limited by the fact that they cannot take advantage of this background knowledge, except with complicated postprocessing [1].

Application on real data We consider the challenging task of reconstructing a signalling network from flow cytometry data [23] under different experimental conditions. Here we consider one experimental condition as the observational setting and seven others as interventional settings. More details and more evaluations are reported in the Supplementary Material. In contrast to likelihood-based approaches like [23, 5, 17, 21], in our approach we do not need to model the interventions quantitatively. We only need to know the intervention *targets*, while the intervention *types* do not matter. Another advantage of our approach is that it takes into account possible latent variables.

We use a t-test to test for each intervention and for each variable whether its distribution changes with respect to the observational condition. We use the p-values of these tests as in (14) in order to obtain weighted ancestral relations that are used as input (with threshold $\alpha = 0.05$). For example, if adding U0126 (a MEK inhibitor) changes the distribution of RAF significantly with respect to the observational baseline, we get a weighted ancestral relation MEK$\dashrightarrow$RAF. In addition, we use partial correlations up to order 1 (tested in the observational data only) to obtain weighted independences used as input. We use ACI with (16) to score the ancestral relations for each ordered pair of variables. The main results are illustrated in Figure 3, where we compare ACI with bootstrapped anytime CFCI



(a) Bootstrapped (100) anytime CFCI (input: independences of order ≤ 1)

(b) ACI (input: weighted ancestral relations)

(c) ACI (input: independences of order ≤ 1, weighted ancestral relations)

Figure 3: Results for flow cytometry dataset. Each matrix represents the ancestral relations, where each row represents a cause and each column an effect. The colors encode the confidence levels: green is positive, black is unknown, while red is negative. The intensity of the color represents the degree of confidence. For example, ACI identifies MEK to be a cause of RAF with high confidence.

under different inputs. The output for boostrapped anytime FCI is similar, so we report it only in the Supplementary Material. Algorithms like (anytime) (C)FCI can only use the independences in the observational data as input and therefore miss the strongest signal, *weighted ancestral relations*, which are obtained by comparing interventional with observational data. In the Supplementary Material, we compare also with other methods ([19], [17]). Interestingly, as we show there, our results are similar to the best acyclic model reconstructed by the score-based method from [17]. As for other constraint-based methods, HEJ is computationally unfeasible in this setting, while COMBINE assumes perfect interventions (while this dataset contains mostly activity interventions).

Notably, our algorithms can correctly recover from faithfulness violations (e.g., the independence between MEK and ERK), because they take into account the weight of the input statements (the weight of the independence is considerably smaller than that of the ancestral relation, which corresponds with a quite significant change in distribution). In contrast, methods that start by reconstructing the skeleton, like (anytime) (C)FCI, would decide that MEK and ERK are nonadjacent, and are unable to recover from that erroneous decision. This illustrates another advantage of our approach.

6 Discussion and conclusions

As we have shown, ancestral structures are very well-suited for causal discovery. They offer a natural way to incorporate background causal knowledge, e.g., from experimental data, and allow a huge computational advantage over existing representations for error-correcting algorithms, such as [9]. When needed, ancestral structures can be mapped to a finer-grained representation with direct causal relations, as we sketch in the Supplementary Material. Furthermore, confidence estimates on causal predictions are extremely helpful in practice, and can significantly boost the reliability of the output. Although standard methods, like bootstrapping (C)FCI, already provide reasonable estimates, methods that take into account the confidence in the inputs, as the one presented here, can lead to further improvements of the reliability of causal relations inferred from data.

Strangely (or fortunately) enough, neither of the optimization methods seems to improve much with higher order independence test results. We conjecture that this may happen because our loss function essentially assumes that the test results are independent from another (which is not true). Finding a way to take this into account in the loss function may further improve the achievable accuracy, but such an extension may not be straightforward.

Acknowledgments

SM and JMM were supported by NWO, the Netherlands Organization for Scientific Research (VIDI grant 639.072.410). SM was also supported by the Dutch programme COMMIT/ under the Data2Semantics project. TC was supported by NWO grant 612.001.202 (MoCoCaDi), and EU-FP7 grant agreement n.603016 (MATRICS). We also thank Sofia Triantafillou for her feedback, especially for pointing out the correct way to read ancestral relations from a PAG.

7 Proofs

7.1 ACI causal reasoning rules

We give a combined proof of all the ACI reasoning rules. Note that the numbering of the rules here is different from the numbering used in the main paper.

Lemma 3. *For X, Y, Z, U, $\boldsymbol{W}$ disjoint (sets of) variables:*

1. $(X \perp\!\!\!\perp Y \mid \boldsymbol{W}) \wedge (X \not\dashrightarrow \boldsymbol{W}) \implies X \not\dashrightarrow Y$
2. $X \not\perp\!\!\!\perp Y \mid \boldsymbol{W} \cup [Z] \implies (X \not\perp\!\!\!\perp Z \mid \boldsymbol{W}) \wedge (Z \not\dashrightarrow \{X,Y\} \cup \boldsymbol{W})$
3. $X \perp\!\!\!\perp Y \mid \boldsymbol{W} \cup [Z] \implies (X \not\perp\!\!\!\perp Z \mid \boldsymbol{W}) \wedge (Z \dashrightarrow \{X,Y\} \cup \boldsymbol{W})$
4. $(X \perp\!\!\!\perp Y \mid \boldsymbol{W} \cup [Z]) \wedge (X \perp\!\!\!\perp Z \mid \boldsymbol{W} \cup U) \implies (X \perp\!\!\!\perp Y \mid \boldsymbol{W} \cup U)$
5. $(Z \not\perp\!\!\!\perp X \mid \boldsymbol{W}) \wedge (Z \not\perp\!\!\!\perp Y \mid \boldsymbol{W}) \wedge (X \perp\!\!\!\perp Y \mid \boldsymbol{W}) \implies X \not\perp\!\!\!\perp Y \mid \boldsymbol{W} \cup Z$

Proof. We assume a causal DAG with possible latent variables, the causal Markov assumption, and the causal faithfulness assumption.

1. This is a strengthened version of rule $\mathcal{R}2$(i) in [6]: note that the additional assumptions made there ($Y \not\dashrightarrow \boldsymbol{W}$, $Y \not\dashrightarrow X$) are redundant and not actually used in their proof. For completeness, we give the proof here. If $X \dashrightarrow Y$, then there is a directed path from X to Y. As all paths between X and Y are blocked by $\mathbf{W}$, the directed path from X to Y must contain a node $W \in \boldsymbol{W}$. Hence $X \dashrightarrow W$, a contradiction with $X \not\dashrightarrow \mathbf{W}$.

