Causal Relationship Prediction with Continuous Additive Noise Models

Mandar Chaudhary MSCHAUDH@NCSU.EDU
Department of Computer Science
North Carolina State University
Raleigh, NC 27695, USA

Nagiza F. Samatova SAMATOVA@CSC.NCSU.EDU
Department of Computer Science
North Carolina State University
Raleigh, NC 27695, USA

Editor: Thuc Duy Le, Kun Zhang, Emre Kıcıman, Aapo Hyvärinen, and Lin Liu

Abstract

We consider the problem of learning causal relationships in continuous additive noise models (ANM) from a machine learning perspective. Causal discovery from ANMs has primarily focused on testing for independence between the residuals and the true parent set of a variable. We posit that this unique association between residuals and the true parent set can be leveraged with kernel mean embedding to predict causal relationships in observational data. In particular, we propose a framework that finds useful patterns and constructs the causal graph by predicting the true parent set of each variable. We present an analysis of the patterns from kernel mean embeddings that explains their discriminative ability in predicting causal relationships. Finally, we perform simulations that demonstrate the effectiveness of our method.

Keywords: Causal relationship, structural equation models, additive noise, kernel mean embedding, classification

1. Introduction

Distinguishing cause from effect is an important task to reveal meaningful insights in several domains. For example, identifying novel causal associations between climatological factors and seasonal rainfall in a geographical region can aid climate scientists to develop new hypotheses. Likewise in biology, performing gene-knockout experiments can establish causality between a gene and a phenotype (such as disease status). The gold standard approach of identifying such relationships in a system is to carry out randomized control experiments (RCEs). RCEs use external interventions to manipulate a particular factor, then measure the change in the variable of interest. In practice, collecting such experimental data using RCEs might be expensive, time consuming and even unethical. On the other hand, a vast amount of observational data is easily available, but is accompanied by its own set of complexities such as high-dimensionality, missing ground-truth to validate causal relationships and smaller sample size. Causal discovery using observational data has contributed signifi-

cantly towards understanding the mechanism of complex systems (Ebert-Uphoff and Deng, 2014; Maathuis et al., 2010).

Traditional causal discovery methods such as constraint-based, search-and-score and hybrid methods output a partially directed graph that represents a set of Markov equivalent causal graphs. These structures encode the same set of conditional independence relationships but not all causal relationships can be determined. On the other hand, causal discovery using additive noise models (ANM) has gained much attention given their ability of identifying exact cause-effect relationships (Shimizu et al., 2006; Mooij et al., 2009; Hoyer et al., 2009; Janzing et al., 2012; Peters et al., 2014). The presence of non-linearity in the data generating process coupled with additive noise structure has made it possible to fully discover directed acyclic causal graphs. The main idea is to detect asymmetry between the cause and effect using regression and statistical tests to establish causality (Shimizu et al., 2006; Hoyer et al., 2009). This idea has been extended from a two-variable case to construct causal graphs in a multivariate setting (Shimizu et al., 2006; Janzing et al., 2012; Peters et al., 2014). Overall, these methods output better quality of causal graphs which were previously restricted to a set of Markov equivalent graphs.

More recently, the problem of causal discovery in a bivariate case has been presented as a learning problem (Guyon, 2013, 2014). Given two variables X and Y the goal is to predict the causal relationship between them. These competitions have brought forward another opportunity of solving the problem of causal discovery from a machine learning perspective. Encouraging results have been obtained in predicting the true causal relationship between two variables (Lopez-Paz et al., 2015a,b; Fonollosa, 2016). These methods extract features from the input random variables and train classifiers. Their prediction performance has been shown to outperform the state-of-the-art methods. However, there has been no work on constructing a complete multivariate causal graph purely by learning patterns that can predict causal relationships.

In this work, we present a Causal Relationship Prediction in Additive Noise Models (CRPAM) framework that learns patterns from the true parents and the non-parents of a given variable, X, and predicts causal relationships. Informally, we define a non-parent as any variable that is not a parent or an ancestor of X. To this end, we use a characteristic kernel to featurize: 1) the residuals from regressing X on its parents, and 2) residuals from regressing X on its non-parents. Our major contributions are as follows,

1. We present the first method to construct a multivariate causal graph purely by finding discriminative patterns between a variable and its parents, and its non-parents.

2. We develop an approach with kernel mean embedding for creating a training set that is used to train a nonlinear binary classifier.

3. We perform simulations to compare the effectiveness of our method with state-of-the-art causal discovery methods.

4. We present an analysis of the patterns from kernel mean embeddings and illustrate their discriminative ability.

2. Preliminaries

In this section, we define the concepts and notations that will be used throughout the paper.

Definition 1 A causal graph G over a variable set $\mathbf{X}$ is defined as a graph containing directed edges and no directed cycles. A directed edge $X_j \to X_i$ indicates a causal relationship between X_j and X_i, where X_j is the *direct cause* or the *parent* and X_i is the *effect* or the *child*. We denote pa_i as the parent set of X_i.

Definition 2 A *path* in a causal graph G is defined as a sequence of at least two distinct adjacent nodes (variables). A *directed path* between X_i and X_j is a path with all edges oriented in the same direction.

Definition 3 An *ancestor* of variable X_i is a variable that has a directed path towards X_i. We denote an_i as the variable set containing all the ancestors of X_i.

Next, we define two kinds of additive noise models (ANM) namely linear structural equation models (Linear SEMs) and nonlinear structural equation models (Nonlinear SEMs).

Definition 4 (Shimizu et al., 2006): A linear SEM with a causal graph G represents each variable as a linear function of its direct causes and an additive non-Gaussian noise structure,

$$X_i = \sum_{j \in pa_i} \beta_{ij} X_j + \epsilon_i \tag{1}$$

where β_{ij} are non-zero causation coefficients for all $i \in \{1, 2, ..., p\}$ and $j \in pa_i$, and all ϵ_i are mutually independent.

Definition 5 (Hoyer et al., 2009): A nonlinear SEM with a causal graph G represents each variable as a nonlinear function of its direct causes and normally distributed noise structure ϵ_i,

$$X_i = \sum_{j \in pa_i} f_{ij}(X_j) + \epsilon_i \tag{2}$$

where the function $f_{ij}()$ is nonlinear and three times differentiable.

