Computational Statistics and Data Analysis 156 (2021) 107141

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Causal network learning with non-invertible functional relationships

Bingling Wang [a], Qing Zhou [b,*]

[a] Department of Biostatistics, University of California, Los Angeles, USA
[b] Department of Statistics, University of California, Los Angeles, USA

ARTICLE INFO

Article history:
Received 14 July 2020
Received in revised form 29 September 2020
Accepted 2 November 2020
Available online 12 November 2020

Keywords:
Causal discovery
Directed acyclic graph
Structural equation model
Nonlinearity
Non-invertible relation

ABSTRACT

Discovery of causal relationships from observational data is an important problem in many areas. Several recent results have established the identifiability of causal directed acyclic graphs (DAGs) with non-Gaussian and/or nonlinear structural equation models (SEMs). Focusing on nonlinear SEMs defined by non-invertible functions, which exist in many data domains, a novel test is proposed for non-invertible bivariate causal models. Algorithms are further developed to incorporate this test in structure learning of DAGs that contain both linear and nonlinear causal relations. Extensive numerical comparisons show that the proposed algorithms outperform existing DAG learning methods in identifying causal graphical structures. The practical application of the methods is illustrated by learning causal networks for combinatorial binding of transcription factors from ChIP-Seq data.

1. Introduction

Inferring causal relations from data is a fundamental problem in many areas of science. Randomized controlled experiments are the gold standard tool used for causal discovery. However, there are certain limitations, such as expenses, time, ethics, practicalities etc., in the application of randomized experiments. Even when experiments are possible to carry out, with hundreds and thousands of variables easily collected nowadays, performing a large number of experiments on these variables is unrealistic when background knowledge is limited. Identifying causal relationships from observational data has therefore attracted much attention from many researchers in the past few decades (Verma and Pearl, 1990; Meek, 1995; Chickering, 1996; Heckerman et al., 2006; Pearl, 2009; Spirtes, 2010).

In this paper, we model causal relations among a set of random variables by a directed acyclic graph (DAG), following Pearl (2009). Under this approach, causal structure learning is achieved by estimating the structure of the underlying causal DAG from observed data. Traditional structure learning methods can be classified into two categories. The first category is the constraint-based approach which seeks to recover the underlying graphical structure by identifying conditional independence relationships between variables. Examples of constraint-based algorithms include the PC algorithm by Spirtes and Glymour (1991) and the Fast Causal Inference (FCI) algorithm by Spirtes et al. (2000). The second category is the score-based approach that aims to find the causal DAG by maximizing certain scoring function, e.g. Bayesian Dirichlet scores, Bayesian information criterion, or regularized likelihood among others. Algorithms in this category, such as Heckerman et al. (1995), Chickering (2003), search the space of graphs for an optimal structure using greedy, local, or some other search strategies.

* Corresponding author.
E-mail address: zhou@stat.ucla.edu (Q. Zhou).

https://doi.org/10.1016/j.csda.2020.107141
0167-9473/

For continuous data, most existing DAG learning methods assume linear parent–child relations with additive Gaussian noises, sometimes called linear Gaussian DAGs (Pearl, 2009; Spirtes et al., 2000). Although models under assumptions of linearity and Gaussianity are well understood and convenient to work with, they are not always realistic in real-world applications. It is arguable that most causal relations in real data are more or less nonlinear in nature. Moreover, linear Gaussian DAGs are not identifiable from observational data. Every DAG in the Markov equivalence class of the true causal DAG gives identical likelihood of observational data and implies an identical set of conditional independence relations. Therefore, neither constraint-based nor score-based approaches can identify the causal DAG. This is the well-known non-identifiability issue of linear Gaussian DAGs. Consider a simple problem of inferring the causality between two variables, whether X causes Y or vice versa. In terms of DAGs, we are considering either $X \to Y$ or $Y \to X$. Under linear and Gaussian assumptions, the two DAGs merely represent two ways to factorize the same bivariate Gaussian density $p(y|x)p(x) = p(x|y)p(x)$, and thus one cannot distinguish the two causal models from observational data in this case.

In recent years, many efforts have been made to tackle the causal discovery problem from different perspectives under various identifiability assumptions (Shimizu et al., 2006; Zhang and Hyvärinen, 2008, 2009; Hoyer et al., 2009; Mooij et al., 2010; Shimizu et al., 2011; Peters et al., 2014; Peters and Bühlmann, 2014; Peters et al., 2016; Blöbaum et al., 2018; Ghoshal and Honorio, 2018; Monti et al., 2019; Ghoshal and Honorio, 2018). In particular, a few methods have been proposed to identify the true causal DAG from observational data by making use of nonlinear and/or non-Gaussian structural equation models (SEMs). Shimizu et al. (2006) showed that the true causal DAG is identifiable assuming non-Gaussian errors under linear SEMs and proposed a linear non-Gaussian acyclic model (LiNGAM) for causal structure learning. Hoyer et al. (2009) pointed out that nonlinearity can break the symmetry between observed variables, which leads to identifiable causal models, and proposed a nonlinear additive noise model which was further extended and implemented by Peters et al. (2014). Zhang and Hyvärinen (2009) proposed a post-nonlinear (PNL) causal model under which one can distinguish the cause from effect and investigated conditions for identifiability of the model. See Mooij et al. (2016) and Glymour et al. (2019) for recent reviews of relevant works.