2. If $X \not\perp\!\!\!\perp Y \mid \boldsymbol{W} \cup [Z]$ then there exists a path π between X and Y such that each noncollider on π is not in $\boldsymbol{W} \cup \{Z\}$, every collider on π is ancestor of $\boldsymbol{W} \cup \{Z\}$, and there exists a collider on π that is ancestor of Z but not of $\boldsymbol{W}$. Let C be the collider on π closest to X that is ancestor of Z but not of $\boldsymbol{W}$. Note that

   (a) The path $X \cdots C \to \cdots \to Z$ is d-connected given $\boldsymbol{W}$.
   (b) $Z \not\dashrightarrow \boldsymbol{W}$ (because otherwise $C \dashrightarrow Z \dashrightarrow \boldsymbol{W}$, a contradiction).
   (c) $Z \not\dashrightarrow Y$ (because otherwise the path $X \cdots C \to \cdots \to Z \to \cdots \to Y$ would be d-connected given $\boldsymbol{W}$, a contradiction).

   Hence we conclude that $X \not\perp\!\!\!\perp Z \mid \boldsymbol{W}$, $Z \not\dashrightarrow \boldsymbol{W}$, $Z \not\dashrightarrow Y$, and by symmetry also $Z \not\dashrightarrow X$.

3. Suppose $X \perp\!\!\!\perp Y \mid \boldsymbol{W} \cup [Z]$. Then there exists a path π between X and Y, such that each noncollider on π is not in $\boldsymbol{W}$, each collider on π is an ancestor of $\boldsymbol{W}$, and Z is a noncollider on π. Note that

   (a) The subpath $X \cdots Z$ must be d-connected given $\boldsymbol{W}$.
   (b) Z has at least one outgoing edge on π. Follow this edge further along π until reaching either X, Y, or the first collider. When a collider is reached, follow the directed path to $\boldsymbol{W}$. Hence there is a directed path from Z to X or Y or to $\boldsymbol{W}$, i.e., $Z \dashrightarrow \{X,Y\} \cup \boldsymbol{W}$.

4. If in addition, $X \perp\!\!\!\perp Z \mid \boldsymbol{W} \cup U$, then U must be a noncollider on the subpath $X \cdots Z$. Therefore, $X \perp\!\!\!\perp Y \mid \boldsymbol{W} \cup U$.

5. Assume that $Z \not\perp\!\!\!\perp X \mid \boldsymbol{W}$ and $Z \not\perp\!\!\!\perp Y \mid \boldsymbol{W}$. Then there must be paths π between Z and X and ρ between Z and Y such that each noncollider is not in $\boldsymbol{W}$ and each collider is ancestor of $\boldsymbol{W}$. Let U be the node on π closest to X that is also on ρ (this could be Z). Then we have a path $X \cdots U \cdots Y$ such that each collider (except U) is ancestor of $\boldsymbol{W}$ and each noncollider (except U) is not in $\boldsymbol{W}$. This path must be blocked given $\boldsymbol{W}$ as $X \perp\!\!\!\perp Y \mid \boldsymbol{W}$. If U would be a noncollider on this path, it would need to be in $\boldsymbol{W}$ in order to block it; however, it must then also be a noncollider on π or ρ and hence cannot be in $\boldsymbol{W}$. Therefore, U must be a collider on this path and cannot be ancestor of $\boldsymbol{W}$. We have to show that U is ancestor of Z. If U were a collider on π or ρ, it would be ancestor of $\boldsymbol{W}$, a contradiction. Hence U must have an outgoing arrow pointing towards Z on π and ρ. If we encounter a collider following the directed edges, we get a contradiction, as that collider, and hence U, would be ancestor of $\boldsymbol{W}$. Hence U is ancestor of Z, and therefore, $X \not\perp\!\!\!\perp Y \mid \boldsymbol{W} \cup Z$.

□

7.2 Soundness

Theorem 3. *Let $\mathcal{R}$ be sound (not necessarily complete) causal reasoning rules. For any feature f, the confidence score $C(f)$ of (16) is sound for oracle inputs with infinite weights, i.e., $C(f) = \infty$ if f is identifiable from the inputs, $C(f) = -\infty$ if $\neg f$ is identifiable from the inputs, and $C(f) = 0$ otherwise (neither are identifiable).*

Proof. We assume that the data generating process is described by a causal DAG which may contain additional latent variables, and that the distributions are faithful to the DAG. The theorem then follows directly from the soundness of the rules and the soundness of logical reasoning. □

7.3 Asymptotic consistency of scoring method

Theorem 4. *Let $\mathcal{R}$ be sound (not necessarily complete) causal reasoning rules. For any feature f, the confidence score $C(f)$ of (16) is asymptotically consistent under assumption (14) or (15) in the main paper, i.e.,*

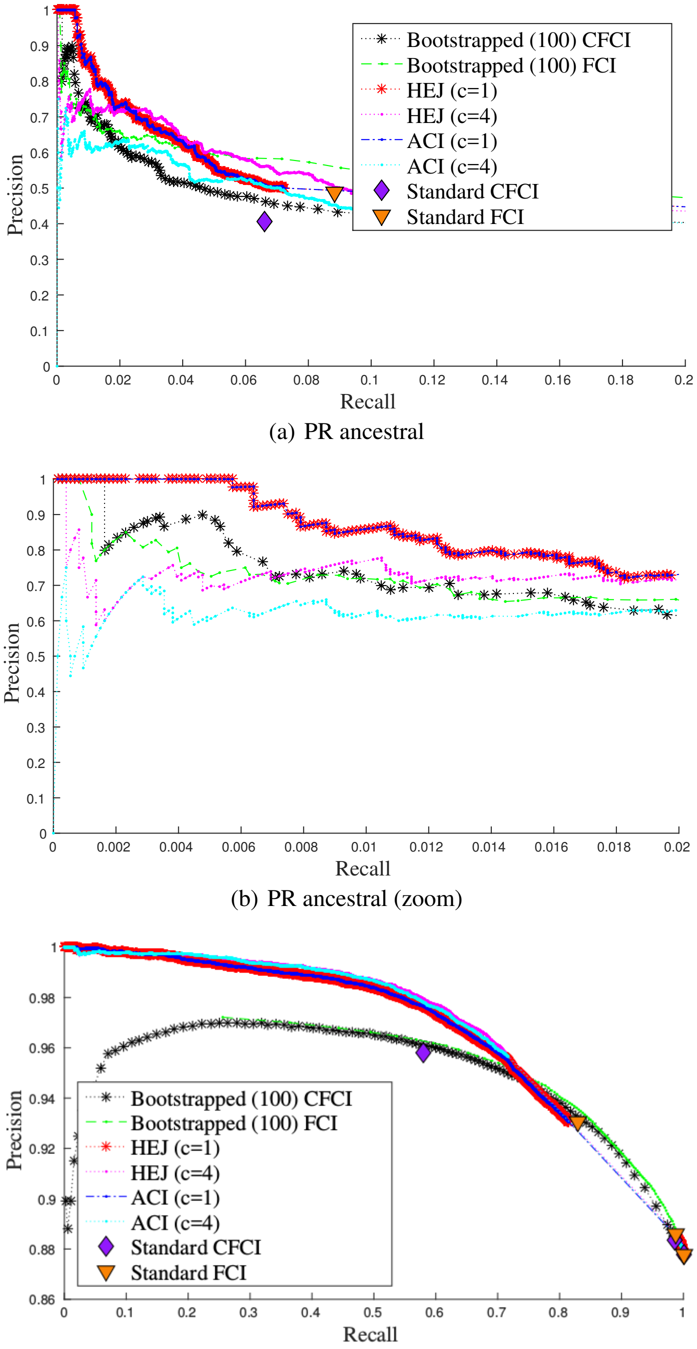
- $C(f) \to \infty$ *in probability if f is identifiably true,*
- $C(f) \to -\infty$ *in probability if f is identifiably false,*
- $C(f) \to 0$ *in probability otherwise (neither are identifiable).*

