Leveraging Directed Causal Discovery to Detect Latent Common Causes in Cause-Effect Pairs

Ciarán M. Gilligan-Lee, Christopher Hart, Jonathan Richens, and Saurabh Johri

Abstract—The discovery of causal relationships is a fundamental problem in science and medicine. In recent years, many elegant approaches to discovering causal relationships between two variables from observational data have been proposed. However, most of these deal only with purely directed causal relationships and cannot detect latent common causes. Here, we devise a general heuristic which takes a causal discovery algorithm that can only distinguish purely directed causal relations and modifies it to also detect latent common causes. We apply our method to two directed causal discovery algorithms, the geometric causal inference (IGCI) of (Daniusis et al., 2010) and the kernel conditional deviance for causal inference of (Mitrovic et al., 2018), and extensively test on synthetic data—detecting latent common causes in additive, multiplicative and complex noise regimes—and on real data, where we are able to detect known common causes. In addition to detecting latent common causes, our experiments demonstrate that both the modified algorithms preserve the performance of the original in distinguishing directed causal relations.

Index Terms—Causal discovery, causal inference, counterfactual inference, hidden common cause, latent confounding.

I. INTRODUCTION

Causal knowledge is crucial in medicine, as causal relations—unlike correlations—allow one to reason about the consequences of possible treatments [1], [2], [3]. Indeed, determining whether a particular treatment causes a reduction in the severity of a disease is an important problem in healthcare. However, learning from medical records that a treatment is correlated with recovery is not sufficient to conclude that treatment will cause a new patient to recover, due to the presence of latent confounding. More expensive treatment could be given to wealthier patient’s who are more likely to recover regardless of treatment, due to better lifestyle and diet. Here, the patients wealth acts as a latent confounder between receiving treatment and recovering. In reality, the treatment could be detrimental. Thus, learning the causal structure is vital. This problem is made harder when no controlled trial data exists—meaning the standard approach to learning causal structure is not available. Hence, methods for discovering causal structure from uncontrolled, or observational, data are crucial to making correct treatment decisions.

II. RELATED WORK

In this work, causal structure will be graphically represented as a directed acyclic graph (DAG). Methods for discovering the causal structure underlying a data generating process largely

Fig. 1. Blue nodes represent observed variables, green nodes latent noise. (a) A causes B. (b) B causes A. (c) Latent common cause between A and B.

Recently, a number of approaches to discovering causal relations between two variables have been proposed, including [4], [5], [6], [7] among many others, which we describe in Section II. However, apart from [8] and [9], these deal only with purely directed causal relationships and cannot detect latent common causes. That is, given two variables A and B, these algorithms can only distinguish A causes B, graphically depicted in Fig. 1(a), from B causes A, depicted in Fig. 1(b). Those algorithms that can detect latent common causes either impose stern restrictions on the underlying causal model—such as enforcing linearity [5], [10], demanding that noise be additive [8], or requiring huge amounts of training data [9]. As the presence of latent common causes is a significant problem, surmounting these shortcomings and developing methods for detecting latent common causes that do not rely on strong assumptions, such as additive noise or linearity, is an important problem in causal discovery.

Here, we devise a heuristic method which takes a purely directed causal discovery algorithm and modifies it so that it also discovers latent common causes. That is, our method takes as input a causal discovery algorithm which can only distinguish the causal structures in Fig. 1(a) and (b), and outputs a causal discovery algorithm which can distinguish between all three structures in Fig. 1. We apply our method to two directed causal discovery algorithms, information geometric causal inference (IGCI) algorithm of [11], and kernel conditional deviance for causal inference algorithm of [7], which were both shown in [7] to perform very well on the Tübingen cause-effect pair dataset1 [12], with IGCI scoring 74% and kernel conditional deviance causal discovery (KCDC) scoring 73%. We extensively test on synthetic data—detecting latent common causes in additive, multiplicative and complex noise regimes—as well as on real data, where we are able to detect known common causes. Importantly, and in addition to detecting latent common causes with high accuracy, our experiments show that both modified algorithms do not sacrifice the performance of the original in distinguishing directed causal relations.

1https://webdatav.tuebingen.mpg.de/cause-effect/
fall into two categories. The first, which we term global causal discovery, attempts to reconstruct a (partially) undirected version of the causal structure. This approach is broadly split into two categories: constraint-based and score-based. The constraint-based approach employs conditional independence tests between the variables in question to determine which should share an edge in the causal structure. Examples include the probabilistic causation (PC) algorithm and the fast causal inference (FCI) algorithm [13], the inductive causality (IC) algorithm [1], as well as algorithms which allow for latent variables [14] and selection bias [15]. The score-based approach introduces a scoring function, such as minimum description length, that evaluates each network with respect to some training data, and searches for the best network according to this function [16]. Greedy equivalent search (GES) [17] is a prime example of the score-based causal discovery. Hybrid approaches employing both constraint- and score-based techniques outperform either method alone [18].

The main drawback of constraint-based algorithms is that, as they can only recover Markov equivalence classes, they are not always able to orient edges between dependent variables. Given correlated variables \( A \) and \( B \), these methods are unable to distinguish the structures depicted in Fig. 1. The second category of causal discovery algorithm, which we term local or bivariate causal discovery, aims to address this by exploiting notions of asymmetry between cause and effect. These methods specify different assumptions that aim to make such asymmetries testable at an observational level.

The first type of assumption proposed in the literature, known as the functional causal model approach, specifies that each effect is a deterministic function of its cause together with some latent, independent noise term [19]. The first such algorithm, termed Linear Non-Gaussian Additive noise Model (LiNGAM) [5], assumes the functions are linear and the latent noise variables are non-Gaussian. Given these assumptions, this method can distinguish between the causal structures in Fig. 1(a) and (b); a separate hidden variables extension [20] extends this result to all structures in Fig. 1. The other prototypical example, the additive noise model (ANM) [4], allows the effect to be an arbitrary function of the cause, but assumes the effect only depends on the latent noise term additively. The ANM can distinguish between the two structures in Fig. 1(a) and (b) as long as the additivity assumption holds. ANM has been extended to allow the effect to depend on the cause and noise in a nonlinear fashion [21]. Finally, the confounding additive noise (CAN) algorithm [8] extends the ANM algorithm to deal with latent common causes, allowing all structures in Fig. 1 to be distinguished. We review the CAN model in Section II-A.

The second type of assumption stipulates that the cause \( P(cause) \) is independent of the mechanism \( P(cause|effect) \). An information geometric approach to measuring such independence has been proposed [6], [11], known as the IGCI algorithm. This method can only distinguish the two structures in Fig. 1(a) and (b). IGCI does not require specific parametric functional relationships to hold.

The last type of assumption entails that the causal mechanism \( P(effect|cause) \) should be “simpler,” given some quantifiable notion of “simple,” than the anti-causal mechanism \( P(cause|effect) \). The KCDC algorithm [7] algorithm uses conditional kernel mean embeddings to establish a norm on conditional probabilities and uses the variance of these as a simplicity measure. As experimentally shown in [7], IGCI and KCDC algorithms constitute the current state-of-the-art at distinguishing between the two causal structures in Fig. 1(a) and (b).