A key ingredient in the above methods is the use of general independence tests to determine the causal directions (Shimizu et al., 2006, 2011; Peters et al., 2014; Mooij et al., 2016). Take the simple bivariate case as an example. Suppose the true causal model is $X \to Y$ so that the corresponding SEM is $Y = f(X)+\epsilon$, where the noise ϵ is independent of the causal parent X. If f is a nonlinear function satisfying some mild conditions, one cannot find a function $g(\cdot)$ such that $X = g(Y)+\epsilon'$ and that ϵ' is independent of Y. Thus, to identify the correct causal DAG, one must test whether the residual after a nonlinear regression of Y onto X is independent of X. This is in general a very difficult problem, since in many regression techniques the residual is uncorrelated with X by design. Thus, advanced and complex test procedures such as the Hilbert–Schmidt Independence Criterion (HSIC) (Gretton et al., 2005), a kernel-based independence test, is often used in this approach. To estimate a causal DAG on many variables, a sequence of such independence tests is usually performed by these methods to identify a causal ordering and the parent set of each variable.

In this paper, we restrict our attention to non-invertible causal relations between variables in a DAG, which has not been explored in the literature. In the bivariate case, we develop a novel method to identify the causal direction by test for non-invertibility of f. This is less general than the above methods that apply to many nonlinear functions, however, our approach can be more powerful for the problem we consider and does not rely on complicated independence tests. Moreover, we assume that the causal relations in a DAG are a mix of linear and nonlinear relationships. Accordingly, we propose a few approaches to combine traditional structure learning methods with our non-invertible function identification in a principled way to estimate the causal DAG structure. Our numerical comparisons show that our combined approach outperforms both the traditional structure learning methods and the recent nonlinear DAG learning methods.

Our causal learning method is widely applicable to many data domains. First, causal structure estimation under the DAG framework has become popular in different applied fields, including genomics (Sachs et al., 2005; Gao and Cui, 2015), epidemiology (Greenland et al., 1999; Joffe et al., 2012) and social sciences (Velikova et al., 2014; Garvey et al., 2015). Second, identification of nonlinear and non-invertible causal relations by our method will bring new insights into the underlying scientific problem. Most graphical model approaches to large-scale problems work under linearity assumptions, which serve as a good approximation if the underlying nonlinear relationship is monotone and close to a linear function. These methods are usually not sensitive enough to identify non-invertible causal relations. Our method fills this gap. A non-invertible relationship can review complicated causality among the variables of interest. Use gene regulation as an example. The expression of a gene is often regulated by the binding of multiple proteins, called transcription factors (TFs), to the upstream sequence of the gene. The presence or absence of one TF X may cause a change of the binding of another TF Y. The causality of the binding activities among a set of TFs may be reviewed by learning a DAG from their binding data, in particular, ChIP-Seq data. There could be nonlinear relations in this problem, which reflects the complexity in combinatorial gene regulation. We will apply our method to ChIP-Seq data to demonstrate its use in scientific discovery.

The remainder of this paper is organized as follows. We start with introducing our bivariate non-invertible SEM and test of causal direction in Section 2. We then incorporate this method into structure learning of causal networks with both linear and nonlinear SEMs in Section 3. Section 4 evaluates the performance of the proposed algorithms under different simulation settings and compares with other competing DAG learning methods. Section 5 presents an application to ChIP-Seq data for the construction of a TF binding causal network. The paper concludes with discussions in Section 6. In the supplementary material, we provide some technical details of our algorithms and additional numerical results.

2. Non-invertible bivariate causal relations

2.1. Bivariate non-invertible SEM

Consider two random variables X and Y that may be causally related. Our task here is to decide whether there is indeed a causal relation between the two variables and if so whether the relation is $X \to Y$ (i.e. X causes Y) or $Y \to X$. We assume only observational data are available. To make the causal relation identifiable from observational data, we will consider a non-invertible SEM between X and Y defined as follows.

Definition 2.1. Suppose two random variables X and Y satisfy a nonlinear SEM, $Y = f(X) + \epsilon$, where ϵ is independent of X and the function f is non-trivial (i.e. not a constant function) and non-invertible (i.e. $f^{-1}(\cdot)$ does not exist). Then we say that X and Y follow a bivariate non-invertible SEM (NISEM), which defines the causal relation $X \to Y$.

It follows immediately from the identifiability of nonlinear SEMs (Zhang and Hyvärinen, 2009, Corollary 10) that a bivariate NISEM is identifiable. Besides identifiability, non-invertible functions capture many important nonlinear causal relations in real world. In many biological or chemical interactions, the effect of X on Y may change significantly, say from positive effect to negative, beyond some threshold. For example, a drug usually shows a positive treatment effect over a certain range of dosage, but then could become harmful. Another example of nonlinear function is the sigmoid function which exists in many natural processes. The positive effect starts when $X > x_1$, and reaches a saturate point at x_2. In addition, from a practical point of view, if f is invertible, a linear function is usually sufficient to model the causal relation. If f is non-invertible, our method provides a simple yet quite powerful way for detection.

For now, we assume that either $X \to Y$ or $Y \to X$, or they are not causally related so that X is independent of Y. To infer the causal relation between X and Y, we consider three scenarios accordingly:

1. X and Y are independent. We infer that there is no causal relationship between X and Y.
2. X and Y are dependent, but the function f is invertible. In this case there exists causality between X and Y, but we are not able to identify the direction of the relation.
3. X and Y are dependent, and the function f is non-invertible. We conclude that there exists causal relationship between X and Y and we are able to determine the causal direction.