Proof. As the number of statistical tests is fixed (or at least bounded from above), the probability of *any* error in the test results converges to 0 asymptotically. The loss function of all structures that do not correspond with the properties of the true causal DAG converges to $+\infty$ in probability, whereas the loss function of all structures that are compatible with properties of the true causal DAG converges to 0 in probability. □

8 Additional results on synthetic data

In Figures 4 and 5 we show the performance of ACI and HEJ [9] for higher order independence test results ($c = 4$). As in the main paper, for (bootstrapped) FCI and CFCI we use $c = 4$, because it gives the best predictions for these methods. In Figure 4 we report more accuracy results on the frequentist test with $\alpha = 0.05$, the same setting as Figure 2 (a-c) in the main paper. As we see, the performances of ACI and HEJ do not really improve with higher order but actually seem to deteriorate.

In Figure 5 we report accuracy results on synthetic data also for the Bayesian test described in the main paper, with prior probability of independence $p = 0.1$. Using the Bayesian test does not change the overall conclusions: ACI and HEJ overlap for order $c = 1$ and they perform better than bootstrapped (C)FCI.



(a) PR ancestral

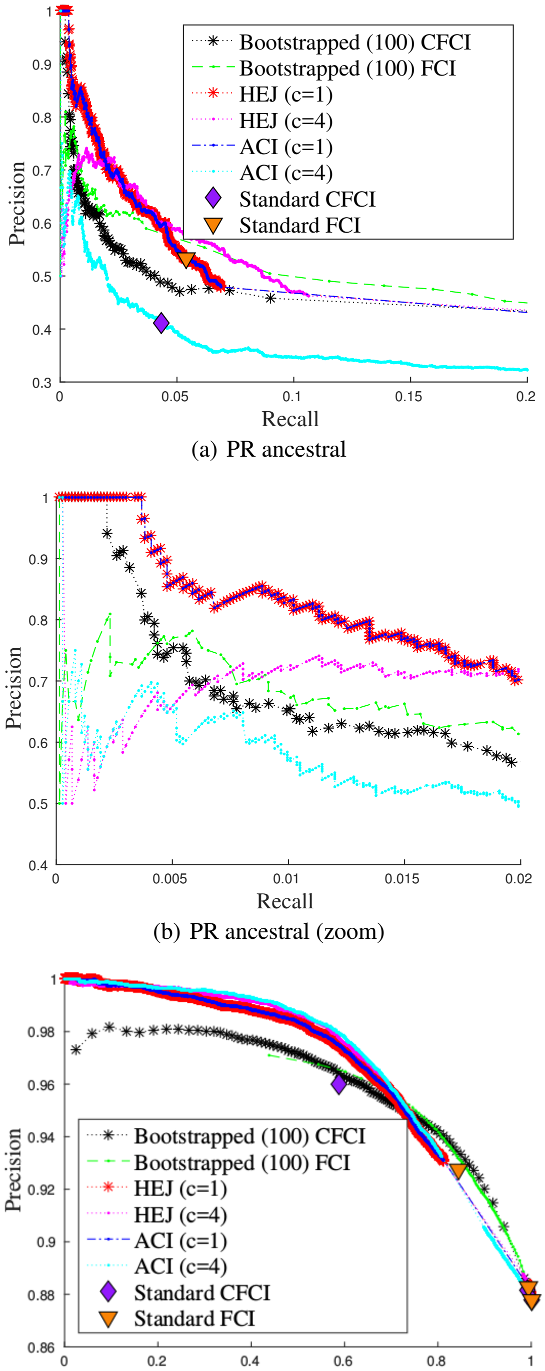
(b) PR ancestral (zoom)

(c) PR nonancestral

Figure 4: Synthetic data: accuracy for the two prediction tasks (ancestral and nonancestral relations) for $n = 6$ variables using the frequentist test with $\alpha = 0.05$, also for higher order c.

9 Application on real data

We provide more details and more results on the real-world dataset that was briefly described in the main paper, the flow cytometry data [23]. The data consists of simultaneous measurements of expression levels of 11 biochemical agents in individual cells of the human immune system under 14 different experimental conditions.



(a) PR ancestral

(b) PR ancestral (zoom)

(c) PR nonancestral

Figure 5: Synthetic data: accuracy for the two prediction tasks (ancestral and nonancestral relations) for $n = 6$ variables using the Bayesian test with prior probability of independence $p = 0.1$.

9.1 Experimental conditions

The experimental conditions can be grouped into two batches of 8 conditions each that have very similar interventions:

- “no-ICAM”, used in the main paper and commonly used in the literature;

Table 1: Reagents used in the various experimental conditions in [23] and corresponding intervention types and targets. The intervention types and targets are based on (our interpretation of) biological background knowledge. The upper table describes the “no-ICAM” batch of conditions that is most commonly used in the literature. The lower table describes the additional “ICAM” batch of conditions that we also use here.

no-ICAM:

Reagents			Intervention	
α-CD3, α-CD28	ICAM-2	Additional	Target	Type
+	-	-	-	(observational)
+	-	AKT inhibitor	AKT	activity
+	-	G0076	PKC	activity
+	-	Psitectorigenin	PIP2	abundance
+	-	U0126	MEK	activity
+	-	LY294002	PIP2/PIP3	mechanism change
-	-	PMA	PKC	activity + fat-hand
-	-	β2CAMP	PKA	activity + fat-hand

ICAM:

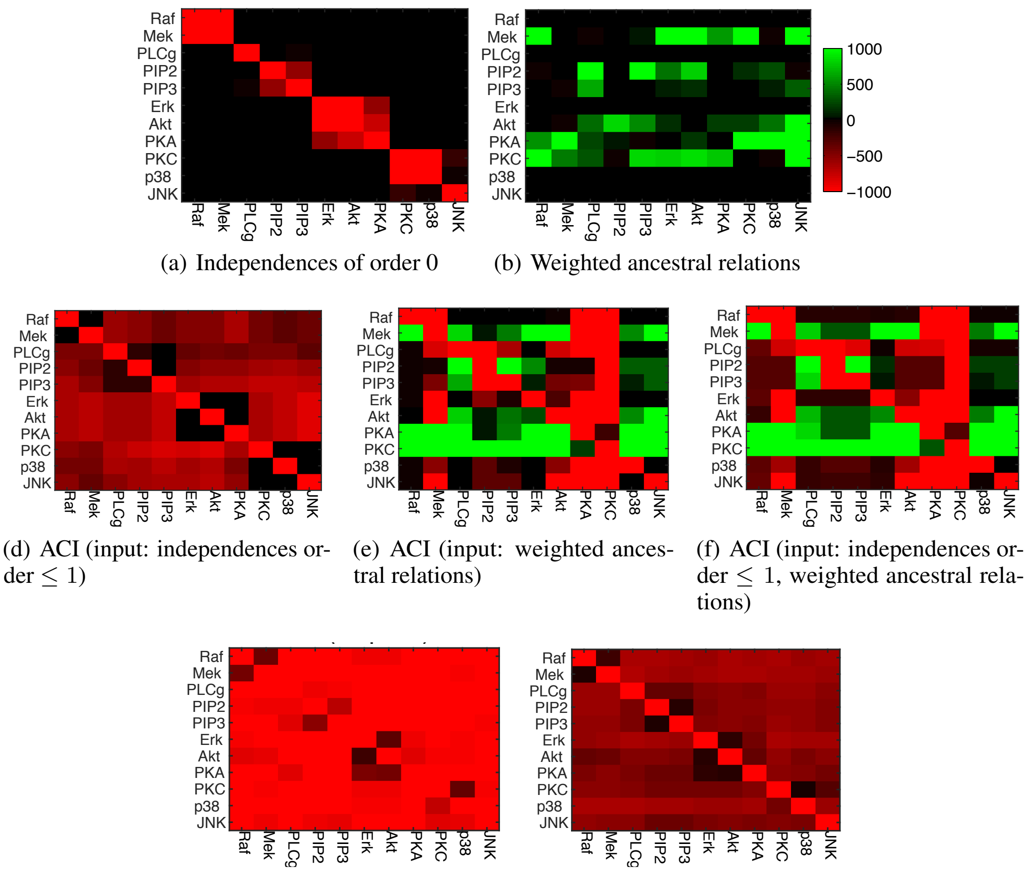
Reagents			Intervention	
α-CD3, α-CD28	ICAM-2	Additional	Target	Type
+	+	-	-	(observational)
+	+	AKT inhibitor	AKT	activity
+	+	G0076	PKC	activity
+	+	Psitectorigenin	PIP2	abundance
+	+	U0126	MEK	activity
+	+	LY294002	PIP2/PIP3	mechanism change
-	-	PMA	PKC	activity + fat-hand
-	-	β2CAMP	PKA	activity + fat-hand

- “ICAM”, where Intercellular Adhesion Protein-2 (ICAM-2) was added (except when PMA or β2CAMP was added).

For each batch of 8 conditions, the experimenters added α-CD3 and α-CD28 to activate the signaling network in 6 out of 8 conditions. For the remaining two conditions (PMA and β2CAMP), α-CD3 and α-CD28 were not added (and neither was ICAM-2). We can consider the *absence* of these stimuli as a global intervention relative to the observational baseline (where α-CD3 and α-CD28 are present, and in addition ICAM-2 is present in the ICAM batch). For each batch (ICAM and no-ICAM), we can consider an observational dataset and 7 interventional datasets with different activators and inhibitors added to the cells, as described in Table 1. Note that the datasets from the last two conditions are the same in both settings. For more information about intervention types, see [17].

In this paper, we ignore the fact that in the last two interventional datasets in each batch (PMA and β2CAMP) there is also a global intervention. Ignoring the global intervention allows us to compute the weighted ancestral relations, since we consider any variable that changes its distribution with respect to the observational condition to be an effect of the main target of the intervention (PKC for PMA and PKA for β2CAMP). This is in line with previous work [23, 17]. Also, we consider only PIP3 as the main target of the LY294002 intervention, based on the consensus network [23], even though in [17] both PIP2 and PIP3 are considered to be targets of this intervention. In future work, we plan to extend ACI in order to address the task of learning the intervention targets from data, as done by [5] for a score-based approach.

In the main paper we provide some results for the most commonly used no-ICAM batch of experimental conditions. Below we report additional results on the same batch. Moreover, we provide results for causal discovery on the ICAM batch, which are quite consistent with the no-ICAM batch. Finally, we compare with other methods that were applied to this dataset, especially with a score-based approach ([17]) that shows surprisingly similar results to ACI, although it uses a very different method.



(a) Independences of order 0

(b) Weighted ancestral relations

(d) ACI (input: independences order ≤ 1)

(e) ACI (input: weighted ancestral relations)

(f) ACI (input: independences order ≤ 1, weighted ancestral relations)

(g) Bootstrapped (100) anytime FCI (input: independences order ≤ 1)

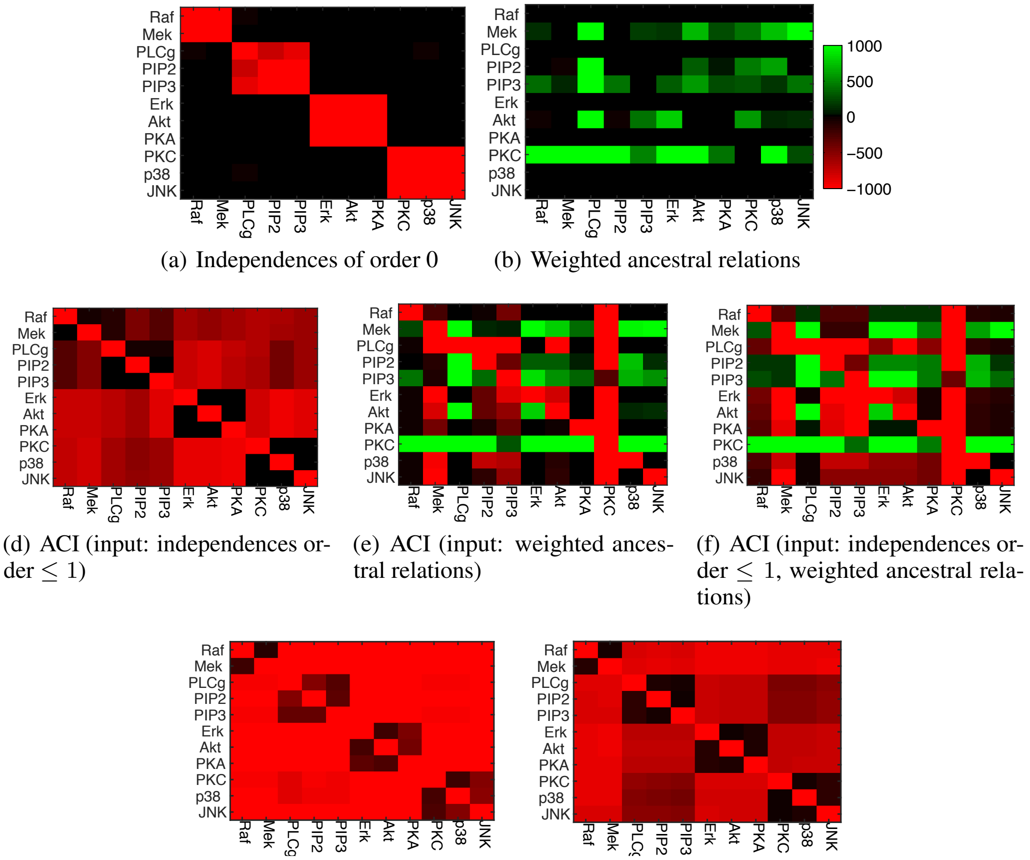
(h) Bootstrapped (100) anytime CFCI (input: independences order ≤ 1)