Definition 6: In a causal graph G defined over a variable set $\mathbf{X}$, we define three sets of variables for a variable $X_i \in \mathbf{X}$ with at least one parent,

1. **True parent set** of X_i contains all the variables that are direct causes of X_i. In other words, any variable is adjacent to X_i and has a directed edge towards X_i is part of its true parent set. We denote it as $\mathbf{X_{pa_i}}$.

2. **Non-parent set** of X_i is the set of variables that do not have any causal influence on X_i. Any variable that does not have a direct edge or a directed path towards X_i is a non-parent of X_i. We denote the non-parent set as $\mathbf{X_{non\_pa_i}}$.

3. **Mix parent set** contains all the variables that are ancestors of X_i. We denote the mix parent set as $\mathbf{X_{mix\_pa_i}}$.

We assume there are no hidden common causes of variables in the observational data. In other words, we assume causal sufficiency. We also assume that the underlying causal graph in the observational data is acyclic, i.e., there exist no directed cycles in the graph.

3. Related work

The field of causal discovery has seen several interesting developments over the past few years. Traditional causal discovery methods such as the constraint-based PC algorithm (Spirtes et al., 2000) and its extensions (Colombo and Maathuis, 2014; Colombo et al., 2012; Le et al., 2016; Chaudhary et al., 2017), search-and-score based methods such as Greedy Equivalent Search (GES) (Chickering, 2002), Fast Greedy Equivalent Search (fGES) (Ramsey et al., 2017), and hybrid methods such as Max-Min Hill Climbing (MMHC) (Tsamardinos et al., 2006) have delivered promising results. These methods output a partially directed graph (PDAG) that represents a Markov equivalent class (MEC) of graphs. Every MEC graph encodes the same conditional independence relationship and the true causal graph is expected to be one of the markov equivalent graphs.

Identifying exact causal identities in the output graph has become possible by introducing assumptions about the data generating process. One of the first works that presented promising results was the Linear Non-Gaussian Additive Model (LiNGAM) (Shimizu et al., 2006). LiNGAM is able to recover the entire causal graph assuming that the data generating process is additive, linear and the noise structure follows a non-Gaussian distribution. Inspired by LiNGAM's success, several methods were developed that could recover the exact causal graph. The Post-Nonlinear Model (PNL) was developed to find true causal graphs when the data is generated with nonlinear effect of the causes and the noise variables (Zhang and Hyvärinen, 2008). Another set of methods use nonlinear regression and p-values from kernel-based statistical tests to identify the true parents of a variable (Hoyer et al., 2009; Mooij et al., 2009). The idea behind these methods is to regress a variable on different sets of the remaining variables, and test for independence between the residuals and the variable set. The smallest variable set that leads to independence is considered as the true parent set. This process is repeated for every variable in the data. An improvement over these methods was proposed by using the least dependent residuals instead of relying on the p-values of the hypothesis test (Peters et al., 2014).

Recently, causal discovery has been presented as a learning problem of classifying the causal relationships between two variables (Guyon, 2013, 2014). The training set consists of a large number of cause-effect samples $\{(S_i, l_i)\}_{i=1}^{n}$ where each sample consists of data collected over two random variables X and Y. A sample is a tuple, $\{(x_{ij}, y_{ij})\}_{j=1}^{n_i}$ attested with a binary label l_i which indicates different types of relationships. Two state-of-the-art methods have emerged from these competitions by developing new features that are used to learn the patterns between different types of causal relationships (Lopez-Paz et al., 2015b; Fonollosa, 2016). However, these methods are not developed to build an entire multivariate causal graph.

In this work, we develop the first causal relationship prediction framework that builds a causal graph by predicting the true parent set of each variable. Our framework is developed for causal graphs generated from additive noise models.

4. Method

In this section we present the details of the causal relationship prediction in additive noise models (CRPAM) framework. There are three main steps involved, first simulate causal graphs and for each variable having at least one parent, build regression models by regressing the variable on its parents and non-parents respectively (Section 4.1). Second, embed the distributions of the residuals and the regressors from the regression models with kernel mean embedding (Section 4.2) and third, train a binary classifier to learn the patterns of the parents and the non-parents to predict the causal relationships in the observational data (Section 4.2).

4.1 Data partition for building regression models

Causal discovery between two variables X and Y generated from an additive noise model, $Y = \beta X + \epsilon_Y$, can be performed by fitting two regression models: 1) a forward model $Y \sim X + \epsilon_Y$ and 2) a reverse model $X \sim Y + \epsilon_X$. Assuming the data follows a non-Gaussian distribution, the residuals from the forward model ϵ_Y should be independent of X, and the causality $X \to Y$ would be inferred. The Linear Non-Gaussian Additive Model (LiNGAM) was the first method to develop this idea. Several methods were later developed to exploit this property between the residuals and a parent set to identify the true parent set of each variable.

Our framework borrows the inspiration of the unique association between the residuals and the parent set to learn different patterns. In contrast to existing methods we do not perform statistical tests to identify the parent set. Instead, our aim is to find patterns between the residuals obtained by regressing a variable, X_i, on its parents, and the residuals from non-parents. We define a non-parent set as a set of variables that does not contain the parents and ancestors of X_i (see Definition 6). Regressing a variable, X_i, on its parent set yields a unique set of residuals that cannot be obtained from any other variable set. We posit that this pattern of residuals from the parent set is distinct from the pattern of residuals from any other variable set not containing the parent set. This idea forms the intuition of our method.