Given observational data, we develop a method to determine which of the above three cases is supported by the data. Our main method is a two-step approach: First, test whether X and Y are statistically independent. If the two variables are not independent, we then continue to fit a bivariate nonlinear SEM for (X, Y) and test whether the function f is invertible. Although the general identifiability results in Peters et al. (2014) apply to a large class of nonlinear functions, including invertible functions, our primary focus in this work is on the non-invertible cases. By limiting to non-invertible functions, our method gains substantial increase in power and accuracy, as demonstrated in our numerical comparisons in Section 4.2.

Many non-invertible functions can be well approximated by piecewise linear functions. Here, we use a piecewise linear function with two pieces to approximate the functional relation between X and Y. That is, we assume

$$f(x) = \begin{cases} a_l + b_l x & x \le \tau_x \\ a_h + b_h x & x > \tau_x \end{cases},$$

where τ_x is the cut point between the two pieces of linear functions and a_l, b_l, a_h, b_h are coefficients of the linear functions.

Our motivation to use a piecewise linear function stems from the fact that nonlinear relationships can generally be approximated using a sufficient number of pieces. A non-invertible function by definition is not monotonic and a piecewise linear function can easily capture the nonlinear trend with a well-chosen cutoff point. For example, a quadratic function can be approximated by two pieces of linear functions, each having a very different slope. For nonlinear relationships with multiple tuning points, it is more accurate to use multiple pieces of linear functions. However, determining the number of pieces and fitting a many-piece model can be inaccurate and may increase the risk of overfitting in practice. Fortunately, simply capturing two linear pieces with a significant change in their slopes is sufficient for our purpose of detecting the causal direction in a non-invertible relationship. See Fig. 1 for an illustration. Simulation results on different nonlinear patterns in Section 4.2 confirm the robustness of our approach. On the other hand, it is possible to generalize our method to allow multiple linear pieces with a change point detection procedure (e.g. Pettitt, 1979; Reeves et al., 2007; Chen and Gupta, 2014), which will be left for future work.

2.2. Model fitting

We will first discuss how to fit a piecewise linear function from data and then propose a statistic to measure the goodness of fit, which will be used in our determination of the causal direction in next subsection.

Suppose we have observed data $(\mathbf{x}, \mathbf{y}) = \{(x_i, y_i) : i = 1, \ldots, n\}$, an i.i.d. sample from the joint distribution of (X, Y). Assuming the causal direction is $X \to Y$, a corresponding bivariate nonlinear SEM is estimated in the following way. First we find the cut point τ_x of the piecewise function. We restrict the domain of τ_x to be a set of quantiles of $\mathbf{x}$, denoted by

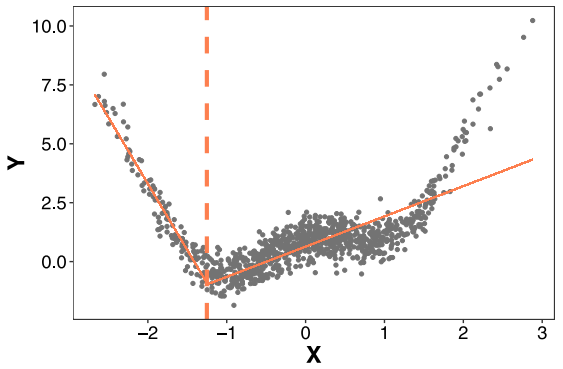


Fig. 1. Approximating a nonlinear relationship by piecewise linear function with two pieces. The cut point is indicated by the vertical dashed line.

$T = \{t_j, j = 1, \ldots, m\}$. For each $t_j \in T$, fit a piecewise linear function, which yields two residual sums of squares $rss_l(t_j)$ for $x \le t_j$ and $rss_h(t_j)$ for $x > t_j$. Then the estimate of τ_x is found by minimizing the total residual sum of squares of the two segments:

$$\hat{\tau}_x = \arg\min_{t \in T} \{rss_l(t) + rss_h(t)\}. \tag{1}$$

Second, given $\hat{\tau}_x$ we fit a linear function in each segment. Write the estimated f as

$$\hat{f}(x) = \begin{cases} \hat{a}_l + \hat{b}_l x & x \le \hat{\tau}_x \\ \hat{a}_h + \hat{b}_h x & x > \hat{\tau}_x \end{cases}, \tag{2}$$

where $\hat{\tau}_x$ is the estimated cut point, and $\hat{a}_l$, $\hat{b}_l$, $\hat{a}_h$, $\hat{b}_h$ are the estimated coefficients for the two linear segments. In this work, the cardinality of T is set to $\min(n/20, 100)$.

To measure the goodness of fit of the piecewise model, we define a statistic $\bar{R}^2$, which is a weighted average of the R^2 of each piece:

$$\bar{R}^2_{X \to Y} = \frac{n_l r_l^2 + n_h r_h^2}{n_l + n_h}, \tag{3}$$

where n_l and n_h are the number of observations in the subsets $\{i : x_i \le \hat{\tau}_x\}$ and $\{i : x_i > \hat{\tau}_x\}$, respectively; r_l, r_h are the corresponding sample correlation coefficients between x_i and y_i.

To test whether f is non-invertible, we will swap x and y in the above procedure to fit a nonlinear SEM for $Y \to X$ and evaluate the model fitting by calculating $\bar{R}^2_{Y \to X}$. Then we will design a test to decide the causal direction between X and Y based on $\bar{R}^2_{X \to Y}$ and $\bar{R}^2_{Y \to X}$.