Figure 6: Results on flow cytometry dataset, no-ICAM batch. The top row represents some of the possible inputs: weighted independences of order 0 from the observational dataset (the inputs include also order 1 test results, but these are not visualized here) and weighted ancestral relations recovered from comparing the interventional datasets with the observational data. In the bottom two rows each matrix represents the ancestral relations that are estimated using different inputs and different methods (ACI, bootstrapped anytime FCI or CFCI). Each row represents a cause, while the columns are the effects. The colors encodes the confidence levels, green is positive, black is unknown, while red is negative. The intensity of the color represents the degree of confidence.

9.2 Results on no-ICAM batch

In Figure 6 we provide additional results for the no-ICAM batch. In the first row we show some of the possible inputs: weighted independences (in this case partial correlations) from observational data and weighted ancestral relations from comparing the interventional datasets with the observational data. Specifically, we consider as inputs only independences up to order 1 (but only independences of order 0 are visualized in the figure). The color encodes the weight of the independence. As an example, the heatmap shows that Raf and Mek are strongly dependent.

For the weighted ancestral relations, in Figure 6 we plot a matrix in which each row represents a cause, while the columns are the effects. As described in the main paper we use a t-test to test for each intervention and for each variable whether its distribution changes with respect to the observational condition. We use the biological knowledge summarised in Table 1 to define the intervention target, which is then considered the putative "cause". Then we use the p-values of these tests and a threshold $\alpha = 0.05$ to obtain the weights of the ancestral relations, similarly to what is proposed in the main



(a) Independences of order 0

(b) Weighted ancestral relations

(d) ACI (input: independences order ≤ 1)

(e) ACI (input: weighted ancestral relations)

(f) ACI (input: independences order ≤ 1, weighted ancestral relations)

(g) Bootstrapped (100) anytime FCI(input: independences order ≤ 1)

(h) Bootstrapped (100) anytime CFCI (input: independences order ≤ 1)

Figure 7: Results on flow cytometry dataset, ICAM batch. Same comparison as in Figure 6, but for the ICAM batch.

paper for the frequentist weights for the independence tests:

$$w = |\log p - \log \alpha|.$$

For example, if adding U0126 (which is known to be a MEK inhibitor) changes the distribution of RAF with $p = 0.01$ with respect to the observational baseline, we get a weighted ancestral relation (MEK⇢RAF, 1.609).

9.3 ICAM batch

In Figure 7 we show the results for the ICAM setting. These results are very similar to the results for the no-ICAM batch (see also Figure 8), showing that the predicted ancestral relations are robust. In particular it is clear that also for the ICAM batch, weighted ancestral relations are a very strong signal, and that methods that can exploit them (e.g., ACI) have a distinct advantage over methods that cannot (e.g., FCI and CFCI).

In general, in both settings there appear to be various faithfulness violations. For example, it is well-known that MEK causes ERK, yet in the observational data these two variables are independent. Nevertheless, we can see in the data that an intervention on MEK leads to a change of ERK, as expected. It is interesting to note that our approach can correctly recover from this faithfulness violation because it takes into account the weight of the input statements (note that the weight of the independence is smaller than that of the ancestral relation, which corresponds with a quite significant change in distribution). In contrast, methods that start by reconstructing the skeleton (like (C)FCI or

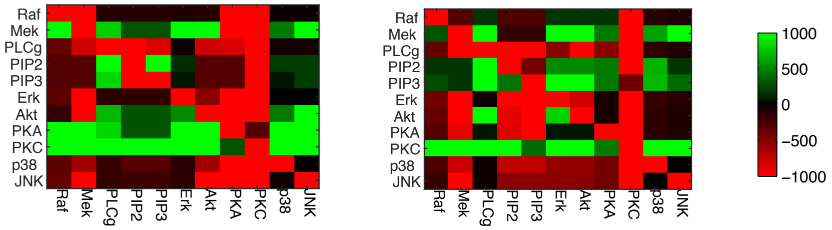


Figure 8: ACI results (input: independences of order ≤ 1 and weighted ancestral relations) on no-ICAM (left) and ICAM (right) batches. These heatmaps are identical to the ones in Figures 6 and 7, but are reproduced here next to each other for easy comparison.

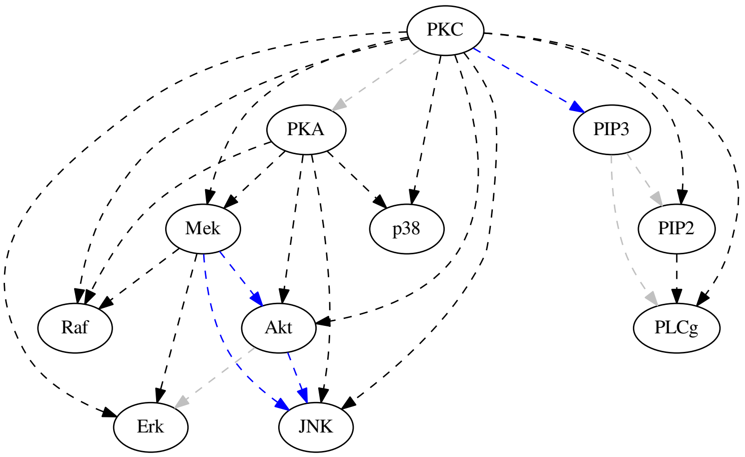


Figure 9: Comparison of ancestral relations predicted by ACI and the score-based method from [17], both using the no-ICAM batch. Depicted are the top 21 ancestral relations obtained by ACI and the transitive closure of the top 17 direct causal relations reported in [17], which results in 21 ancestral relations. Black edges are ancestral relations found by both methods, blue edges were identified only by ACI, while grey edges are present only in the transitive closure of the result from [17].

LoCI [2]) would decide that MEK and ERK are nonadjacent, unable to recover from that erroneous decision. This illustrates one of the advantages of our approach.