While the ANM algorithm has been extended to deal with latent common causes via the CAN algorithm, its generalizations, such as KCDC and IGCI, have not. The current article introduces a heuristic method that takes as input any purely directed causal discovery algorithm and outputs a new algorithm that can distinguish all causal structures in Fig. 1.

While many purely directed causal discovery algorithms have not been extended to deal with the presence of latent confounders, there are works that—given certain assumptions—can determine with a certain degree of confidence whether the correlation observed between two variables can be explained by a latent common cause or a directed causal relation. It should be noted however that these algorithms are agnostic to the orientation of the directed causal relation if a latent common cause explanation has been ruled out. One such algorithm [10] assumes a linear relationship between the variables in question, including any latent common causes and outputs the causal structure with the smallest Minimum Description Length—an approximation to the Kolmogorov complexity—as the best causal explanation of the data. The authors of this work report good performance of their method. Another work, [22] uses a generalization of the Rényi entropy to determine if there exists a latent common cause that can explain the observed correlations, or whether they must be due to a direct causal link. Motivated by use cases in genetics, Janzing et al. [23] provides a method that can determine if a low-complexity, by which they mean, low-cardinality—e.g., a binary variable—latent common cause can explain the correlations in the data. Their method is based on the analysis of the location of the conditional distributions \( P(Y|x) \) in the simplex of all distributions of \( Y \). Lastly, [24] use spectral analysis and concentration of measure results to determine whether a direct cause explanation of the data is viable, or whether a scalar-valued latent common cause is required to explain the data. As in [10] above, Janzing and Schölkopf [24] assume linear models. Subsequent work in [25] extends the method of [24] to deal with high-dimensional latent common causes.

The main contribution of this article is a heuristic method that takes as input any purely directed causal discovery algorithm and outputs a new algorithm that can distinguish all causal structures in Fig. 1. Unlike the methods described above, our approach not only distinguishes between a purely directed and purely common cause, causal structures, it can also orient the directed causal link if a common cause has been ruled out. Moreover, the main innovation of our method is that if any new purely directed causal discovery algorithm is devised, then this can be immediately transformed into one that can also distinguish latent common causes by applying our algorithm.

### A. Identifying Latent Common Causes Using ANMs

As a prelude to our method for detecting latent common causes, we review a current state-of-the-art algorithm for distinguishing all structures in Fig. 1, the confounding additive noise (CAN) model of [8]. In CAN, one assumes that the functions underlying the causal structure from Fig. 1(c) are...
A = f(C) + uA and B = g(C) + uB, for arbitrary f, g. The first step of CAN is to fit such a model using samples from A, B. To do this, manifold learning is employed to learn a representation of C from the observed samples of A, B. Next, functions are fit to map the learned representation of C to A and B using e.g., Gaussian Process regression. The residuals resulting from this regression correspond to representations of the noise terms, uA and uB. In fitting this model, one must ensure that C, uA, uB are all mutually independent, as implied by the graphical structure of Fig. 1(c).

To ensure independence, CAN employs a computationally expensive optimization procedure. See Section IV for further discussion.

With such a model fit using samples of A, B, the next step is to use it to distinguish all structures in Fig. 1. Note that the additive noise assumption for Fig. 1(a) means B = g(A) + uB and A = uA, for arbitrary g. CAN distinguishes Fig. 1(a)–(c) if the variance of the learned uA is small, relative to the variance of uB, and prefers Fig. 1(c) if the variances are approximately equal. The case of Fig. 1(b) follows by switching A and B. The authors justify this as follows. Suppose A causes B, and, by a slight measurement error, we observe A', which differs from A by a small additive term. Here, we would expect the functions to be as described above for Fig. 1(c). But we should not distinguish A from its measurement result A' if both variables almost coincide. In this case, after normalizing A, B to unit variance, the variance of uA would be small compared to uB, as A is close to A'. In general, the CAN algorithm distinguishes between all structures in Fig. 1 using the following decision criteria: 1) A → B if var(uA)/var(uB) ≫ 1; 2) A → B if var(uA)/var(uB) ≪ 1; and 3) A ← C → B if var(uA)/var(uB) ≈ 1. Determining thresholds for the above criteria is a tricky problem, with values being decided on a case-by-case basis; see [8] for more information.

III. METHODS

We work in the functional causal model framework [19]. Here, a causal structure corresponds to a DAG between observed and latent variables, with each variable a deterministic function of its parents and some latent, independent noise term. For instance, in Fig. 1(a), this means A = f(uA) and B = g(A, uB) for deterministic functions f, g, while in Fig. 1(c) it means A = f(C, uA) and B = g(C, uB).

A causal model corresponds to a DAG, the functions, and a specification of the priors over the latent noise terms. Hence one can write the observed probability of A = a given B = b as P(A = a | B = b) = ∑_{u}P(A = a | f(U, u))Q(U = u) where f is the deterministic function mapping B and the latent noise term U to A, and Q(U) is the prior over the latent noise term. We will say that two causal models are observationally equivalent if they are characterized by the same set of distributions over observed variables. Denote the set of distributions over observed variables, which is specified by the functions and latent priors as discussed above, for a particular causal model M, by P_M. Two causal models M and M' are then observationally equivalent if and only if P_M = P'_M.

A. Description and Intuition of Our Heuristic Algorithm

In this section we will describe our heuristic and provide an outline of the intuition it is derived from. First, we will provide a quick description of our method, the mechanics of which will then be explained in the rest of the section.

Description of heuristic method 1: We start by fitting a curve to samples of observations A_i, B_i (using a manifold learning algorithm such as isomap [26], say) which yields a 1-D parameterization T_i of the data. Proceeding, the heuristic is motivated by the following intuition.

1) If A ← C → B then T is expected to essentially correspond to C and directed causal discovery algorithms applied to (A, T) and (B, T) are expected to conclude that arrows are outgoing from T.

2) If A → B (A ← B follows similarly), then—as we'll justify in this section—T is expected to essentially correspond to f(uA), where A = f(uA), and directed causal discovery algorithms applied to (A, T) yield inconclusive and ambiguous results, while for (B, T) they output a causal arrow outgoing from T (as B = g(A, uB) = g(f(T), uB)) as we'll discuss in this section.

This asymmetry can then be used to distinguish A → B, A ← B and A ← C → B.

We will now proceed with explaining why each of the above steps follows. To this end, and to help with exposition, we have broken down the required conceptual components and present them as Intuition Aids.

Intuition Aid 1: Any causal model with a directed arrow between A & B (A → B or B → A) is observationally equivalent5 to one that is purely common cause.

Proof: This result has been shown before, in [27] for example, but to see that it holds for the case of Fig. 1(a), consider the following. A model for this causal structure corresponds to specifying the functional dependencies between A, B and their parents: A = f(uA) and B = g(A, uB). Substituting f(uA) for A in B, one obtains B = g(f(uA), uB). Defining u'_A := f(uA) results in the relations A = u'_A and B = g(u'_A, uB). The directed arrow from A to B has been replaced by the latent “common cause” u'_A, and Fig. 1(a) has been reduced to a purely common cause model. This is outlined graphically in the first two causal structures from Fig. 2, where ≡ denotes observational equivalence.