Assuming the underlying causal graph is sparse, we simulate data sets from $n_{cg} = 10$ different causal graphs with p variables and sparsity level $p_{con} = 2/(p-1)$, where p is the number of variables in the observational data. We explain the process of creating features from a causal graph, C_g, where $1 \leq g \leq n_{cg}$ and repeat this process n_{cg} times to build the training set. First, we generate $n_{train} = 100$ random data sets of sample size, n, from a simulated causal graph, C_g. Next, we identify the variables having at least one parent in the causal graph. These variables are referred to as the child variables, $\mathbf{X_{child}}$. For each $X_i \in \mathbf{X_{child}}$, we create three groups of variables. The first group contains variables that are parents of X_i denoted by $\mathbf{X_{pa_i}}$, the second group contains variables that are neither parents nor ancestors of X_i denoted by $\mathbf{X_{non\_pa_i}}$, and the third group consists of all the parents and ancestors of X_i, and it is denoted by $\mathbf{X_{mix\_pa_i}}$ (see Definition 6). Note that there is one set of variables in $\mathbf{X_{pa_i}}$ since there can be only one parent set of X_i. As a result, we obtain one set of residuals from the parent set. On the other hand, any variable that does not have a directed path towards X_i is its non-parent. As a result, the number of variable combinations from $\mathbf{X_{non\_pa_i}}$ can be exponential. This creates a significant imbalance in the

distribution of residual patterns that can be learned from the parent set and the non-parent set. To prevent this imbalance, we created a third group of variables called the mix parent set, $\mathbf{X_{mix\_pa_i}}$, which contains all the variables in $\mathbf{X_{pa_i}}$ and any variable that has a directed path towards X_i. We believe that by including the parent set in $\mathbf{X_{mix\_pa_i}}$, the residuals obtained by regressing X_i against the variables in this set would be similar to the residuals obtained from regressing against $\mathbf{X_{pa_i}}$.

In the next step, we build regression models to record the residuals of X_i where the regressors are chosen from: $\mathbf{X_{pa_i}}$, $\mathbf{X_{non\_pa_i}}$, and $\mathbf{X_{mix\_pa_i}}$ respectively. The regression model for the variable set, $\mathbf{X_{pa_i}}$, consists of the parent set as the regressor. On the other hand, there can be an exponential number of combinations to build a regressor set from $\mathbf{X_{non\_pa_i}}$ and $\mathbf{X_{mix\_pa_i}}$ respectively. To prevent an unnecessary computational load of building a large number of regression models, we randomly select few variable sets of different sizes. We create $r_{non\_pa_i}$ sets of randomly chosen variables from $\mathbf{X_{non\_pa_i}}$ where the size of these variable sets can grow from 1 to $s_{non\_pa_i}$. As mentioned earlier, $\mathbf{X_{mix\_pa_i}}$ contains the parents and ancestors of X_i. We build $r_{mix\_pa_i}$ regressor sets of a given size from $\mathbf{X_{mix\_pa_i}}$, where each regressor set is formed by taking the union of all the parents and randomly chosen ancestors. The size of the variable sets chosen from $\mathbf{X_{mix\_pa_i}}$ can grow from $n_{pa_i}+1$ to $n_{pa_i}+s_{mix\_pa_i}$, where n_{pa_i} is the number of variables in $\mathbf{X_{pa_i}}$. Thus, a total of $n_{reg} = 1+r_{non\_pa_i}\cdot s_{non\_pa_i}+r_{mix\_pa_i}\cdot s_{mix\_pa_i}$ regressor sets are generated and an equal number of regression models are built. The residuals and the corresponding regressor sets from n_{reg} regression models are recorded. We refer to a function $v()$ that takes as input a variable, X_i and a variable set to produce a set of residuals and the corresponding regressor sets. This is illustrated as follows,

$$v(X_i, \mathbf{X_{pa_i}}) = \{(\epsilon_{ij}, X_{pa_{ij}})\}_{j=1}^{n_{train}} \tag{3}$$

$$v(X_i, \mathbf{X_{non\_pa_i}}) = \{\{(\epsilon_{ijk}, X_{non\_pa_{ijk}})\}_{j=1}^{n_{train}}\}_{k=1}^{r_{non\_pa_i}\cdot s_{non\_pa_i}} \tag{4}$$

$$v(X_i, \mathbf{X_{mix\_pa_i}}) = \{\{(\epsilon_{ijk}, X_{mix\_pa_{ijk}})\}_{j=1}^{n_{train}}\}_{k=1}^{r_{mix\_pa_i}\cdot s_{mix\_pa_i}} \tag{5}$$

We denote $\mathbf{S_{pa_i}} = \{(\epsilon_{ij}, X_{pa_{ij}})\}_{j=1}^{n_{train}}$ to contain the samples generated from equation 3. Similarly, $\mathbf{S_{non\_pa_i}}$ and $\mathbf{S_{mix\_pa_i}}$ contain the samples from equations 4 and 5 respectively.

4.2 Feature creation with kernel mean embeddings

For any prediction problem, one of the main concerns is to find the feature space that gives the best prediction performance. In order to find the discriminative patterns, we use kernel mean embedding to project the residuals and the corresponding regressor set into a new feature space. Recently, kernel mean embeddings have been used to create features by projecting the probability distribution P over a set of variables in d-dimensional space, $\mathbb{R}^d$, using a kernel function k. Specifically, the randomized causation coefficient (RCC) was developed to featurize the distributions to predict causal relationships in a bivariate case (Lopez-Paz et al., 2015a,b). The mathematical notation of a kernel mean embedding of a

probability distribution P over a variable set in $\mathbb{R}^d$ is shown below,

$$\mu_k(P) = \int_{\mathbb{R}^d} k(x, .)dP(x) \in \mathcal{H}_k \tag{6}$$

where $\mathcal{H}_k$ is the reproducible kernel Hilbert space (RKHS) associated with the kernel function k. The prediction performance of RCC outperformed state-of-the-art methods when evaluated on an independent test set. However, their method has not been developed to construct the full multivariate causal graph. Nonetheless, motivated by their results, we use kernel mean embeddings to create features from the distributions of the residuals and the corresponding set of regressors to build the training set.

In this work, we calculate the embeddings by computing the kernel matrix $\mathbf{K}$ from the empirical distribution. Specifically, we consider the gaussian kernel function, k, for embedding the distributions due to its attractive property of uniquely projecting every distribution into a new feature space i.e., $||\mu_k(P) = \mu_k(Q)||$ iff $P = Q$. The mapping of two samples x and x' with the Gaussian kernel is defined by the following equation,

$$k(x, x') = \exp\big(-\gamma\|x - x'\|_2^2\big) \tag{7}$$

where $\gamma > 0$ and is known as the inverse kernel width.