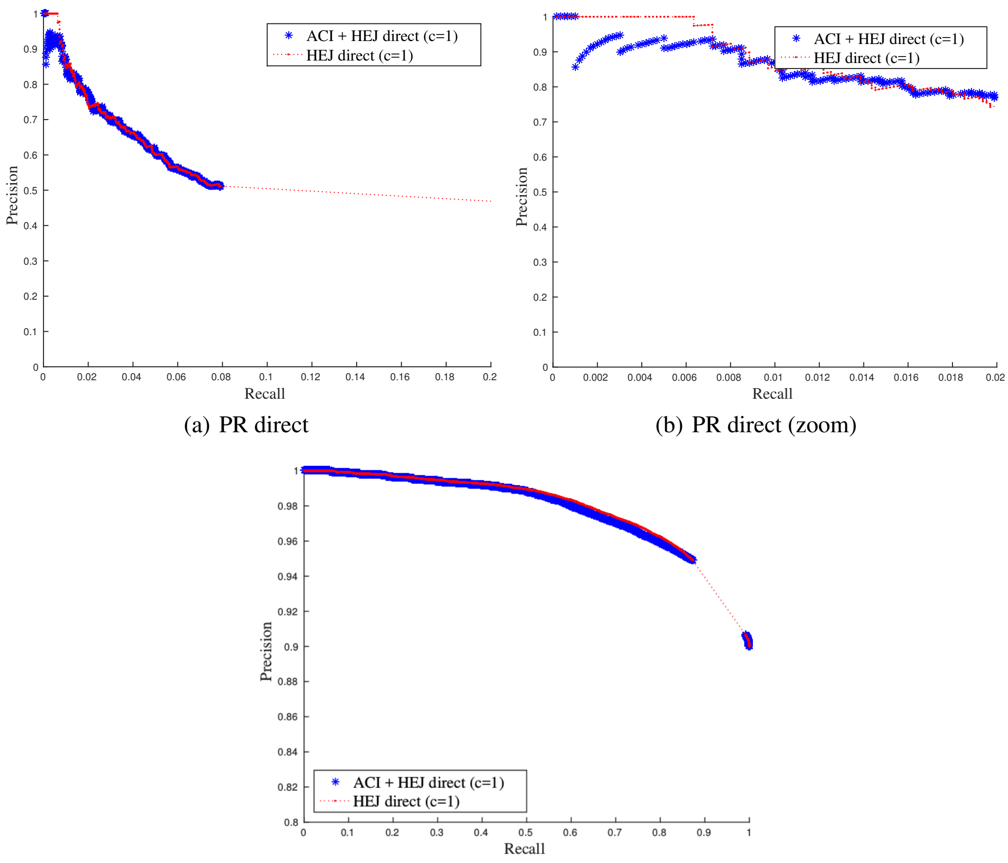
9.4 Comparison with other approaches

We also compare our results with other, mostly score-based approaches. Amongst other results, [17] report the top 17 direct causal relations on the no-ICAM batch that were inferred by their score-based method when assuming acyclicity. In order to compare fairly with the ancestral relations found by ACI, we first perform a transitive closure of these direct causal relations, which results in 21 ancestral relations. We then take the top 21 predicted ancestral relations from ACI (for the same no-ICAM batch), and compare the two in Figure 9. The black edges, the majority, represent the ancestral relations found by both methods. The blue edges are found only by ACI, while the grey edges are found only by [17]. Interestingly, the results are quite similar, despite the very different approaches. In particular, ACI allows for confounders and is constraint-based, while the method in [17] assumes causal sufficiency (i.e., no confounders) and is score-based.

Table 2 summarizes most of the existing work on this flow cytometry dataset. It was originally part of the S1 material of [16]. We have updated it here by adding also the results for ACI and the transitive closure of [17].

Table 2: Updated Table S1 from [16]: causal relationships between the biochemical agents in the flow cytometry data of [23], according to different causal discovery methods. The consensus network according to [23] is denoted here by "[23]a" and their reconstructed network by "[23]b". For [17] we provide two versions: "[17]a" for the top 17 edges in the acyclic case, as reported in the original paper, and "[17]b" for its transitive closure, which consists of 21 edges. To provide a fair comparison, we also pick the top 21 ancestral predictions from ACI.

	Direct causal predictions					Ancestral predictions		
Edge	[23]a	[23]b	[17]a	[5]	ICP [19]	hiddenICP [19]	[17]b	ACI (top 21)
RAF→MEK	✓	✓				✓		
MEK→RAF			✓	✓		✓	✓	✓
MEK→ERK	✓	✓	✓				✓	✓
MEK→AKT								✓
MEK→JNK								✓
PLCg→PIP2	✓	✓		✓	✓	✓		
PLCg→PIP3		✓		✓				
PLCg→PKC	✓			✓				
PIP2→PLCg			✓		✓		✓	✓
PIP2→PIP3				✓				
PIP2→PKC	✓							
PIP3→PLCg	✓						✓	
PIP3→PIP2	✓	✓	✓		✓	✓	✓	
PIP3→AKT	✓							
AKT→ERK			✓		✓	✓	✓	
AKT→JNK								✓
ERK→AKT		✓		✓	✓	✓		
ERK→PKA				✓				
PKA→RAF	✓	✓					✓	✓
PKA→MEK	✓	✓	✓	✓		✓	✓	✓
PKA→ERK	✓	✓			✓		✓	✓
PKA→AKT	✓	✓	✓	✓		✓	✓	✓
PKA→PKC				✓				
PKA→P38	✓	✓	✓				✓	✓
PKA→JNK	✓	✓	✓	✓			✓	✓
PKC→RAF	✓	✓	✓				✓	✓
PKC→MEK	✓	✓	✓	✓			✓	✓
PKC→PLCg			✓				✓	✓
PKC→PIP2			✓				✓	✓
PKC→PIP3								✓
PKC→ERK							✓	✓
PKC→AKT			✓				✓	✓
PKC→PKA		✓	✓				✓	
PKC→P38	✓	✓	✓	✓		✓	✓	✓
PKC→JNK	✓	✓	✓	✓	✓	✓	✓	✓
P38→JNK						✓		
P38→PKC						✓		
JNK→PKC						✓		
JNK→P38				✓		✓		



(a) PR direct

(b) PR direct (zoom)

(c) PR direct acausal

Figure 10: Synthetic data: accuracy for the two prediction tasks (direct causal and noncausal relations) for $n = 6$ variables using the frequentist test with $\alpha = 0.05$ for 2000 simulated models.

Table 3: Average execution times for recovering causal relations with different strategies for 2000 models for $n = 6$ variables using the frequentist test with $\alpha = 0.05$.

Average execution time (s)					
Setting		**Direct causal relations**		**Only second step**	**Ancestral relations**
n	c	ACI with restricted HEJ	direct HEJ	restricted HEJ	ancestral HEJ
6	1	9.77	15.03	7.62	12.09
6	4	16.96	314.29	14.43	432.67
7	1	36.13	356.49	30.68	715.74
8	1	98.92	$\geq$ 2500	81.73	$\geq$ 2500
9	1	361.91	$\geq$ 2500	240.47	$\geq$ 2500

10 Mapping ancestral structures to direct causal relations

An ancestral structure can be seen as the transitive closure of the directed edges of an acyclic directed mixed graph (ADMG). There are several strategies to reconstruct "direct" causal relations from an ancestral structure, in particular in combination with our scoring method. Here we sketch a possible strategy, but we leave a more in-depth investigation to future work.