We say a causal model is in canonical form6 when it is purely common cause. Note that Fig. 1(c) is already in canonical form. The following Intuition Aid provides another key observation underlying our approach.

Intuition Aid 2: Given the causal model in Fig. 1(a),5 the canonical model is consistent with two causal structures: one

Fig. 2. Depiction of three observationally equivalent DAGs discussed in Intuition Aid 2. The symbol ≡ denotes observational equivalence.

5See start of Section III for the definition of observational equivalence.

4Note that this notion of canonical model is different to the one discussed in [20]. There a model is canonical if all functions are linear and each latent variable is a root node with at least two children. Furthermore, although different latent variables may have the same sets of children, no two latent variables should exhibit exactly proportional sets of connection strengths to the observed variables.

6The case of Fig. 1(b) follows by interchanging A and B.
where \( u_A \) is a common cause between \( A \) and \( B \), and the other \( u_B \) is the mediator of the causal arrow from \( A \) to \( B \). This is depicted in the last two structures of Fig. 2.

Proof: Indeed, without loss of generality, the canonical model of Fig. 1(a) is given by \( A = u_A \) and \( B = g(u_A, u_B) \). These functions are consistent with the second causal structure in Fig. 2, in which a causal arrow points from \( u_A \) to \( B \). But the equality \( A = u_A \) is also consistent with the third causal structure in Fig. 2, where the causal arrow points from \( B \) to \( A \). That is, the relationship \( A = u_A \) is consistent with both an arrow from \( u_A \) to \( A \) and an arrow from \( A \) to \( u_A \). Note that due to the independence of the noise terms \( u_A \) and \( u_B \), this is not true of the other arrows in the causal structure. A causal arrow pointing from \( B \) to \( u_A \) in Fig. 2 would induce correlations between \( u_A \) and \( u_B \), in direct contradiction with the independence of noise terms. \( \square \)

Intuition Aid 2 tells us that in the canonical model of Fig. 1(a), the directionality of the causal link between \( A \) and \( u_A \) is underdetermined, as illustrated in the last two structures of Fig. 2. Thus applying a directed causal discovery algorithm to \( A \) and \( u_A \) will result in inconclusive and ambiguous results. All other causal links are fully determined and directed however. If, on the other hand, the original structure had been Fig. 1(b), then the causal link between the canonical “common cause”, \( u_B \), and \( B \) would be underdetermined, with all remaining links determined and directed. Note that if the structure had originally been that of Fig. 1(c), then neither arrow from the true common cause, \( C \), to the observed variables is underdetermined.

This observation is what underlies the conclusions from items (a) and (b) outlined at the start of this section. In Sections III-B and III-C we will describe how to implement our method.

B. From Intuition to Implementation

The presentation of our heuristic method has thus far has not formally defined the notion of an underdetermined causal relation. We now present this definition. Here we focus on directed causal discovery algorithms, \( D \), that decide the causal direction using a quantifiable notion of asymmetry, such as IGCI [11] and KCDC [7], described in Section II. In general, such algorithms assign a real scalar to each causal direction \( v^D_{A \rightarrow B} \) and \( v^D_{B \rightarrow A} \) returning the causal direction with smallest \( v \) value. Due to the pervasiveness of statistical noise, such algorithms have a decision threshold \( \delta \), such that if \( |v^D_{A \rightarrow B} - v^D_{B \rightarrow A}| < \delta \) the algorithm fails to detect a causal direction.

As an illustration, the KCDC algorithm of [7] uses conditional kernel mean embeddings to establish a norm on conditional probabilities \( P(B|A = a) \) and \( P(A|B = b) \) for each value of \( A = a \), \( B = b \). If the variance \( v_{A \rightarrow B} \) of the norms \( \left( P(B|A = a)\right)_a \) is smaller than the variance \( v_{B \rightarrow A} \) of \( \left( P(A|B = b)\right)_b \), the KCDC algorithm returns Fig. 1(a), otherwise it returns Fig. 1(b).

The following definition formalizes the notion of a causal relation being underdetermined relative to a particular directed causal discovery algorithm.

Definition 1: A causal relation between two variables \( A \) and \( B \) is underdetermined relative to a causal discovery algorithm \( D \) if \( |v^D_{A \rightarrow B} - v^D_{B \rightarrow A}| < \delta \) for prespecified decision threshold \( \delta \).

Finally, to implement our heuristic method, the canonical common cause needs to be learned from samples of the observed variables \( A, B \). At the start of Section III-A we stated that manifold learning or dimensionality reduction algorithms, such as the isomap algorithm of [26], would perform this requirement. Indeed, under the hypothesis that such an algorithm will learn the most likely latent representation of the data, this seems intuitively to be the case. However, for this intuition to be borne out, we need to stipulate that the influence of the latent noise terms on the observed variables be small. That is, we need to require that fluctuations around the manifold parameterized by the canonical common cause due to latent noises be small for the canonical common cause from samples of the observed variables \( A, B \).

Intuition Aid 3: Under the assumption that the influence of the latent noise term is small, the canonical common cause can be identified from samples of \( A, B \).

Proof: Consider the causal model of Fig. 1(a),\(^6\) which, following Lemma 1 we can write as \( A = u_A \) and \( B = g(u_A, u_B) \). Now, a Taylor expansion yields \( B = g(u_A, 0) + g'(u_A, 0)u_B + \) (higher order terms in \( u_B \)), where \( g' \) is the partial derivative of \( g \) with respect to \( u_B \). The assumption that the influence of the noise term \( u_B \) on \( B \) is small corresponds to the fact that we can drop higher order terms in \( u_B \) in the above Taylor expansion. Hence, we can write \( B = g(u_A, 0) + g'(u_A, 0)u_B \).

Without loss of generality, we can assume the noise term is a Gaussian random variable with mean 0. This follows as we can always write \( u_B = h(u'_B) \) where \( u'_B \) is a Gaussian random variable [28], hence \( B = g(u_A, u_B) = g(u_A, h(u'_B)) = g'(u_A, h(u'_B)) \). So \( B \) can be written as a function of a Gaussian variable without loss of generality. This implies that the expected value of \( B \) is \( B = g(u_A, 0) \). Thus the expected value of \( (A, B) \) samples is \((u_A, g(u_A, 0))\). That is, they are both functions of the canonical common cause \( u_A \) alone and can hence be identified from \( A, B \) samples.

This implies that given the canonical causal model associated with any of the structures from Fig. 1, and the assumption of small influence of noise, manifold learning can be utilized—as is done in the CAN algorithm [6, Sec. 3], described in Section II—-to determine the approximate parameterization (up to rescaling) of the canonical “common cause” from samples of the observed \( A, B \). As discussed at the start of this section, one can use this to determine the causal structure underlying \( A \) and \( B \) as follows.