2.3. Test for causal direction

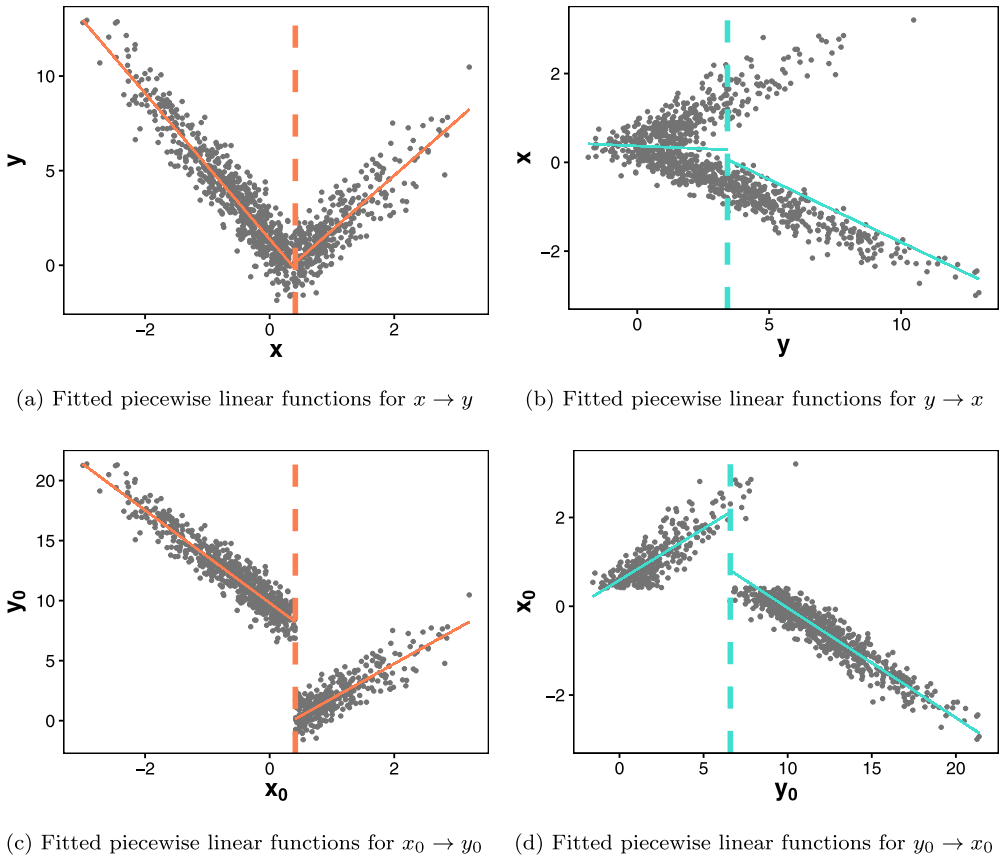
Although one direction may be preferred than the other based on the model fitting statistics, we need to find out whether this preference is statistically significant. Thus, hypothesis testing is necessary to decide whether the function f is indeed non-invertible. Our null hypothesis H_0 is that the functional relationship f between X and Y is invertible.

We define a test statistic

$$\eta = \max\left\{\bar{R}^2_{X \to Y} / \bar{R}^2_{Y \to X}, \bar{R}^2_{Y \to X} / \bar{R}^2_{X \to Y}\right\}, \tag{4}$$

to compare the goodness of fit between the two nonlinear SEMs. Under H_0 that f is invertible, the two SEMs would fit the data equally well so that the values of $\bar{R}^2$ for the two nonlinear SEMs will be close to each other. Therefore, the corresponding η should be close to 1. Under H_a that f is non-invertible, the values of $\bar{R}^2$ for the two nonlinear SEMs will be significantly different and η will be significantly greater than 1. Let η_0 be a random variable following the distribution of η under H_0, and $\hat{\eta}$ be the observed value of the comparison statistic η. The p-value of the hypothesis test is $P(\eta_0 \ge \hat{\eta} | H_0)$. Now the question is how to obtain the distribution of η_0. We propose two different methods to approximate this null distribution.

The first method is based on the bootstrap, a commonly used technique for constructing null distributions by random resampling with replacement. From the observed data $(\mathbf{x}, \mathbf{y})$, we first find the preferred nonlinear SEM, i.e. the one with a greater $\bar{R}^2$ statistic, and its estimated piecewise function $\hat{f}$ and the associated parameters. For an example data set $(\mathbf{x}, \mathbf{y})$ shown in Fig. 2a, the preferred model is $x \to y$ and the fitted function is represented by two red solid lines. The $\bar{R}^2_{X \to Y} = 0.834$ for this direction. On the contrary, it is obvious from Fig. 2b that the alternative model $y \to x$ does not



(a) Fitted piecewise linear functions for $x \to y$ (b) Fitted piecewise linear functions for $y \to x$

(c) Fitted piecewise linear functions for $x_0 \to y_0$ (d) Fitted piecewise linear functions for $y_0 \to x_0$

Fig. 2. Illustration of test for causal direction. Data in (a) and (b) are observed. Data in (c) and (d) are modified data for bootstrap. Solid lines are the fitted piecewise linear functions; dashed lines indicate the cut points found in the two variables. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

fit the data well, for which $\bar{R}^2_{Y\to X} = 0.085$. Thus, the observed test statistic $\hat{\eta} = 9.77$ for this example. Next, we modify our data according to the null hypothesis before resampling. This is best illustrated with the above example data set. We divide the data into two segments by the estimated cut point $\hat{\tau}_x$, indicated by the red dashed line in Fig. 2a. Then, we move one segment of the data points up or down along the y-axis by a minimum distance such that the two fitted line segments do not overlap in the range of the data. As $\hat{f}(x)$ after this modification becomes invertible, this leads to a modified data set $(\mathbf{x}_0, \mathbf{y}_0)$ that satisfies the null hypothesis. As confirmed in Figs. 2c and 2d, the modified data $(\mathbf{x}_0, \mathbf{y}_0)$ can be fitted with an invertible nonlinear function, and the model fitting is comparable between the two directions $x_0 \to y_0$ in panel 2c and $y_0 \to x_0$ in panel 2d.