In the previous subsection we explained how three different groups of variables: $\mathbf{X_{pa_i}}$, $\mathbf{X_{non\_pa_i}}$ and $\mathbf{X_{mix\_pa_i}}$ are used to generate the sample sets, $\mathbf{S_{pa_i}}$, $\mathbf{S_{non\_pa_i}}$ and $\mathbf{S_{mix\_pa_i}}$ (see equations 3-5). Each sample set contains the empirical distribution of the residuals and the regressor sets. For the parent set of a variable X_i, we refer to these distributions as P_{ϵ_i,pa_i} and P_{X,pa_i} respectively. These distributions are featurized by embedding them into kernel matrices using the gaussian kernel function. Thus, for a given pair of residuals and regressor set, we compute three kernel matrices: $\mathbf{K}_{\epsilon_i}$, $\mathbf{K}_{pa_i}$ and $\mathbf{K}_{\epsilon_i,pa_i}$. To summarize the information contained in a kernel matrix, we take the column-wise mean over the matrix and featurize the distributions as illustrated in the following equation,

$$m_k(P_{\epsilon_i,pa_i}, P_{X,pa_i}) = \left\{\left(\overline{\mathbf{K}}_{pa_{il}}, \overline{\mathbf{K}}_{\epsilon_{il}}, \overline{\mathbf{K}}_{\epsilon_{il},pa_{il}}\right)\right\}_{l=1}^{|\mathbf{S_{pa_i}}|} \tag{8}$$

where $\overline{\mathbf{K}}_{pa_{il}}$, $\overline{\mathbf{K}}_{\epsilon_{il}}$, and $\overline{\mathbf{K}}_{\epsilon_{il},pa_{il}}$ represent the column-wise mean vectors of the corresponding kernel matrices $\mathbf{K}_{pa_{il}}$, $\mathbf{K}_{\epsilon_{il}}$ and $\mathbf{K}_{\epsilon_{il},pa_{il}}$.

The output of this equation is a matrix with the number of rows equal to the number of pairs of residuals and regressor sets in $\mathbf{S_{pa_i}}$, and the number of columns equal to $3 \cdot n$ where n is the number of observations in the data. Likewise, the empirical distributions from the sample sets $\mathbf{S_{non\_pa_i}}$ and $\mathbf{S_{mix\_pa_i}}$ are featurized using the above equation.

In the last step we attest a class label for every pair of featurized residual and regressor set that will be used as the ground truth for training a classifier. Recall from section 4.1 that every regressor set from $\mathbf{X_{mix\_pa_i}}$ will be a superset of the parent set, $\mathbf{X_{pa_i}}$, therefore we expect the residual embeddings obtained from these two variable sets to be similar to each other than to $\mathbf{X_{non\_pa_i}}$. Based on this assumption, every pair of featurized residuals and regressor set from $\mathbf{X_{pa_i}}$ and $\mathbf{X_{mix\_pa_i}}$ is labeled as "+1", and the featurized residuals and the regressor sets from $\mathbf{X_{non\_pa_i}}$ are labeled as "0". The embedded distributions of the residuals and the regressor sets: $m_k(P_{\epsilon_i,pa_i}, P_{X,pa_i})$, $m_k(P_{\epsilon_i,non\_pa_i}, P_{X,non\_pa_i})$, and

$m_k(P_{\epsilon_i,mix\_pa_i}, P_{X,mix\_pa_i})$ generated from a causal graph C_g are combined row-wise to form a part of the training set.

$$\begin{bmatrix} (m_k(P_{\epsilon_i,pa_i}, P_{X,pa_i}), +1) \\ (m_k(P_{\epsilon_i,non\_pa_i}, P_{X,non\_pa_i}), 0) \\ (m_k(P_{\epsilon_i,mix\_pa_i}, P_{X,mix\_pa_i}), +1) \end{bmatrix} \tag{9}$$

The above mentioned process is repeated for each variable in causal graph C_g, and then performed for n_{cg} simulated causal graphs. The training set is built by combining all the embedded distributions from n_{cg} graphs.

A random forest classifier is used to learn the patterns of parents and non-parents from the training set. The classifier is then used to predict causal relationships in the observational data. Prior to making predictions, we need to convert the observational data in the same feature space as the training set. The projection of the observational data into this new feature space is simpler than the training set since the parents and non-parents are not known. Thus we do not have to create separate sample sets for creating features. Instead, for each variable X_j in the observational data, we create a combination of variables that will be considered as its parent set. These variable sets are featurized as per the steps mentioned in Sections 4.1 and 4.2. From hereon, we will refer to the transformed observational data as the test set. While generating combinations of variables, the size of the combinatorial variable set can grow from 1 to s_{test}. Note that we do not set the value of s_{test} higher than the maximum size of the variable set used to build the training set. In other words, s_{test} is always less than or equal to $max(s_{non\_pa_i}, n_{pa_i} + s_{mix\_pa_i})$. Deciding the value of s_{test} is a trade-off between time consumption and accuracy. Initializing s_{test} to a high value creates exponentially more combinations of variables for building the test set. As a result, this increases the runtime to featurize the distributions.

Once the classifier is trained, for each variable in the test set we predict whether a variable combination is closer to the pattern of a parent set or a non-parent set. Ideally, if there exists a parent set of a variable, the classifier should predict only the variable combination containing the parent set as the true parent set. Unfortunately, this does not happen in practice and the classifier often predicts more than one combination of variables as the true parent set. We develop a heuristic to aggregate these predictions. To do this, we keep a track of the frequency of each variable being predicted as the true parent set. In the end, the predicted parent set contains the variables having more than 50% frequency.

5. Simulations

We perform a series of simulations to evaluate the effectiveness of the proposed causal relationship prediction framework.

5.1 Synthetic Data

We simulate $n_{cg} = 10$ synthetic sparse causal graphs of p variables to build the training set. The value of p is equal to the number of variables in the observational data. To enforce sparsity, we set the probability of an edge being present between two variables to be $p_{con} = 2/(p-1)$. The resulting causal graph is expected to have p edges which is considered as a sparse setting (Peters et al., 2014). We generate $n_{train} = 100$ synthetic datasets for

each causal graph and leverage the true causal relationships to build the training set (see Section 4). While building the regression models, for any child node, $X_i \in \mathbf{X_{child}}$, we initialize $r_{non\_pa} = 3$, $r_{mix\_pa} = 2$, $1 \leq s_{non\_pa_i} \leq 4$, and $1 \leq s_{mix\_pa_i} \leq 3$. These values are chosen after conducting several experiments and observing their impact on runtime. We test the framework to predict causal relationships in test set with variables $p = \{10, 15\}$ across two different sample sizes, $n = \{100, 200\}$. While embedding the test data into the new feature space, we initialize $s_{test} = 5$ for $p = 10$, and $s_{test} = 3$ for $p = 15$. We initialize the inverse kernel width, $\gamma = 1$ to generate the kernel embeddings as shown in equation 7.