In the case of Fig. 1(a) and (b), the “common cause” corresponds to the variables \( u_A \) and \( u_B \) respectively. In the case of Fig. 1(c) it is the actual common cause, \( C \). Second, given the parameterization of the “common cause,” one can determine the original precanonical causal structure using a purely directed causal discovery algorithm to determine which causal link to the observed variables is underdetermined (if any). If the arrow to \( A \) is underdetermined then the causal structure is Fig. 1(a); if the arrow to \( B \) is underdetermined the structure is Fig. 1(b); if no arrow is underdetermined, the structure is Fig. 1(c).

Our algorithm is described below in Algorithm 1

The experiments performed in Section IV show empirically that Algorithm 1 performs well even on synthetic data when the influence of latent noise is not small. This suggests that the canonical common cause can be learned from \( A, B \) in a wider array of settings. In Section III-D, we set out an open question that would allow the canonical common cause to be

\(^6\)All other cases follow by a similar argument.
Algorithm 1 Our Causal Discovery Algorithm

Input: $A, B$ data, manifold learning algorithm $M$, directed causal discovery algorithm $D$.

Output: Single causal structure from Fig. 1.

1) Run $M$ on $(A, B)$ to obtain parameterisation (up to rescaling, etc.) of canonical common cause $T$ that best fits the data.
2) Implement $D$ between $T\&A$ and $T\&B$
3) if $D$ outputs $T \rightarrow B$ & the causal link between $T\&A$ is underdetermined relative to $D$ do:
   4) Output DAG from Fig. 1(a)
5) else if $D$ outputs $T \rightarrow A$ & the causal link between $T\&B$ is underdetermined relative to $D$ do:
   6) Output DAG from Fig. 1(b)
7) else if $D$ outputs $T \rightarrow B$ & $T \rightarrow A$ do:
   8) Conclude $T$ is a common cause of $A, B$ and output DAG from Fig. 1(c)
9) return DAG output from above

learned from $A, B$ samples under weaker assumptions than that discussed above.

C. Decision Criteria

Due to the pervasiveness of statistical noise, we now introduce a heuristic decision criterion for checking the conditions of Algorithm 1, which is inspired by the decision threshold for KCDC and IGCI discussed in Section II-A. Consider step 3 of Algorithm 1 where the learned canonical common cause is denoted $T$. To simultaneously check whether the causal link between $A$ and $T$ is underdetermined, and also whether $T \rightarrow B$, are consistent with the data, one could apply the following heuristic. Check if

$$\Delta = \frac{|v_{A\rightarrow T} - v_{T\rightarrow A}| - |v_{B\rightarrow T} - v_{T\rightarrow B}|}{\max(|v_{A\rightarrow T} - v_{T\rightarrow A}|, |v_{B\rightarrow T} - v_{T\rightarrow B}|)}$$

lies in the region $(\alpha, 1)$ for some prespecified decision threshold $\alpha$ near 1. This follows as the conditions in step 3 imply $|v_{A\rightarrow T} - v_{T\rightarrow A}| < \delta$ and $|v_{B\rightarrow T} - v_{T\rightarrow B}| > \delta$, for small $\delta$. Thus, the first term in the numerator is generally close to zero, while the second is not.

Alternatively, if $\Delta \in (0, \alpha)$, then the data is consistent with $T \rightarrow B & T \rightarrow A$, as in this case $|v_{A\rightarrow T} - v_{T\rightarrow A}| > \delta$ and $|v_{B\rightarrow T} - v_{T\rightarrow B}| > \delta$. Hence, both terms in the numerator can be roughly thought of as being closer in magnitude than in the case of steps 3 and 5. This allows us to check whether step 7 of Algorithm 1 is true.

Note that the first criterion $\Delta \in (\alpha, 1)$ is also consistent with step 5 of Algorithm 1 as steps 3 and 5 are the same under interchange of $A$ & $B$. Hence if $\Delta \in (\alpha, 1)$, then either step 3 or step 5 is true, and step 7 is false, as per the above. As the two cases in step 3 and 5 are purely directed causal structures, they can be distinguished using the original directed causal discovery algorithm.

Note that the above discussion is heuristic in nature, $\Delta$ is not mathematically constrained to lie in the specific regions described. However, based on the theoretical intuition developed in the last section, it seems a reasonable heuristic to adopt. In the experiments described in Section IV, this heuristic performs very well on both synthetic and real data.

To assign confidence to the outcome of the above heuristic, one could take a bootstrapped approach and calculate the mean of the $\{\Delta_i\}$’s output by running the algorithm on subsamples of the input data, where $\Delta_i$ is the value computed on the $i$th subsample. We suppress the index $i$ in the below for notational ease. As reasoned above, mean$(\Delta) \in (\alpha, 1)$ is consistent with a directed causal structure and mean$(\Delta) \in (0, \alpha]$ with a common cause. Additionally, the variance of the $\{\Delta_i\}$’s calculated in this manner encodes information of the correct DAG. For instance, small variance var$(\Delta) \in (0, \gamma]$ — where $\gamma$ is again a prespecified decision threshold less than 1 — is consistent with a directed causal structure and large variance var$(\Delta) \in (\gamma, 1]$ is consistent with a common cause. This follows because for a directed causal structure only one of the terms in the numerator of each $\Delta_i$ must lie in the region $(0, \delta)$, for small $\delta$, and hence cannot vary much. For a common cause, both terms are outside this region and thus can vary more.

These two implications, high mean$(\Delta)$ indicating a directed cause and high var$(\Delta)$ a common cause, can be used in conjunction to determine the causal structure. For example, it could be the case that a particular common cause structure has moderately high mean$(\Delta)$, but such a high var$(\Delta)$ that a directed cause is ruled out, and a common cause is returned by the algorithm. Similar reasoning can be applied to a directed cause whose mean$(\Delta)$ is lower than expected, but has such a low var$(\Delta)$ that a common cause is ruled out.

To summarize, the above decision criteria are as follows. The mean and variance of $\{\Delta_i\}$’s output by running the algorithm on subsamples of the input data are computed. Recall the possible range of mean$(\Delta) \in [0, 1]$. This range is split into a number of regions. We fix region 1 to be $[0, \alpha_1]$, region 2 to be $[\alpha_1, \alpha_2]$, and so on. Each region $R$ is associated with a threshold value for var$(\Delta)$, $\gamma_R$. If mean$(\Delta)$ lies in region $R$, then if var$(\Delta) \leq \gamma_R$ the algorithm outputs a directed causal structure, whose direction is then determined by the original directed causal discovery algorithm. But if var$(\Delta) > \gamma_R$, a common cause structure is output. Note that as the regions get closer to 0, that is, to $[0, \alpha_1]$, the variance threshold $\gamma_R$ gets increasing small. Hence, $\gamma_R < \gamma_{R+1}$. This encodes our previous reasoning of how mean and variance imply different causal structures. Additionally, there should be a failure mode for very low variance and mean, and very high mean and variance, as these are not indicative of a direct nor a common cause.

As with the decision criteria for the CAN algorithm, described in Section II-A, and KCDC [7], setting such is a challenge, due to the varying impact of noise within experimental datasets. An appropriate method we take in this work is to use data of similar provenance with known causal relations and determine appropriate thresholds for that domain, which are then fixed.