After generating the modified data $(\mathbf{x}_0, \mathbf{y}_0)$, our bootstrap test procedure works as follows:

1. Sample the null data $(\mathbf{x}_0, \mathbf{y}_0)$ with replacement to generate a bootstrap sample $(\mathbf{x}_0^b, \mathbf{y}_0^b)$;
2. For a bootstrap sample $(\mathbf{x}_0^b, \mathbf{y}_0^b)$, fit two piecewise linear functions as in (2), one for each of the two directions;
3. Calculate the comparison statistic η_0^b between the two directions using Eq. (4);
4. Repeat the first three steps for $b = 1, \ldots, B$ to generate the bootstrap null distribution of $\{\eta_0^b\}$, based on which we calculate the p-value of $\hat{\eta}$.

Fig. 3 shows the bootstrap distribution of η_0 from the example in Fig. 2. It ranges from 1 to 1.2, while the observed $\hat{\eta} = 9.77$ is way much larger, indicating that f is not invertible as shown in Fig. 2.

The bootstrap method can be computationally intensive, especially for approximating small p-values. Therefore, we develop a more efficient alternative method to approximate the null distribution of the test statistic η and calculate the p-values. We know from the definition of η that it is a function of Pearson's correlation coefficients. It is well-known that Pearson's correlation coefficient after Fisher transformation follows approximately a normal distribution when the sample size n is large. Thus, the distribution of η_0 can be obtained by sampling Pearson's correlation coefficient from this approximate distribution. The details are presented below.

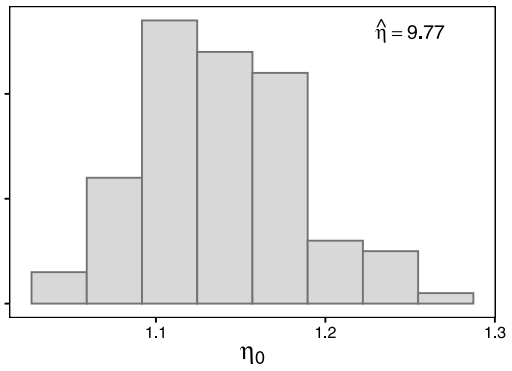


Fig. 3. Null distribution of η_0 estimated by the histogram of $\{\eta_0^b\}$.

Let r be the Pearson's correlation coefficient between $\mathbf{x}$ and $\mathbf{y}$. Fisher's z-transformation of r is

$$z = \frac{1}{2}\log\frac{1+r}{1-r} = \operatorname{arctanh}(r).$$

If $(\mathbf{x}, \mathbf{y})$ is an i.i.d. sample from a bivariate normal distribution with true correlation ρ, then z is approximately normally distributed as $\mathcal{N}(\operatorname{arctanh}(\rho), 1/(n-3))$, where n is the sample size. Assuming that each segment of the null data $(\mathbf{x}_0, \mathbf{y}_0)$ follows a bivariate normal distribution, then the inverse of the transformation $r = \tanh(z)$ can be used to construct the distribution of η_0. Given the null data, we first estimate the optimal cut point in each direction (the red and blue dashed lines in Fig. 2(c)). For the direction $x_0 \to y_0$, we separate the null data into two subsets: $(\mathbf{x}_0, \mathbf{y}_0)_l$ and $(\mathbf{x}_0, \mathbf{y}_0)_h$ according to the estimated cut point $\hat{\tau}_{x_0}$ of $\mathbf{x}_0$. For each subset of data, we compute its correlation coefficient, denoted by ρ_l, ρ_h respectively. Now sample z_l, z_h from

$$z_l \sim \mathcal{N}\big(\operatorname{arctanh}(\rho_l), 1/(n_l-3)\big), \quad z_h \sim \mathcal{N}\big(\operatorname{arctanh}(\rho_h), 1/(n_h-3)\big),$$

where n_l, n_h are the sample sizes of $(\mathbf{x}_0, \mathbf{y}_0)_l$ and $(\mathbf{x}_0, \mathbf{y}_0)_h$, respectively. Substituting Pearson's correlation coefficient r with $\tanh(z)$ in the formula of $\bar{R}^2$ in Eq. (3), we get

$$\bar{R}^2_{x_0 \to y_0} = [n_l \tanh(z_l)^2 + n_h \tanh(z_h)^2]/(n_l + n_h).$$

Similarly, we can draw $\bar{R}^2_{y_0 \to x_0}$ using the same procedure, and obtain a large sample of η_0 to approximate the null distribution.

Simulation was performed to validate p-value calculated by the bootstrap and the normal approximation procedures under the null hypothesis. We generated 100 data sets each with $n = 1000$ observations under invertible SEMs, and used the above two procedures to calculate the p-value for each data set. Fig. 4 shows the quantile–quantile plots of these p-values against Unif(0, 1). The bootstrap p-values are approximately uniformly distributed between (0, 1) while the p-values calculated via normal approximation seem to be a little left-skewed compared to the uniform distribution. Accordingly, at a significance level of 0.05, the rejection rate was controlled at 0.05 for the bootstrap test, while the normal approximation p-values were more conservative with a rejection rate around 0.02. Note that for both tests, the type-I error was controlled at or below the desired level of 0.05. We also repeated this simulation with smaller sample sizes $n \in \{50, 100, 500\}$. The rejection rate did not change that much and was around 0.03 across different sample sizes.