We generate 20 test sets from a linear as well as a non-linear setting for a given set of variables and sample size. For the linear setting, we use linear regression and for the non-linear setting we use generalized additive model regression to obtain the residuals. The output causal graph is evaluated in terms of the structural hamming distance (SHD) which calculates the number of edges to be added, removed or flipped in the estimated causal graph to match the true causal graph; the accuracy value $d = \sqrt{(1 - precision)^2 + (1 - recall)^2}$ where precision is the fraction of true edges found in the estimated causal graph, and recall is the fraction of true edges in the estimated graph that are also present in the true causal graph.

We compare the performance of CRPAM against regression with subsequent independence test (RESIT), greedy DAG search (GDS), linear non-Gaussian additive models (LiNGAM), the PC algorithm with Fisher's Z test and significance level of 0.05, the greedy equivalence search (GES), and max-min hill climbing algorithm (MMHC).

5.1.1 Models for Data Generation

We simulate training set and test set from two kinds of models: linear structural equation models and non-linear structural equation models (see Definition 4 and 5). We follow the procedure mentioned in (Peters et al., 2014) and use their publicly available code[1] to simulate causal graphs and synthetic data sets. In the linear setting, each variable is a linear combination of its parents and an additive non-Gaussian noise variable (see Definition 4). The coefficients β_{ij} are uniformly chosen from [-2, -0.1] $\cup$ [0.1, 2] and the noise variables are independent and distributed according to $E_i \cdot \text{sign}(M_i) \cdot |M_i|^{\alpha_i}$ where M_i is normally distributed with mean 0 and standard deviation 1, E_i is uniformly distributed between [0.1, 0.5] and α_i is also uniformly distributed between [2, 4]. We also generate test data sets from nonlinear SEMs (see Definition 5). The nonlinear functions f_{ij} are sampled from a Gaussian process with bandwidth one.

Table 1 summarizes the performance of the CRPAM framework against state-of-the-art causal discovery methods with Linear SEMs. We observe that CRPAM has the lowest SHD in two out of four cases and the next best SHD in the remaining two cases. The performance of CRPAM improves with increasing sample size. The PC and GES algorithms have the lowest accuracy value d in most cases with GES having slightly better mean SHD values than PC. However, their SHD values are not the lowest which indicates that the output graph contains false positives or the output edges have incorrect orientations or it could be both. LiNGAM and MMHC have the lowest SHD values in one case each. However, LiNGAM performs better with increasing sample size and its SHD values are lower than

1. `https://github.com/bquast/ANM/tree/master/codeANM`

Table 1: Performance metrics for Linear Structural Equation Models on sparse causal graphs, $p_{con} = 2/(p-1)$. The reported metric is its mean value over 20 simulations. The best performance is highlighted.

	CRPAM	RESIT	GDS	LiNGAM	PC	GES	MMHC
$p = 10, n = 100$							
SHD	7.3 ± 3.3	14.7 ± 3.8	8.3 ± 4.5	7.5 ± 2.8	8.4 ± 3.4	7.3 ± 2	6.6 ± 3.9
d	0.4 ± 0.2	0.7 ± 0.1	0.6 ± 0.3	0.7 ± 0.1	0.4 ± 0.2	0.3 ± 0.1	0.4 ± 0.2
$p = 15, n = 100$							
SHD	8.7 ± 3.5	25.9 ± 11.8	14.3 ± 5.2	10.2 ± 2.4	11.6 ± 3.6	10.7 ± 3.1	9.8 ± 3.1
d	0.5 ± 0.2	0.8 ± 0.1	0.7 ± 0.2	0.7 ± 0.1	0.4 ± 0.1	0.4 ± 0.1	0.4 ± 0.1
$p = 10, n = 200$							
SHD	3.8 ± 2.8	15.8 ± 7.7	4.9 ± 3.2	4.4 ± 3	7 ± 3.6	6.7 ± 4	6 ± 3.5
d	0.3 ± 0.2	0.6 ± 0.2	0.3 ± 0.2	0.4 ± 0.2	0.3 ± 0.2	0.3 ± 0.2	0.4 ± 0.2
$p = 15, n = 200$							
SHD	7.3 ± 3.4	36.6 ± 10.7	11.1 ± 5.5	6.8 ± 2.8	11 ± 3.7	10.3 ± 3.8	8.4 ± 3.4
d	0.4 ± 0.1	0.8 ± 0.1	0.5 ± 0.2	0.4 ± 0.1	0.3 ± 0.1	0.3 ± 0.1	0.4 ± 0.2

all the methods except CRPAM. While the accuracy value d of CRPAM is not the best in all cases, it is consistently better than RESIT, GDS, LiNGAM and MMHC. There is no method that performs the best in all cases across both performance metrics. Nonetheless, one can conclude that CRPAM and LiNGAM seem to perform better compared to other methods.

Table 2 presents the results with non-linear SEMs. We observe that CRPAM significantly outperforms all the other methods. The mean values of SHD and accuracy value d output by CRPAM are the lowest in all cases. There is a minimum difference of at least 3 and 0.2 in the mean values of SHD and d respectively when compared with the next best performing method. Note that both CRPAM and RESIT make use of the nonlinearity of the functions in the SEMs to identify causal relationships. However, the features extracted by CRPAM seem to have greater discriminative power in identifying the parents than RESIT. These results suggest that there is a distinct pattern in the embeddings of the parents that differentiates them from the non-parents.

Motivated by the performance of CRPAM with nonlinear SEMs we also perform another set of experiments. We increase the complexity of the simulated causal graphs by setting $p_{con} = 2 \times 2/(p-1)$ so that the probability of an edge between two variables is twice that in sparse causal graphs. We refer to this setting as dense causal graphs. Next, we generate data with $p = \{10, 15\}$, $n = 100$ and compare the performance with the other methods. All the parameters used in the previous experiments remain the same as mentioned in Section 5.1. The results are presented in Table 3. We observe that CRPAM continues to outperform all the methods in terms of both SHD and accuracy d. The next best performing method has 1.5 times the mean SHD value than CRPAM and at least 1.4 times the mean accuracy d.