D. Open Questions and Future Extensions

Given the assumption that the strength of latent noise is relatively small, a representation of the canonical common cause can be extracted from samples of the observed $A, B$. The discussion in Section III-B and Intuition Aids 1 and 2 show how this representation can be used to distinguish all three causal structures in Fig. 1.
In the experiments performed in Section IV, and specifically in Section IV-C, it’s shown empirically that while the performance of Algorithm 1 does decrease when the influence of latent noise is not small, it is quite robust to the strength of latent noise. This gives hope that the canonical common cause can be learned from $A, B$ in a wider array of settings than was shown in this section. We now set out an open question that would allow the canonical common cause to be learned from $A, B$ samples under weaker assumptions than that discussed above.

Open Problem 1: Given samples from $A = f(C, u)$ and $B = g(C, v)$, under what conditions can $C$ be learned?

Some progress has been made toward the solution of this open problem by [29], who show that given $A = f(C, u)$ and $B = g(C, v)$, $C$ can be extracted as long as $A$ and $B$ are “sufficiently different,” where, informally, this means the distribution of $u$ and $v$ should vary significantly when conditioning on different values of the $C$.

E. Future Extensions (Detecting Simultaneous Direct and Latent Common Causes)

We now discuss the extension of our Algorithm to the situation where there is both a directed cause and a latent common cause. To distinguish between the two causal structures in Fig. 3, a similar procedure as Section III can be applied. In this case, the directed arrow between $A$ and $B$ can be mathematically replaced by another common cause, resulting in a purely common cause structure with two common causes. When determining if an arrow from one of the common causes can be reversed, to check which of the two structures from Fig. 3 is consistent with the data, all remaining common causes must be conditioned upon or held fixed. The rational behind reversing an arrow is the same as that given in Section III and is graphically outlined in Fig. 3(c).

The difference between this case and the one presented in Section III is the dimensionality reduction algorithm must here output the parameterizations (up to individual rescaling) of two common causes rather than one. However, for the above described reversing of arrows to work, the dimensionality reduction algorithm must output independent and individual parameterizations of each common cause. Hence, nonlinear blind source separation [30] or disentangling of latents [31] algorithms must be employed. This will affect the identifiability of the algorithm, as there are certain functional relationships for which blind source separation fails to return the correct parameterization. This will be explored further in future work.

IV. EXPERIMENTAL RESULTS

We test our method by showing it can turn both the KCDC algorithm of [7] and the IGCI algorithm of [11], which can each only distinguish Fig. 1(a) and (b), into algorithms that can distinguish all causal structures in Fig. 1. We refer to the modified KCDC and IGCI algorithms output by our method as modKCDC and modIGCI respectively. To benchmark performance of modKCDC and modIGCI, we compare against CAN [8] the only previous causal discovery algorithm explicitly designed to distinguish all structures in Fig. 1, which we outlined in Section II-A, as well as the method from [10] which we outlined in Section II. Note that this method can distinguish a direct causal relation from a latent common cause relation, however, if a directed causal link is detected, the algorithm is not designed to orient the causal link. Both the CAN algorithm as well as the algorithm from [10] proved to be very computational intensive to run compared to modKCDC and modIGCI. To reduce computational cost of CAN we only enforced two of the three independence constraints discussed in Section II-A: $u_A, u_B \perp C$.

For KCDC, we use a radial basis function kernel with hyper-parameters fixed in all experiments; there is scope to extend this to multiple kernel majority vote following [7], though performance with a single kernel was effective here. We use the implementation of IGCI from [11]. We use the implementation of CAN outlined in [8], using Gaussian Process regression, Hilbert Schmidt Independence Criterion (HSIC) to test independence, and employing up to five iterations of the independence optimization loops, capped at 5000 iterations of the solver each time. For the manifold learning subroutine of modKCDC, modIGCI, and CAN, we employ Isomap [26]. This is the implementation of manifold learning originally used for CAN [8]. For the algorithm from [10] we used the authors open source implementation. All data was normalized to have mean 0 and variance 1.

We follow the bootstrapping procedure outlined in Section III-C, using 25 bootstraps each randomly sampling 95% of the data. Due to the high computational cost of the algorithm from [10], we did not use the bootstrapping procedure, only applying the algorithm to the full data. This reflects how the algorithm was originally used in [10]’s experiments. As such, we report the performance of this algorithm after the tables in the experiments on synthetic data to set thresholds, we generated synthetic data from directed causal models with additive, multiplicative, and complex noise functions and determined the threshold values which approximately reproduced the performance of the original directed causal discovery algorithms on these different models. We did this for modKCDC and modIGCI, as well as for CAN, whose decision criteria were discussed in Section II-A. This is similar to the way the CAN thresholds were set in [6]. We report the thresholds set in this manner in the supplementary material. These thresholds are used in all experiments.

A. Synthetic Data

We first test on synthetic data for both purely directed and purely common causes. The functions used to generate this synthetic data were adapted from the work of [7], who used them to test performance of KCDC and compare its performance to IGCI. As such, we believe using similar synthetic functions and the same range of synthetic noise
distributions provides a fair and robust benchmark of KCDC, IGCI, and their modifications.

On the synthetic directed datasets, we compare the modified algorithms to the original to determine whether the modification noticeably reduced the performance in detecting directed causal structure. Our experiments show modKCDC and modIGCI preserve the level of performance of the original KCDC and IGCI algorithms.

1) Directed Causal Structures: In the below experiments, we sample 100 datasets each of 250 observations and test modKCDC and modIGCI. The CAN algorithm proved to be very computational intensive to run, taking 300~500 times as long to run on a dataset of size 250 compared to modKCDC and modIGCI. Hence we only tested it on ten datasets, each of 250 observations. For our three algorithms, we sample the normal noise distributions. In (2), this happened in 7/10, 3/10, 5/10 cases. CAN’s poor performance shows the difficulty in developing an algorithm that can distinguish directed and common causes.

The algorithm from [10] performed well, and could detect that all but case (6) were directed causal relationships. Note that this method assumes a linear relationship between variables, so it is perhaps not surprising that it output the wrong answer for case (6), given it high nonlinearity.