Suppose $y = f(x)$ is indeed non-invertible. When the true cutting point is close to either boundary of the domain of x, the nonlinear pattern will not be significantly different from a linear approximation and consequently our test is expected to have a lower power. To confirm this, we did a simulation with the true τ_x being the 1%-quantile. For this extreme setting, the power of our test increased to 70% when the sample size $n \geq 800$. On the contrary, when the true cutting point was not too close to either boundary (between 5% and 95% percentiles), our test had power $\geq 70\%$ for a quite small sample size $n = 100$.

2.4. Algorithm for bivariate case

We summarize below our Algorithm 1 for inferring the causal relation between two variables from observed data $(\mathbf{x}, \mathbf{y})$. This algorithm will serve as a unit in our structure learning of causal networks in Section 3. Thus, we represent its output as an edge between the two variables X and Y, regarded as two nodes in a graph. There are four possible outcomes: $E(X, Y) \in \{\emptyset, X \to Y, Y \to X, X - Y\}$. The case $E(X, Y) = \emptyset$ means that the two variables are independent (no edge between the two nodes). An undirected edge $X - Y$ indicates that the SEM is invertible and the causal direction cannot be decided. A directed edge will be output if the test in the previous section is rejected at the significance level α.

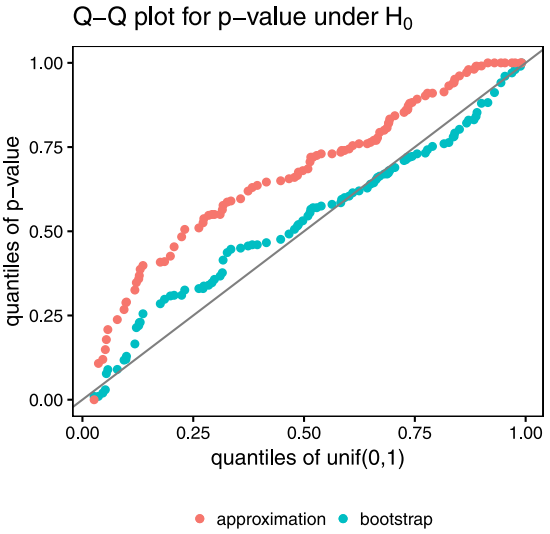


Fig. 4. The Q-Q plot of p-value distributions under null against Unif(0,1).

Algorithm 1: Bivariate non-invertible causal discovery algorithm

Input: Observed data (X, Y), significance level α

Initialize $E(X, Y) = \emptyset$

Independence test between X and Y

if $X \not\perp\!\!\!\perp Y$ **then**

Fit a bivariate nonlinear SEM for each direction as described in Sections 2.1 and 2.2

Calculate corresponding goodness of fit statistics $\bar{R}^2_{X\to Y}, \bar{R}^2_{Y\to X}$

Preferred edge direction

$$E(X, Y) := \begin{cases} X \to Y & \text{if } \bar{R}^2_{X\to Y} \geq \bar{R}^2_{Y\to X} \\ Y \to X & \text{if } \bar{R}^2_{X\to Y} < \bar{R}^2_{Y\to X} \end{cases}$$

Calculate p-value of the causal direction test (Section 2.3)

If $p > \alpha$, set $E(X, Y) := X - Y$

end if

Output: $E(X, Y)$

3. Nonlinear causal structure learning

In this section, we incorporate non-invertible causal discovery into structure learning of a causal network among p variables, $X_1, \ldots, X_p$. The generative distribution for these p random variables is given by a set of SEMs whose structures are defined by an underlying directed acyclic graph (DAG), which will be called the causal DAG or causal graph. We allow both nonlinear and linear causal relations in the model. Our proposed method combines existing structure learning methods, such as the PC algorithm and regularized likelihood methods (Fu and Zhou, 2013), with our non-invertible causal discovery approach (Algorithm 1).

Before a detailed description of our new method, we give a quick review of causal DAGs and general SEMs in Section 3.1.

3.1. Causal DAGs and SEMs

Pearl (2009) and Spirtes (2010) pioneered the use of DAGs in causal modeling and inference. A causal DAG $\mathcal{G}$ on a set of variables $X_1, \ldots, X_p$ encodes assumptions about the data-generating process and is a great tool for visualization of causal relations among these variables. There is a directed edge $X_i \to X_j$ if and only if X_i is a direct cause of X_j, in which case we say X_i is a (causal) parent of X_j. Let $\mathbf{PA}_i$ denote the set of parents of X_i. The joint generative distribution $\mathbb{P}$ over the variables $\mathbf{V} = \{X_1, \ldots, X_p\}$ modeled by the causal DAG $\mathcal{G}$ is specified by a set of SEMs