Table 2: Performance metrics for Nonlinear Structural Equation Models on sparse causal graphs, $p_{con} = 2/(p-1)$. The reported metric is its mean value over 20 simulations. The best performance is highlighted.

	CRPAM	RESIT	GDS	LiNGAM	PC	GES	MMHC
$p = 10, n = 100$							
SHD	4.6 ± 3.3	8.9 ± 3.6	9 ± 4.9	10.9 ± 3	10.7 ± 3.3	14 ± 4.6	9.8 ± 3.2
d	0.2 ± 0.2	0.6 ± 0.2	0.5 ± 0.2	0.8 ± 0.2	0.5 ± 0.1	0.6 ± 0.2	0.6 ± 0.1
$p = 15, n = 100$							
SHD	4.6 ± 2.9	14 ± 5.3	16.6 ± 6	14 ± 3.4	13.6 ± 3.5	22.7 ± 5.1	12 ± 4.2
d	0.3 ± 0.2	0.7 ± 0.2	0.6 ± 0.2	0.9 ± 0.2	0.5 ± 0.2	0.8 ± 0.2	0.5 ± 0.2
$p = 10, n = 200$							
SHD	2.7 ± 2.5	5.8 ± 4	8 ± 5.1	10 ± 3	9.3 ± 3.2	13.3 ± 5	8.2 ± 3.3
d	0.1 ± 0.1	0.4 ± 0.2	0.5 ± 0.1	0.5 ± 0.2	0.5 ± 0.2	0.7 ± 0.2	0.5 ± 0.2
$p = 15, n = 200$							
SHD	2.7 ± 3	9 ± 3.7	13.9 ± 8	14.4 ± 3.8	12.3 ± 3.5	23.3 ± 7.4	11.1 ± 4.4
d	0.2 ± 0.2	0.4 ± 0.1	0.5 ± 0.2	0.6 ± 0.1	0.5 ± 0.1	0.7 ± 0.1	0.4 ± 0.2

Table 3: Performance metrics for Nonlinear Structural Equation Models on dense causal graphs, $p_{con} = 2 \times 2/(p-1)$. The reported metric is its mean value over 20 simulations. The best performance is highlighted.

	CRPAM	RESIT	GDS	LiNGAM	PC	GES	MMHC
$p = 10, n = 100$							
SHD	11.2 ± 4.6	17.3 ± 5.6	20.4 ± 5.4	18.8 ± 3.9	18.2 ± 3.6	21.6 ± 4.5	16.6 ± 3.4
d	0.3 ± 0.1	0.6 ± 0.1	0.6 ± 0.1	0.9 ± 0.1	0.7 ± 0.1	0.6 ± 0.1	0.7 ± 0.1
$p = 15, n = 100$							
SHD	16.7 ± 8.3	32 ± 8.8	36.1 ± 9	29.5 ± 6.4	28.2 ± 6.9	39.2 ± 7.8	26.4 ± 7.5
d	0.5 ± 0.2	0.8 ± 0.1	0.7 ± 0.1	0.9 ± 0.1	0.7 ± 0.1	0.7 ± 0.1	0.7 ± 0.1

5.1.2 Feature Analysis

The prediction performance of CRPAM on linear SEMs and nonlinear SEMs is driven by the features created using kernel mean embedding (see Section 4.2). We provide some understanding behind the behavior of our framework by presenting a visual analysis of the distribution of the features in the training sets and their importance in training the classifier. Recall from equation 9 that the distributions of the residuals and the corresponding regressor sets across three sample sets were embedded into a new feature space. The features from the true parent set are stored in $m_k(P_{\epsilon,pa}, P_{X,pa})$, the features from the non-parent set are stored in $m_k(P_{\epsilon,non\_pa}, P_{X,non\_pa})$, and the features from the third group with the mix parent set are stored in $m_k(P_{\epsilon,mix\_pa}, P_{X,mix\_pa})$.

Figures 1 and 2 summarize the information contained in the training set with $p = 10$ and $n = 100$ for linear SEMs and nonlinear SEMs respectively. In particular, we present the mean and standard deviation values of the features in the training set. For sample size $n = 100$ the training set has $3 \cdot n$ features where the first hundred features are the embeddings