2) Common Cause: The common cause setup closely follows that of Section IV-A1. For these synthetic experiments, the common cause T was sampled from T ∈ N(0, 1). Here, CAN was only able to fit a model for experiment (2), below. In all other cases, we could not report CAN’s performance. Results are presented in Table II.

| Experiment | Algorithm | Normal | Uniform | Expon. |
|------------|-----------|--------|---------|--------|
|            | modKCDC  | 99%    | 100%    | 98%    |
| 1          | KCDC     | 100%   | 100%    | 100%   |
|            | modIGCI  | 100%   | 99%     | 100%   |
|            | IGCI     | 100%   | 100%    | 100%   |
|            | CAN      | 40%    | 33%     | 57%    |
|            | modKCDC  | 99%    | 100%    | 98%    |
| 2          | KCDC     | 100%   | 100%    | 100%   |
|            | modIGCI  | 100%   | 100%    | 100%   |
|            | IGCI     | 100%   | 100%    | 100%   |
|            | CAN      | 43%    | 66%     | 50%    |
|            | modKCDC  | 100%   | 100%    | 100%   |
| 3          | KCDC     | 100%   | 100%    | 100%   |
|            | modIGCI  | 100%   | 99%     | 98%    |
|            | IGCI     | 100%   | 100%    | 100%   |
|            | modKCDC  | 99%    | 99%     | 100%   |
| 4          | KCDC     | 100%   | 100%    | 100%   |
|            | modIGCI  | 99%    | 100%    | 96%    |
|            | IGCI     | 100%   | 100%    | 100%   |
|            | modKCDC  | 98%    | 98%     | 98%    |
| 5          | KCDC     | 100%   | 100%    | 100%   |
|            | modIGCI  | 99%    | 98%     | 99%    |
|            | IGCI     | 100%   | 100%    | 100%   |
|            | modKCDC  | 100%   | 97%     | 99%    |
| 6          | KCDC     | 100%   | 100%    | 100%   |
|            | modIGCI  | 29%    | 57%     | 99%    |
|            | IGCI     | 31%    | 100%    | 100%   |

cases. CAN’s poor performance shows the difficulty in developing an algorithm that can distinguish directed and common causes.

The algorithm from [10] performed well, and could detect that all but case (6) were directed causal relationships. Note that this method assumes a linear relationship between variables, so it is perhaps not surprising that it output the wrong answer for case (6), given it high nonlinearity.

In all synthetic direct cause experiments, KCDC and IGCI perform extremely well, achieving 100% accuracy in all situations with the exception of IGCI in experiment (6) with normally distributed noise. In all cases, bar one, modKCDC, and modIGCI either match this performance or come within 1%~2% of it. It is also interesting to note that the one case in which IGCI performed badly, so did modIGCI. modIGCI failed to match the performance of IGCI in one case, however, that of experiment (6) with uniform noise. As IGCI itself performed badly on this experiment with normal noise, it seems to present a challenging dataset for this algorithm, so it is perhaps not surprising that modIGCI yielded poor performance in this case.

The CAN algorithm did not perform well on any of these synthetic datasets, failing to fit a model that complied with the independence criteria in cases beyond additive noise. We only reported accuracy in cases where it successfully fit a model. This in principle gives CAN an added advantage. In (1), this happened in 5/10, 9/10, 7/10 cases for the three different noise distributions. In (2), this happened in 7/10, 3/10, 5/10 situations with the exception of IGCI in experiment (6) with normally distributed noise. In all cases, bar one, modKCDC, and modIGCI either match this performance or come within 1%~2% of it. It is also interesting to note that the one case in which IGCI performed badly, so did modIGCI. modIGCI failed to match the performance of IGCI in one case, however, that of experiment (6) with uniform noise. As IGCI itself performed badly on this experiment with normal noise, it seems to present a challenging dataset for this algorithm, so it is perhaps not surprising that modIGCI yielded poor performance in this case.

The CAN algorithm did not perform well on any of these synthetic datasets, failing to fit a model that complied with the independence criteria in cases beyond additive noise. We only reported accuracy in cases where it successfully fit a model. This in principle gives CAN an added advantage. In (1), this happened in 5/10, 9/10, 7/10 cases for the three different noise distributions. In (2), this happened in 7/10, 3/10, 5/10 situations with the exception of IGCI in experiment (6) with normally distributed noise. In all cases, bar one, modKCDC, and modIGCI either match this performance or come within 1%~2% of it. It is also interesting to note that the one case in which IGCI performed badly, so did modIGCI. modIGCI failed to match the performance of IGCI in one case, however, that of experiment (6) with uniform noise. As IGCI itself performed badly on this experiment with normal noise, it seems to present a challenging dataset for this algorithm, so it is perhaps not surprising that modIGCI yielded poor performance in this case.

The CAN algorithm did not perform well on any of these synthetic datasets, failing to fit a model that complied with the independence criteria in cases beyond additive noise. We only reported accuracy in cases where it successfully fit a model. This in principle gives CAN an added advantage. In (1), this happened in 5/10, 9/10, 7/10 cases for the three different noise distributions. In (2), this happened in 7/10, 3/10, 5/10 situations with the exception of IGCI in experiment (6) with normally distributed noise. In all cases, bar one, modKCDC, and modIGCI either match this performance or come within 1%~2% of it. It is also interesting to note that the one case in which IGCI performed badly, so did modIGCI. modIGCI failed to match the performance of IGCI in one case, however, that of experiment (6) with uniform noise. As IGCI itself performed badly on this experiment with normal noise, it seems to present a challenging dataset for this algorithm, so it is perhaps not surprising that modIGCI yielded poor performance in this case.
Additive and Multiplicative Noise:

\begin{align}
(5) & \quad A = \log(T + 10) + T^6 + n_A \\
& \quad B = (T^2 + T^6)e^{\lambda g} \\
(6) & \quad A = \sin(10T) + T^3 + n_A \\
& \quad B = (T^2 + T^6)e^{\lambda g}.
\end{align}

Here, modKCDC and modIGCI achieve impressive accuracy across a range of different synthetic noise functions—including additive and multiplicative noise—and diverse distributions over these noise terms. Despite the fact that neither KCDC nor IGCI could detect latent common causes, our method turned them into algorithms that could detect them. CAN performed well on the only experiment where it managed to fit a model, (2). Again, accuracy was only reported when it fit a model, in principle giving CAN an added advantage. In (2), this happened in 5/10, 3/10, 1/10 cases for the different noise distributions. CAN failed to fit a model in (1), (3)–(6) above, again demonstrating the difficulty in distinguishing common and directed causes.

The algorithm from [10] did not perform well in these cases. Indeed, in cases (2), (5), and (6) with uniform noise the algorithm output a direct causal relation. In cases (1), (3), and (4) with uniform noise the algorithm was unable to output a clear decision as the scores assigned to the directed and latent common cause structures were within 1% of one another. However, in cases (1) and (3) with uniform noise, the algorithm output the correct decision. Note that this method assumes a linear relationship between variables, so it is perhaps not surprising that it struggled with the above cases, as they all have varying degrees of nonlinearity. This again highlights the difficulty in distinguishing common and directed causes in general contexts.

B. Common Cause Robustness Tests

In the above experiments, we tested the robustness of our algorithms in different noise regimes. We now test robustness to more complex functional relationships. First, we test the accuracy of the algorithms in a regime beyond additive and multiplicative noise. We then test on functions drawn from Gaussian Processes. Results are presented in Table III.

**Complex Noise:**

(3) Let \( A = f(T)e^{\lambda A} + B = g(T)e^{\lambda g} \) with \( f, g \) drawn from the same Gaussian process whose kernel is a sum of polynomial (with only \( T^2 + T^6 \) terms) and periodic exponential.

(4) \( f \) is drawn from a GP with polynomial kernel (only \( T^2 + T^6 \) terms) and \( g \) is drawn from GP with a sum of polynomial (only \( T^3 + T^5 \) terms) and a periodic exponential kernel.

Both modKCDC and modIGCI perform well in these datasets beyond additive and multiplicative noise. Experiments (3) and (4) provide an analysis of robustness to changes in the parameters of the underlying functions.