$$X_i = f_i(\mathbf{PA}_i, \epsilon_i), \quad i = 1, \ldots, p, \tag{5}$$

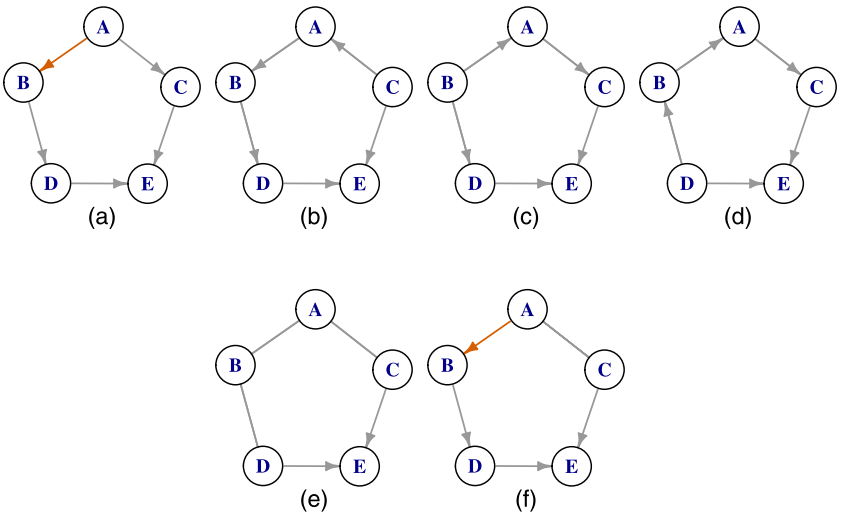


Fig. 5. DAGs (a) to (d) in an equivalence class represented by the CPDAG (e), and the restricted CPDAG (f) subject to a non-invertible edge (the red edge). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where ϵ_i is a background or noise variable independent of $\mathbf{PA}_i$. Moreover, all background variables ϵ_j, $j = 1, \ldots, p$ are mutually independent so that the joint distribution $\mathbb{P}$ satisfies Markov properties with respect to the causal DAG $\mathcal{G}$.

We make the following assumptions on the above causal DAG model:

1. **Causal sufficiency:** The set of variables $\mathbf{V}$ is *causally sufficient*. That is, there is no variable not in $\mathbf{V}$ that is a direct cause of more than one variable in $\mathbf{V}$ (Spirtes, 2010). In other words, all common causes of variables in the DAG are included in the set $\mathbf{V}$ of measured variables.
2. **Faithfulness:** Every conditional independence relation implied by the joint distribution $\mathbb{P}$ over $\mathbf{V}$ is entailed by d-separation in the causal DAG (Spirtes, 2010).

These are common assumptions in structure learning of DAGs (Spirtes and Glymour, 1991; Chickering, 2003; Aragam and Zhou, 2015).

3.2. Restricted equivalence class

In this work, we assume that there are both linear and nonlinear causal relations in the SEMs (5) under an additive model framework:

$$X_i = \sum_{j \in \mathbf{PA}_i^l} \beta_{ji} X_j + \sum_{k \in \mathbf{PA}_i^n} f_{ki}(X_k) + \epsilon_i, \quad i = 1, \ldots, p, \tag{6}$$

where f_{ki} is a nonlinear function. We call $\mathbf{PA}_i^l$ and $\mathbf{PA}_i^n$ the linear and nonlinear parent sets of X_i, and accordingly, call an edge $j \to i$ a linear and nonlinear edge, respectively, for $j \in \mathbf{PA}_i^l$ and $j \in \mathbf{PA}_i^n$.

Note that the linear and nonlinear parents are learned by our method without any prior information. The linear parents are detected by a linear structure learning algorithm, while the nonlinear parents are detected by our nonlinear learning algorithms. The detailed procedures will be discussed in Section 3.3.

When $\mathbf{PA}_i^n = \emptyset$ in Eq. (6), we have the regular linear SEMs. Linear SEMs with Gaussian errors are not identifiable due to the so-called Markov equivalence class of DAGs, which is a set of DAGs encoding the same set of conditional independence relations. Two DAGs are Markov equivalent if and only if they have the same skeleton and the same v-structures (Verma and Pearl, 1990). Here, the skeleton of a DAG is the underlying undirected graph obtained by ignoring the direction of every edge, and a v-structure is an ordered triplet of nodes (i, j, k) of the form $i \to k \leftarrow j$, where i, j are not connected by an edge. A Markov equivalence class can be uniquely represented by a completed partially DAG (CPDAG), which contains both directed and undirected edges. As illustrated in Fig. 5, DAGs (a)–(d) have the same skeleton and the same v-structure $C \to E \leftarrow D$, and they constitute all the DAGs in the Markov equivalence class, which is represented by the corresponding CPDAG (e).

If some of the undirected edges in a CPDAG can be oriented, say by non-invertible relations in our problem, then the equivalence class will be reduced. One can apply Meek's rules (Meek, 1995) to orient other undirected edges and obtain a maximally oriented partially DAG (PDAG) that represents a restricted equivalence class. We call this maximally oriented PDAG a *restricted CPDAG*, which serves as the ground-truth for our structure learning. Suppose the edge $A \to B$ of the

DAG (a) in Fig. 5 is non-invertible and thus not reversible. Keeping the orientation of this edge in the CPDAG (e), we then maximally orient the rest of the undirected edges, which leads to the orientation of $B \to D$ since $D \to B$ would introduce an extra v-structure with the non-reversible edge $A \to B$. Thus we obtain the restricted CPDAG (f) for this example. In general, a restricted CPDAG, subject to a set of non-reversible edges, represents the subset of DAGs in the Markov equivalence class that have the same orientations for those non-reversible edges. In Fig. 5, the restricted equivalence class includes DAGs (a) and (b), out of the four DAGs in the Markov equivalence class.