A possible strategy is to first recover the ancestral structure from ACI with our scoring method and then use it as "oracle" input constraints for the HEJ [9] algorithm. Specifically, for each weighted output $(X \dashrightarrow Y, w)$ obtained by ACI, we add $(X \dashrightarrow Y, \infty)$ to the input list I, and similarly for each $X \not\dashrightarrow Y$. Then we can use our scoring algorithm with HEJ to score direct causal relations (e.g.,

$f = X \to Y$) and direct acausal relations (e.g., $f = X \not\to Y$):

$$C(f) = \min_{W \in \mathcal{W}} \mathcal{L}(W; I \cup \{(\neg f, \infty)\}) - \min_{W \in \mathcal{W}} \mathcal{L}(W; I \cup \{(f, \infty)\}). \qquad (19)$$

In the standard HEJ algorithm, $\mathcal{W}$ are all possible ADMGs, but with our additional constraints we can reduce the search space to only the ones that fit the specific ancestral structure, which is on average and asymptotically a reduction of $2^{n^2/4+o(n^2)}$ for n variables. We will refer to this two-step approach as *ACI with restricted HEJ (ACI + HEJ)*. A side effect of assigning infinite scores to the original ancestral predictions instead of the originally estimated scores is that some of the estimated direct causal predictions scores will also be infinite, flattening their ranking. For this preliminary evaluation, we fix this issue by reusing the original ancestral scores also for the infinite direct predictions scores. Another option may be to use the ACI scores for (a)causal relations as soft constraints for HEJ, although at the time of writing it is still unclear whether this would lead to the same speedup as the previously mentioned version.

We compared accuracy and execution times of standard HEJ (without the additional constraints derived from ACI) with ACI with restricted HEJ on simulated data. Figure 10 shows PR curves for predicting the presence and absence of direct causal relations for both methods. In Table 3 we list the execution times for recovering direct causal relations. Additionally, we list the execution times of only the second step of our approach, the *restricted HEJ*, to highlight the improvement in execution time resulting from the restrictions. In this preliminary investigation with simulated data, ACI with restricted HEJ is much faster than standard HEJ (without the additional constraints derived from ACI) for predicting direct causal relations, but only sacrifices a little accuracy (as can be seen in Figure 10). In the last column of Table 3, we show the execution times of standard HEJ when used to score ancestral relations. Interestingly, predicting direct causal relations is faster than predicting ancestral relations with HEJ. Still, for 8 variables the algorithm takes more than 2,500 seconds for all but 6 models of the first 40 simulated models.

Another possible strategy first reconstructs the (possibly incomplete) PAG [25] from ancestral relations and conditional (in)dependences using a procedure similar to LoCI [2], and then recovering direct causal relations. There are some subtleties in the conversion from (possibly incomplete) PAGs to direct causal relations, so we leave this and other PAG based strategies, as well as a better analysis of conversion of ancestral relations to direct causal relations as future work.

11 Complete ACI encoding in ASP

Answer Set Programming (ASP) is a widely used declarative programming language based on the stable model semantics of logical programming. A thorough introduction to ASP can be found in [13, 8]. The ASP syntax resembles Prolog, but the computational model is based on the principles that have led to faster solvers for propositional logic [13]. ASP has been applied to several NP-hard problems, including learning Bayesian networks and ADMGs [9]. Search problems are reduced to computing the stable models (also called answer sets), which can be optionally scored.

For ACI we use the state-of-the-art ASP solver `clingo 4` [7]. We provide the complete ACI encoding in ASP using the clingo syntax in Table 4. We encode sets via their natural correspondence with binary numbers and use boolean formulas in ASP to encode set-theoretic operations. Since ASP does not support real numbers, we scale all weights by a factor of 1000 and round to the nearest integer.

12 Open source code repository

We provide an open-source version of our algorithms and the evaluation framework, which can be easily extended, at `http://github.com/caus-am/aci`.

Table 4: Complete ACI encoding in Answer Set Programming, written in the syntax for clingo 4.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%  Ancestral Causal Inference (ACI)    %%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%% Preliminaries:
%%% Define ancestral structures:
{ causes(X,Y) } :- node(X), node(Y), X!=Y.
:- causes(X,Y), causes(Y,X), node(X), node(Y), X < Y.
:- not causes(X,Z), causes(X,Y), causes(Y,Z), node(X), node(Y), node(Z).

%%% Define the extension of causes to sets.
% existsCauses(Z,W) means there exists I \in W that is caused by Z.
1{causes(Z, I): ismember(W,I)} :- existsCauses(Z,W), node(Z), set(W), not ismember(W,Z).
existsCauses(Z,W) :- causes(Z, I), ismember(W,I), node(I), node(Z), set(W), not ismember(W,Z), Z!=I.

%%% Generate in/dependences in each model based on the input in/dependences.
1{ dep(X,Y,Z);indep(X,Y,Z) }1 :- input_indep(X,Y,Z,_).
1{ dep(X,Y,Z);indep(X,Y,Z) }1 :- input_dep(X,Y,Z,_).

%%% To simplify the rules, add symmetry of in/dependences.
dep(X,Y,Z) :- dep(Y,X,Z), node(X), node(Y), set(Z), X!=Y, not ismember(Z,X), not ismember(Z,Y).
indep(X,Y,Z) :- indep(Y,X,Z), node(X), node(Y), set(Z), X!=Y, not ismember(Z,X), not ismember(Z,Y).

%%%%% Rules from LoCI:
%%% Minimal independence rule (4) : X || Y | W u [Z] => Z -/-> X, Z -/-> Y, Z -/-> W
:- not causes(Z,X), not causes(Z,Y), not existsCauses(Z,W), dep(X,Y,W), indep(X,Y,U),
U==W+2**(Z-1), set(W), node(Z), not ismember(W, Z), Y != Z, X != Z.

%%% Minimal dependence rule (5): X |/| Y | W u [Z] => Z --> X or Z-->Y or Z-->W
:- causes(Z,X), indep(X,Y,W), dep(X,Y,U), U==W+2**(Z-1), set(W), set(U), node(X),
 node(Y), node(Z), not ismember(W, Z), not ismember(W, X), not ismember(W,Y),
 X != Y, Y != Z, X != Z.
% Note: the version with causes(Z,Y) is implied by the symmetry of in/dependences.
:- existsCauses(Z,W), indep(X,Y,W), dep(X,Y,U), U==W+2**(Z-1), set(W), set(U), node(X),
 node(Y), node(Z), not ismember(W, Z), not ismember(W, X), not ismember(W,Y),
 X != Y, Y != Z, X != Z.