C. Sensitivity to the Assumption of Small Influence of Latent Noise

As discussed in Section III, a key assumption underlying our heuristic is that in the case where \( A \leftarrow C \rightarrow B \) is the true causal structure, applying manifold learning to \( A, B \) samples should result in an approximate parameterization of \( C \). Intuitively, this happens if the influence of noise terms is small. That is if the influence of \( n_A \) in \( A = f(N(n_A)) \) and \( n_B \) in \( B = g(N(n_B)) \) does not preclude \( C \) from being learned. The sensitivity of our heuristic to this assumption will now be empirically tested using modIGCI. We synthetically generate both directed and common cause data from models which violate the assumption. Directed data are generated by sampling \( A \in N(0, 1) \) and \( n_A \in N(0, 1) \) and inputting to

\[ B = \sin(10A) + e^{3A} + \lambda e^{\lambda g}. \]

Common cause data is generated by sampling \( T \in N(0, 1), u_A \in N(0, 1) \) and inserting to

\[ A = 3T + \lambda e^{\lambda g} \]

\[ B = \log(T + 10) + \lambda e^{\lambda g}. \]

Here, the latent noise terms influence the observed variables via a double exponential function, which violates the assumption underlying Intuition Aid 3. In both cases, the smaller \( \lambda \) is, the smaller the influence of the noise terms. We would expect that the smaller \( \lambda \), the higher the accuracy of our method, and the larger \( \lambda \), the lower the accuracy of our method. This intuition is borne out in the experiments. Results are in Fig. 4. Despite the fact that the noise influences the observed variables via a double exponential function, the accuracy of modIGCI is quite resilient. For instance, when \( \lambda = 1 \) the accuracy on directed data is 46% and the accuracy on common cause data is 62%. Full results are in Fig. 4. This gives hope that identifiability can be demonstrated in a wider array of settings than was shown in Section III.

D. Real Data

1) Directed Data: We now test on real-world data. For this we consider the Tübingen cause-effect pairs dataset,\footnote{https://webdav.tuebingen.mpg.de/cause-effect/} discussed
in [12]. This dataset is comprised of real-world cause-effect samples that are collected across diverse subject areas, from physics to education, with the true causal direction provided by human domain experts.

We use Tübingen cause-effect pairs dataset version 1.0 which consists of 100 cause-effect pairs. Each pair is assigned a weight in order to account for potential sources of bias given that different pairs are sometimes selected from the same multivariable dataset. Following the widespread approach present in the literature of testing only on scalar-valued pairs, we remove pairs 52, 53, 54, 55, and 71 from Tübingen cause-effect pairs dataset to ensure fair comparison to previous work.

For modKCDC and modIGCI, we use the thresholds fixed in Section IV, which are reported in the supplementary material. We tested against KCDC and IGCI and also against the ANM algorithm of [4], a well-known directed causal discovery algorithm discussed in Section II. The results are shown in Table IV.

It is important to note is that both modKCDC and modIGCI achieved largely similar performance to that of KCDC and IGCI, illustrating that our method does not substantially reduce the ability of the original, unmodified algorithms to detect direct causal relations. The performance of KCDC is slightly worse than the performance reported in [7], possibly due to the specific hyperparameter optimization undertaken by those authors. Note that an adaptation of the KCDC approach, with multiple kernel majority vote for each cause-effect pair, is also undertaken in [7] with better accuracies than KCDC with fixed hyperparameters. We did not implement the majority-vote approach here. The performance of IGCI and ANM is also worse than reported in [12], again possibly because of the specific extensive hyperparameter optimization undertaken by those authors. However, the performance of IGCI and ANM reported here is largely similar to that reported in [7].

It is instructive to consider NLSchools data, the 99th pair in the Tübingen cause-effect data. Here “socioeconomic status” is expected to be a direct cause of “language score.” However, some researchers have argued that the correlation between SES and test scores could have a genetic common cause [32], raising the possibility that the underlying structure could be purely common cause. When applied on this data, modKCDC returns mean(Δ) = 0.933 and var(Δ) = 0.007. modKCDC’s output—with its high mean and low variance—is within the thresholds fixed in Section IV, which are reported in Appendix, and is hence indicative of a directed causal structure. Hence, modKCDC returns a direct cause from “socioeconomic status” to “language score,” ruling out a purely common cause explanation of this data and agreeing with the ground truth. A plot of this data is given in Fig. 5(a).

2) Common Cause Data: Next, we consider Breast Cancer Wisconsin (Diagnostic) Data [33]. Here the tumor (malignant or benign) is the common cause of two attributes, “perimeter” and “compactness,” as they are conditionally independent given it. When run on this “perimeter” and “compactness” data, modKCDC returns mean(Δ) = 0.491 and var(Δ) = 0.081, and modIGCI returns mean(Δ) = 0.308 and var(Δ) = 0.062. In both cases, mean(Δ) is relatively small, with var(Δ) in both cases relatively large. From the discussion in Section III-C, we should conclude that both algorithms are consistent with a common cause. In addition, all values are within the thresholds fixed in Section IV, which are reported in the supplementary material, for a common cause. We conclude that both algorithms return a common cause between “perimeter” and “compactness,” agreeing with the ground truth. A plot of this data is given in Fig. 5(b).

---

**Table III**

| Complex Noise Common Cause Experiment | Algorithm | Result |
|---------------------------------------|-----------|--------|
| 1                                     | modKCDC   | 93%    |
|                                       | modIGCI   | 95%    |
| 2                                     | modKCDC   | 89%    |
|                                       | modIGCI   | 91%    |
| 3                                     | modKCDC   | 90%    |
|                                       | modIGCI   | 95%    |
| 4                                     | modKCDC   | 91%    |
|                                       | modIGCI   | 91%    |

**Table IV**

| Tübingen cause-effect pairs dataset | Algorithm | Result |
|-------------------------------------|-----------|--------|
| KCDC                                | 71%       |
| modKCDC                             | 69.6%     |
| IGCI                                | 72%       |
| modIGCI                             | 67.4%     |
| ANM                                 | 59.5%     |

---

Fig. 4. Effect of increasing λ on algorithm accuracy.

Fig. 5. Plots of three real-world datasets used in Section IV-D. (a) NLSchools data, x-axis: language score and y-axis: socioeconomic status. (b) Breast cancer Wisconsin data, x-axis: perimeter of tumor and y-axis: socioeconomic status. (c) AutoMPG data, x-axis: MPG and y-axis: compactness.
Next we consider the AutoMPG dataset from [33]. Here, attributes “acceleration” and “miles per gallon” (MPG) are correlated but have a common cause, the model year. This is checked by performing a conditional independence test between “acceleration” and “MPG” given model year. When applied on this data, modKCDC returns mean(\(\Delta\)) = 0.639 and var(\(\Delta\)) = 0.086, with modIGCI yielding mean(\(\Delta\)) = 0.124 and var(\(\Delta\)) = 0.00062. As per the thresholds reported in the supplementary material, we conclude that modKCDC returns a common cause between “acceleration” and “MPG.” While modIGCI returned a low mean(\(\Delta\)) value—which is indicative of a common cause—that its value for var(\(\Delta\)) is also low, which is not in general consistent with one. Given this relatively large size difference, following the discussion at the end of Section III-C one could conclude that modIGCI’s output is consistent with a common cause between “acceleration” and “MPG.” A plot is given in Fig. 5(c).