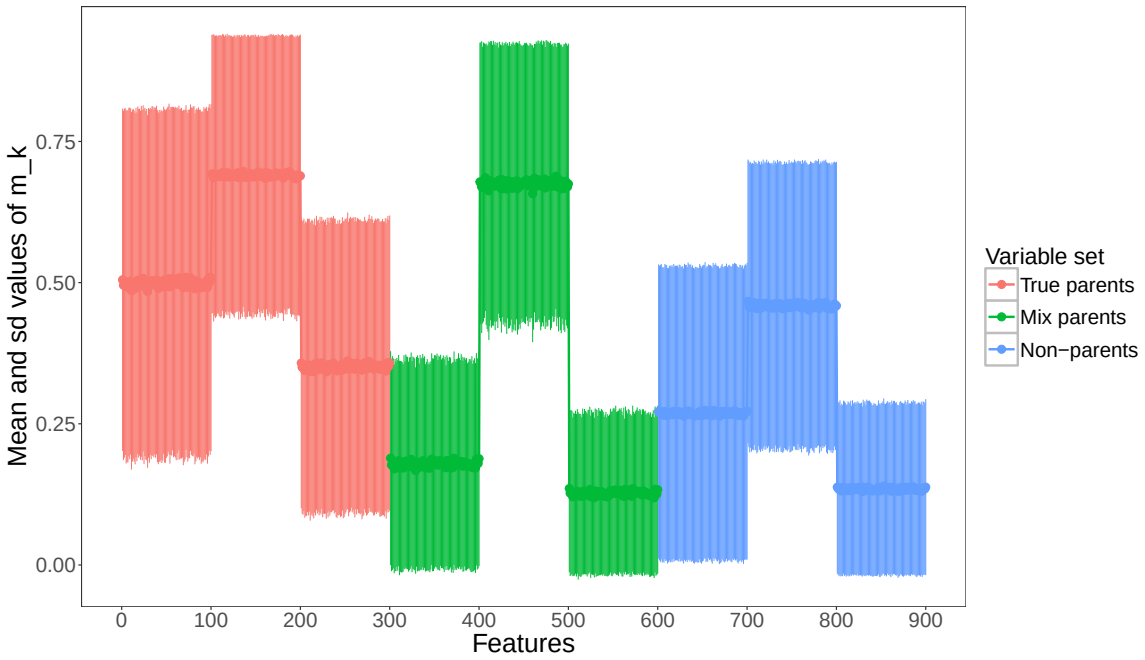


Figure 1: The mean and standard deviation values of $m_k(P_{\epsilon,pa}, P_{X,pa})$ (1:300), $m_k(P_{\epsilon,mix\_pa}, P_{X,mix\_pa})$ (301:600), and $m_k(P_{\epsilon,non\_pa}, P_{X,non\_pa})$ (601:900) for all the child variables in the training set with $p = 10$ and $n = 100$ for linear SEMs. The first hundred features in a variable set represent the embedding of the regressor set, the next hundred represent the embedding of residuals and the last hundred represent the embedding of both. The true parent set (pink) and the mix parent set (green) are assigned the same class label and the non-parent set (blue) is assigned a different class label.

of the regressor set, the next hundred features are the embeddings of the residuals and the last hundred are the embeddings of both the regressor set and the residuals (see equation 8). In figure 1, we observe that the mean and standard deviation values of the residual embeddings of the true parent set and the mix parent set are similar to each other than the non-parent set. Although there is some discriminative pattern between the two classes, there is a high overlap between their values. This can potentially explain the prediction performance of CRPAM on linear SEMs (see Table 1).

Figure 2 presents the same information for nonlinear SEMs. We observe a distinct pattern in the residual embeddings of the true parent set, mix parent set and the non-parent set with very little overlap. These features are very similar for the true parent set and the mix parent set but different for the non-parent set. The mean values of the residual embeddings from the true parent set and the mix parent set are centered around 0.9 with a very small standard deviation. On the other hand, the same values for non-parent sets are centered around 0.7 with a much higher standard deviation. We believe this discriminative pattern would be leveraged by the classifier to accurately predict the true causal relationships. This pattern also supports our assumption made in Section 4.2 about the residual embeddings being similar for the true parent set and the mix parent set.

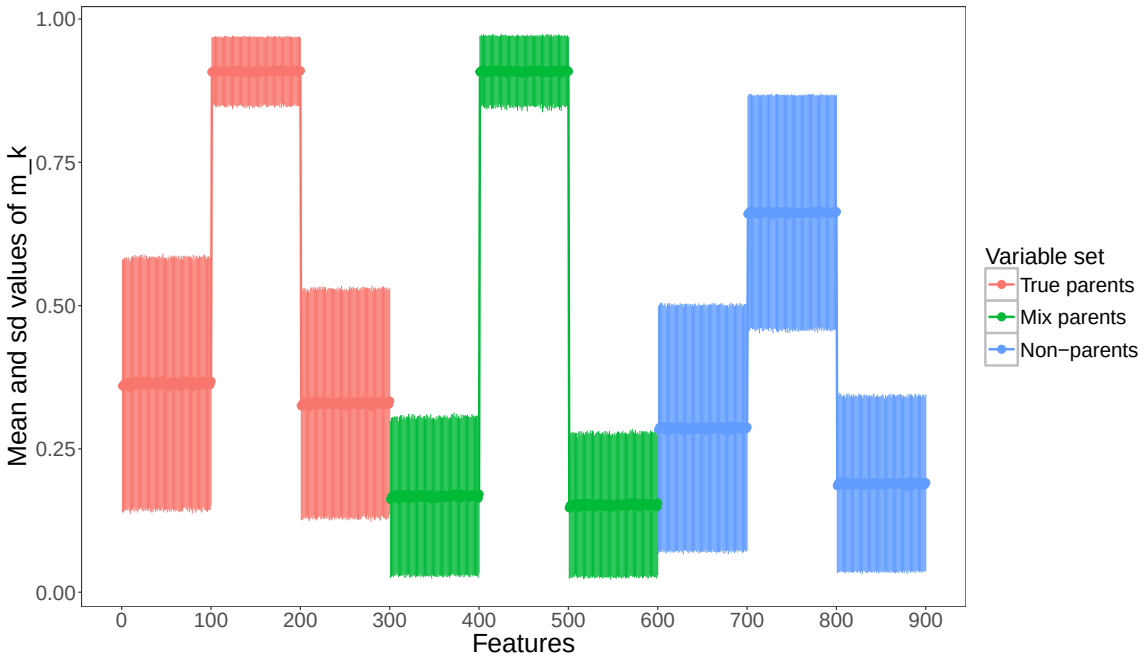
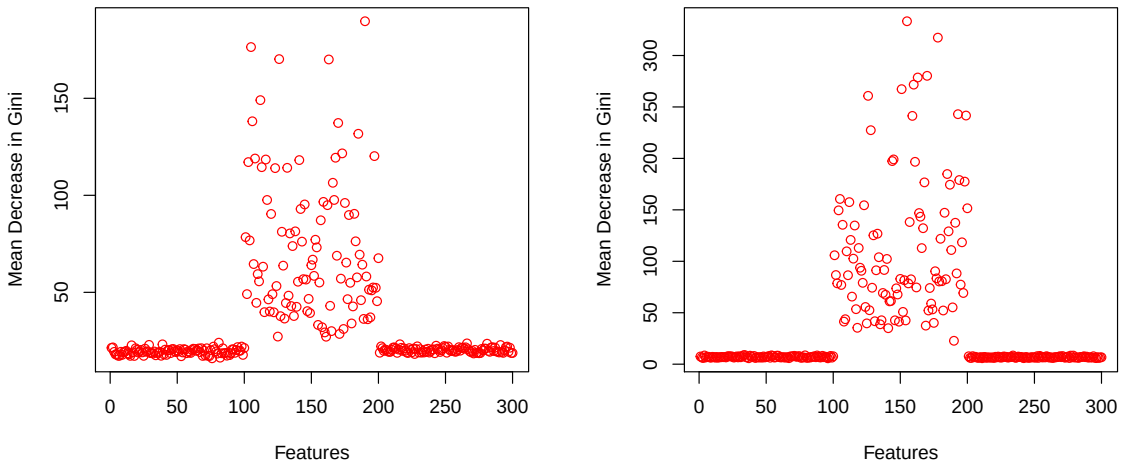


Figure 2: The mean and standard deviation values of $m_k(P_{\epsilon,pa}, P_{X,pa})$ (1:300), $m_k(P_{\epsilon,mix\_pa}, P_{X,mix\_pa})$ (301:600), and $m_k(P_{\epsilon,non\_pa}, P_{X,non\_pa})$ (601:900) for all the child variables in the training set with $p = 10$ and $n = 100$ for nonlinear SEMs. The first hundred features in a variable set represent the embedding of the regressor set, the next hundred represent the embedding of residuals and the last hundred represent embedding of both. The true parent set (pink) and the mix parent set (green) are assigned the same class label and the non-parent set (blue) is assigned a different class label.