%%%%% ACI rules:
%%% Rule 1: X || Y | U and X -/-> U => X -/->Y
:- causes(X,Y), indep(X,Y,U), not existsCauses(X,U), node(X), node(Y), set(U), X != Y,
not ismember(U,X), not ismember(U,Y).

%%% Rule 2: X || Y | W u [Z] => X |/| Z | W
dep(X,Z,W) :- indep(X,Y,W), dep(X,Y,U), U==W+2**(Z-1), set(W), set(U),
node(X), node(Y), node(Z), X != Y, Y != Z, X != Z, not ismember(W,X), not ismember(W,Y).

%%% Rule 3: X |/| Y | W u [Z] => X |/| Z | W
dep(X,Z,W) :- dep(X,Y,W), indep(X,Y,U), U==W+2**(Z-1), set(W), set(U),
node(X), node(Y), node(Z), X != Y, Y != Z, X != Z, not ismember(W,X), not ismember(W,Y).

%%% Rule 4: X || Y | W u [Z] and X || Z | W u U => X || Y | W u U
indep(X,Y,A) :- dep(X,Y,W), indep(X,Y,U), U==W+2**(Z-1), indep(X,Z,A), A==W+2**(B-1),
 set(W), set(U), not ismember(W,X), not ismember(W,Y), node(X), node(Y), node(Z),
 set(A), node(B), X!=B, Y!=B, Z!=B, X != Y, Y != Z, X != Z.

%%% Rule 5: Z |/| X | W and Z |/| Y | W and X || Y | W => X |/| Z | W u Z
dep(X,Y,U) :- dep(Z,X,W), dep(Z,Y,W), indep(X,Y,W), node(X), node(Y), U==W+2**(Z-1),
 set(W), set(U), X != Y, Y != Z, X != Z, not ismember(W,X), not ismember(W,Y).

%%%%% Loss function and optimization.
%%% Define the loss function as the incongruence between the input in/dependences
%%% and the in/dependences of the model.
fail(X,Y,Z,W) :- dep(X,Y,Z), input_indep(X,Y,Z,W).
fail(X,Y,Z,W) :- indep(X,Y,Z), input_dep(X,Y,Z,W).

%%% Include the weighted ancestral relations in the loss function.
fail(X,Y,-1,W) :- causes(X,Y), wnotcauses(X,Y,W), node(X), node(Y), X != Y.
fail(X,Y,-1,W) :- not causes(X,Y), wcauses(X,Y,W), node(X), node(Y), X != Y.

%%% Optimization part: minimize the sum of W of all fail predicates that are true.
#minimize{W,X,Y,C:fail(X,Y,C,W) }.
```

References

[1] G. Borboudakis and I. Tsamardinos. Incorporating causal prior knowledge as path-constraints in Bayesian networks and Maximal Ancestral Graphs. In *ICML*, pages 1799–1806, 2012.

[2] T. Claassen and T. Heskes. A logical characterization of constraint-based causal discovery. In *UAI*, pages 135–144, 2011.

[3] T. Claassen and T. Heskes. A Bayesian approach to constraint-based causal inference. In *UAI*, pages 207–216, 2012.

[4] D. Colombo, M. H. Maathuis, M. Kalisch, and T. S. Richardson. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, 40(1):294–321, 2012.

[5] D. Eaton and K. Murphy. Exact Bayesian structure learning from uncertain interventions. In *AISTATS*, pages 107–114, 2007.

[6] D. Entner, P. O. Hoyer, and P. Spirtes. Data-driven covariate selection for nonparametric estimation of causal effects. In *AISTATS*, pages 256–264, 2013.

[7] M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub. *Clingo* = ASP + control: Extended report. Technical report, University of Potsdam, 2014. `http://www.cs.uni-potsdam.de/wv/pdfformat/gekakasc14a.pdf`.

[8] M. Gelfond. Answer sets. In *Handbook of Knowledge Representation*, pages 285–316. 2008.

[9] A. Hyttinen, F. Eberhardt, and M. Järvisalo. Constraint-based causal discovery: Conflict resolution with Answer Set Programming. In *UAI*, pages 340–349, 2014.

[10] M. Kalisch and P. Bühlmann. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research*, 8:613–636, 2007.

[11] M. Kalisch, M. Mächler, D. Colombo, M. Maathuis, and P. Bühlmann. Causal inference using graphical models with the R package pcalg. *Journal of Statistical Software*, 47(1):1–26, 2012.

[12] D. J. Kleitman and B. L. Rothschild. Asymptotic enumeration of partial orders on a finite set. *Transactions of the American Mathematical Society*, 205:205–220, 1975.

[13] V. Lifschitz. What is Answer Set Programming? In *AAAI*, pages 1594–1597, 2008.

[14] D. Margaritis and F. Bromberg. Efficient Markov network discovery using particle filters. *Computational Intelligence*, 25(4):367–394, 2009.

[15] F. Markowetz, S. Grossmann, and R. Spang. Probabilistic soft interventions in conditional Gaussian networks. In *AISTATS*, pages 214–221, 2005.

[16] N. Meinshausen, A. Hauser, J. M. Mooij, J. Peters, P. Versteeg, and P. Bühlmann. Methods for causal inference from gene perturbation experiments and validation. *Proceedings of the National Academy of Sciences*, 113(27):7361–7368, 2016.

[17] J. M. Mooij and T. Heskes. Cyclic causal discovery from continuous equilibrium data. In *UAI*, pages 431–439, 2013.

[18] J. Pearl. *Causality: models, reasoning and inference*. Cambridge University Press, 2009.

[19] J. Peters, P. Bühlmann, and N. Meinshausen. Causal inference using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society, Series B*, 8(5):947–1012, 2015.

[20] J. Ramsey, J. Zhang, and P. Spirtes. Adjacency-faithfulness and conservative causal inference. In *UAI*, pages 401–408, 2006.

[21] D. Rothenhäusler, C. Heinze, J. Peters, and N. Meinshausen. BACKSHIFT: Learning causal cyclic graphs from unknown shift interventions. In *NIPS*, pages 1513–1521, 2015.

[22] A. Roumpelaki, G. Borboudakis, S. Triantafillou, and I. Tsamardinos. Marginal causal consistency in constraint-based causal learning. In *Causation: Foundation to Application Workshop, UAI*, 2016.

[23] K. Sachs, O. Perez, D. Pe'er, D. Lauffenburger, and G. Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308:523–529, 2005.

[24] P. Spirtes. An anytime algorithm for causal inference. In *AISTATS*, pages 121–128, 2001.

[25] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT press, 2000.

[26] J. Tian and J. Pearl. Causal discovery from changes. In *UAI*, pages 512–521, 2001.

[27] S. Triantafillou and I. Tsamardinos. Constraint-based causal discovery from multiple interventions over overlapping variable sets. *Journal of Machine Learning Research*, 16:2147–2205, 2015.

[28] J. Zhang. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artifical Intelligence*, 172(16-17):1873–1896, 2008.