V. CONCLUSION

We devised a method to turn a purely directed causal discovery algorithm into one that can also detect latent common causes. In our experiments, we took KCDC [7] and IGCI [11], which each could distinguish directed causes but could not detect latent common causes, and showed our method enabled detection of latent common causes while preserving accuracy in distinguishing directed causes.

REFERENCES

[1] J. Pearl, Causality, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2009.
[2] J. G. Richens, C. M. Lee, and S. Johri, “Counterfactual diagnosis,” 2019, arXiv:1910.06772.
[3] Y. Perov et al., “MultiVerse: Causal reasoning using importance sampling in probabilistic programming,” 2019, arXiv:1910.08091.
[4] P. O. Hoyer, D. Janzing, J. M. Mooij, J. Peters, and B. Schölkopf, “Nonlinear causal discovery with additive noise models,” in Proc. Adv. Neural Inf. Process. Syst., 2009, pp. 689–696.
[5] S. Shimizu, P. O. Hoyer, A. Hyvärinen, and A. Kerminen, “A linear non-Gaussian acyclic model for causal discovery,” J. Mach. Learn. Res., vol. 7, pp. 2003–2030, Dec. 2006.
[6] D. Janzing et al., “Information-geometric approach to inferring causal directions,” Artif. Intell., vol. 182, pp. 1–31, May 2012.
[7] J. Mitrovic, D. Sejdinovic, and Y. W. Teh, “Causal inference via kernel deviance measures,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 6986–6994.
[8] D. Janzing, J. Peters, J. Mooij, and B. Schölkopf, “Identifying confounders using additive noise models,” in Proc. 25th Conf. Uncertainty Artif. Intell., 2009, pp. 249–257.
[9] D. Lopez-Paz, K. Muandet, B. Schölkopf, and I. Tolstikhin, “Towards a learning theory of cause-effect inference,” in Proc. ICML, 2015, pp. 1452–1461.
[10] D. Kalentop and J. Veeken, “We are not your real parents: Telling causal from confounded using MDL,” in Proc. SIAM Int. Conf. Data Mining, 2019, pp. 199–207.
[11] P. Daniissis et al., “Inferring deterministic causal relations,” in Proc. 26th Conf. Uncertainty Artif. Intell. (UAI), 2010, pp. 143–150.
[12] J. M. Mooij, J. Peters, D. Janzing, J. Zscheischler, and B. Schölkopf, “Distinguishing cause from effect using observational data: Methods and benchmarks,” J. Mach. Learn. Res., vol. 17, no. 1, pp. 1103–1204, 2016.
[13] P. Spirtes, C. Glymour, and R. Scheines, Causation, Prediction, and Search (Adaptive Computation and Machine Learning), 2nd ed. Cambridge, MA, USA: MIT Press, 2000.
[14] R. Silva, R. Scheines, C. Glymour, and P. Spirtes, “Learning the structure of linear latent variable models,” J. Mach. Learn. Res., vol. 7, pp. 191–246, Feb. 2006.
[15] P. Spirtes, C. Meek, and T. Richardson, “Causal inference in the presence of latent variables and selection bias,” in Proc. 11th Conf. Uncertainty Artif. Intell. San Mateo, CA, USA: Morgan Kaufmann, 1995, pp. 499–506.
[16] N. Friedman, D. Geiger, and M. Goldszmidt, “Bayesian network classifiers,” Mach. Learn., vol. 29, no. 2, pp. 131–163, Nov. 1997.
[17] D. M. Chickering, “Optimal structure identification with greedy search,” J. Mach. Learn. Res., vol. 3, pp. 507–554, Nov. 2002.
[18] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, “The max-min hill-climbing Bayesian network structure learning algorithm,” Mach. Learn., vol. 65, no. 1, pp. 31–86, 2006.
[19] J. Peters, D. Janzing, and B. Schölkopf, Elements of Causal Inference: Foundations and Learning Algorithms. Cambridge, MA, USA: MIT Press, 2017.
[20] P. O. Hoyer, S. Shimizu, A. J. Kerminen, and M. Palviainen, “Estimation of causal effects using linear non-Gaussian causal models with hidden variables,” Int. J. Approx. Reasoning, vol. 49, no. 2, pp. 362–378, 2008.
[21] K. Zhang and A. Hyvärinen, “On the identifiability of the post-nonlinear causal model,” in Proc. 25th Conf. Uncertainty Artif. Intell., 2009, pp. 647–655.
[22] M. Kocaoglu, S. Shakkottai, A. G. Dimakis, C. Caramanis, and S. Vishwanath, “Applications of common entropy for causal inference,” in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 17514–17525.
[23] D. Janzing, E. Sgouritsa, O. Stegle, J. Peters, and B. Schölkopf, “Detecting low-complexity unobserved causes,” in Proc. 27th Conf. Uncertainty Artif. Intell., 2011, pp. 383–391.
[24] D. Janzing and B. Schölkopf, “Detecting confounding in multivariate linear models via spectral analysis,” J. Causal Inference, vol. 6, no. 1, Mar. 2018.
[25] D. Janzing and B. Schölkopf, “Detecting non-causal artifacts in multivariate linear regression models,” in Proc. 35th Int. Conf. Mach. Learn., 2018, pp. 2245–2253.
[26] J. B. Tenenbaum, V. de Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” Science, vol. 290, no. 5500, pp. 2319–2323, Dec. 2002.
[27] C. M. Lee and R. W. Speckens, “Causal inference via algebraic geometry: Feasibility tests for functional causal structures with two binary observed variables,” J. Causal Inference, vol. 5, no. 2, Mar. 2017.
[28] O. Goudet, D. Kalainathan, P. Caillou, I. Guyon, D. Lopez-Paz, and M. Sebag, “Learning functional causal models with generative neural networks,” in Proc. 27th Conf. Uncertainty Artif. Intell., 2011, pp. 383–391.
[29] D. Jutten and J. Karhunen, “Advances in nonlinear blind source separation,” in Proc. 4th Int. Symp. Independ. Compon. Anal. Blind Signal Separation, 2003, pp. 1–12.
[30] C. Jutten and J. Karhunen, “Advances in nonlinear blind source separation,” in Proc. 4th Int. Symp. Independ. Compon. Anal. Blind Signal Separation, 2003, pp. 1–12.
[31] F. Locatello et al., “Challenging common assumptions in the unsupervised learning of disentangled representations,” 2018, arXiv:1811.12359.
[32] M. Rhenstull and E. M. Tucker-Drob, “Gene-by-socioeconomic status interaction on school readiness,” Behav. Genet., vol. 42, no. 4, pp. 549–558, 2012.
[33] M. Lichman. (2013). UCI Machine Learning Repository. [Online]. Available: http://archive.ics.uci.edu/ml