Lastly, we analyze the features based on their ability to discriminate the patterns of the true parent sets and the non-true parent sets. Figure 3 shows the feature importance from the random forest classifier for linear SEMs and nonlinear SEMs. The importance of a feature is measured by the mean decrease in Gini if that variable was included in training the classifier. In figure 3 we observe that the features created by embedding the residuals are the most important to the classifier as they lead to highest reduction in mean Gini values. This confirms our observations from figures 1 and 2. Additionally, this also provides insight into reducing the feature space by excluding the other two sets of features containing the regressor sets. It would be interesting to experiment with only the features containing the residual embedding given this evidence. The benefits would be twofold: one, this would lead to a 66% reduction in the number of features and two, it would significantly reduce the runtime of the framework. We leave these experiments as part of our future work.

6. Conclusions and future work

In this work, we have approached the problem of causal discovery as a learning problem. The proposed framework builds on the ideas of additive models to create discriminative patterns for true parent sets and non-parent sets from kernel mean embeddings. A nonlinear binary



(a) Linear SEMs

(b) Nonlinear SEMs

Figure 3: The importance of features in training set with $p = 10$ and $n = 100$ in linear SEMs (left) and nonlinear SEMs (right). The features are represented on the X-axis and the variable importance is represented on the Y-axis. The importance of a variable is measured by the mean decrease in gini if the variable were included in training the classifier. The first hundred features represent the embedding of the parent set, the next hundred features represent the embedding of the residuals and the last hundred features represent embedding of both.

classifier is trained to learn these patterns and predict the parent set of each variable in the test data. The framework is evaluated on linear SEMs and nonlinear SEMs to demonstrate its prediction performance. Finally, we present evidence of a strong discriminative pattern of the features in nonlinear SEMs which gives insight into the performance of the framework.

We have planned the research progress of the framework in three directions. First, perform an analysis of the different parameters used in the framework for differing sizes of the data and network complexity. Second, develop a parallel version of the framework since all the steps can be easily parallelized. A parallel CRPAM framework would greatly reduce the runtime and provide better scalability. Third, develop ensemble methods for data sets with large sample size that extracts patterns from different parts of the data and performs majority voting to predict the causal relationships.

Acknowledgments

This material is based upon work supported by the NSF grant 1029711. We thank the anonymous reviewers for their feedback.

References

Mandar S Chaudhary, Stephen Ranshous, and Nagiza F Samatova. A community-driven graph partitioning method for constraint-based causal discovery. In *International Workshop on Complex Networks and their Applications*, pages 253–264. Springer, 2017.

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

Diego Colombo and Marloes H Maathuis. Order-independent constraint-based causal structure learning. *Journal of Machine Learning Research*, 15(1):3741–3782, 2014.

Diego Colombo, Marloes H Maathuis, Markus Kalisch, and Thomas S Richardson. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, pages 294–321, 2012.

Imme Ebert-Uphoff and Yi Deng. Causal discovery from spatio-temporal data with applications to climate science. In *Machine Learning and Applications (ICMLA), 2014 13th International Conference on*, pages 606–613. IEEE, 2014.

José AR Fonollosa. Conditional distribution variability measures for causality detection. *arXiv preprint arXiv:1601.06680*, 2016.

Isabelle Guyon. Cause-effect pairs kaggle competition, 2013. URL `https://www.kaggle.com/c/cause-effect-pairs/`.

Isabelle Guyon. Chalearn fast causation coefficient challenge, 2014. URL `https://www.codalab.org/competitions/1381`.

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

Dominik Janzing, Joris Mooij, Kun Zhang, Jan Lemeire, Jakob Zscheischler, Povilas Daniušis, Bastian Steudel, and Bernhard Schölkopf. Information-geometric approach to inferring causal directions. *Artificial Intelligence*, 182:1–31, 2012.

Thuc Le, Tao Hoang, Jiuyong Li, Lin Liu, Huawen Liu, and Shu Hu. A fast pc algorithm for high dimensional causal discovery with multi-core pcs. *IEEE/ACM transactions on computational biology and bioinformatics*, 2016.

David Lopez-Paz, Krikamol Muandet, and Benjamin Recht. The randomized causation coefficient. *The Journal of Machine Learning Research*, 16(1):2901–2907, 2015a.

David Lopez-Paz, Krikamol Muandet, Bernhard Schölkopf, and Iliya Tolstikhin. Towards a learning theory of cause-effect inference. In *International Conference on Machine Learning*, pages 1452–1461, 2015b.

Marloes H Maathuis, Diego Colombo, Markus Kalisch, and Peter Bühlmann. Predicting causal effects in large-scale systems from observational data. *Nature Methods*, 7(4):247, 2010.

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

Jonas Peters, Joris M Mooij, Dominik Janzing, and Bernhard Schölkopf. Causal discovery with continuous additive noise models. *The Journal of Machine Learning Research*, 15 (1):2009–2053, 2014.

Joseph Ramsey, Madelyn Glymour, Ruben Sanchez-Romero, and Clark Glymour. A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *International journal of data science and analytics*, 3(2):121–129, 2017.

Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, and Antti Kerminen. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7 (Oct):2003–2030, 2006.

Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT press, 2000.

Ioannis Tsamardinos, Laura E Brown, and Constantin F Aliferis. The max-min hill-climbing bayesian network structure learning algorithm. *Machine learning*, 65(1):31–78, 2006.

Kun Zhang and Aapo Hyvärinen. Distinguishing causes from effects using nonlinear acyclic causal models. In *Proceedings of the 2008th International Conference on Causality: Objectives and Assessment-Volume 6*, pages 157–164. JMLR. org, 2008.