3.3. Structure learning algorithms

Our goal is to infer the causal DAG, with both linear and nonlinear edges, from observational data. The overall idea of our approach is to combine an existing linear structure learning algorithm to identify a CPDAG from the data. Then we recursively apply the bivariate non-invertible causal discovery algorithm (Algorithm 1) in Section 2.4 to detect any non-invertible relation and orient more edges. Since linear structure learning algorithms may output a DAG, a CPDAG or a PDAG, we in general assume the output is a PDAG which includes CPDAGs as a special case. Note that a DAG learned by linear structure learning will be converted to a CPDAG before applying our algorithm. Given a PDAG, we first develop a non-invertible nonlinear causal learning (NNCL) algorithm that generalizes the bivariate algorithm described in Section 2.4 to multiple variables. Then we discuss a few approaches that combine a linear structure learning algorithm with the NNCL algorithm to identify a causal graph with both linear and nonlinear edges.

We distinguish directed and undirected neighbors in a PDAG as follows. If there is a directed edge $j \to i$ in a PDAG, we say j is a parent of i and if they are linked by an undirected edge $i - j$, they are called a neighbor of each other. We define $\mathbf{PA} = \{\mathbf{PA}_i, i = 1, \ldots, p\}$ as the collection of all parent sets and $\mathbf{U} = \{(X_i, X_j) : i - j \in E\}$ as the set of all undirected edges in a PDAG $\mathcal{G} = (\mathbf{V}, E)$, where $\mathbf{V}$ is the node set and E is the edge set.

Based on an input initial PDAG $\mathcal{G}$, our NNCL algorithm recursively detects the most significant non-invertible edge among all undirected ones, then fixes the orientation of this edge in the graph and applies the orientation rules in Meek (1995) to orient the remaining undirected edges. A non-invertible edge between X_i and X_j is detected by reducing to the bivariate case (Algorithm 1) after calculating the residuals after projecting each of them to its respective identified parents. An outline of our algorithm is shown in Algorithm 2.

Algorithm 2: Non-invertible nonlinear causal learning (NNCL)

Input: observed data for $(X_1, \ldots, X_p)$, initial PDAG $\mathcal{G} = (\mathbf{V}, E)$, significance level α

1: **repeat**
2: **for** $(X_i, X_j) \in \mathbf{U}$ **do**
3: apply Algorithm 1 Line 4-7 on the residuals of X_i, X_j after regressing on their respective parents $\mathbf{PA}_i$, $\mathbf{PA}_j$
4: **obtain:** test-statistic $\eta_{i,j}$, p-value $p_{i,j}$, and the preferred direction $E_{i,j}$
5: **end for**
6: Sort $\mathbf{U}$ by $p_{i,j}$ *, and let $E_{(1)} = (V_p \to V_c)$ be the edge with minimum p-value $p_{(1)}$.
7: **if**

(i) $p_{(1)} \le \alpha$
(ii) $V_p \not\perp\!\!\!\perp V_c | \mathbf{PA}_{V_c}$
(iii) adding $E_{(1)}$ to $\mathcal{G}$ does not induce any directed cycle

then
8: add $E_{(1)}$ to E
9: orient other undirected edges in $\mathcal{G}$ by Meek's rules
10: update $\{\mathbf{U}, \mathbf{PA}\}$
11: **end if**
12: **until** no more edge can be added

Output: Restricted CPDAG $\mathcal{G}$

$*$ when there is a tie, sort by $\eta_{i,j}$

The initial residuals in Line 3 are calculated from regressing X_i and X_j on their respective linear parents in the initial PDAG $\mathcal{G}$. Then in the following steps, every time a nonlinear parent is added to the structure, the residuals of the child node will be updated to the residuals calculated from the fitted piecewise function. Note that Line 7(ii) is used in place of the independence test (Line 2) in Algorithm 1. For a preferred edge $V_p \to V_c$, where $V_p, V_c \in \mathbf{V}$, we first divide the data according to the cut point $\hat{\tau}$ of V_p estimated in the piecewise linear function, and then perform conditional independence test for each segment of the data. We require both reject the null hypothesis in order to conclude that $V_p \not\perp\!\!\!\perp V_c | \mathbf{PA}_{V_c}$. This procedure takes into account the nonlinear relationship between V_p and V_c. See supplementary material S1 for the details on the conditional independence tests in our procedure.

The initial PDAG in Algorithm 2 can be estimated using an existing structure learning algorithm that produces a CPDAG from observational data. However, the initial graph may fail to detect the dependency among variables in a nonlinear relationship, thus missing nonlinear edges in the estimated skeleton. Therefore, we implement the following algorithm to search outside the skeleton of the initial PDAG after Algorithm 2 is done.

Algorithm 3: Searching nonlinear edges outside the initial skeleton

Input: observed data for $(X_1, \ldots, X_p)$, $\mathcal{G}$ output from Algorithm 2, significance level α

Define the set of non-adjacent pairs in $\mathcal{G}$ as **NE**
1: **for** $(X_i, X_j) \in$ **NE do**
2: apply Algorithm 1 Line 4-7 on the residuals of X_i, X_j after regressing on their respective parents $\mathbf{PA}_i$, $\mathbf{PA}_j$
3: **obtain:** p-value $p_{i,j}$, and the preferred direction $E_{i,j} = V_p \to V_c$
4: **if**

(i) $p_{i,j} \leq \alpha$
(ii) $V_p \not\perp\!\!\!\perp V_c | \mathbf{PA}_{V_c}$
(iii) adding $E_{i,j}$ to $\mathcal{G}$ does not induce any directed cycle

then
5: add $E_{i,j}$ to E
6: orient other undirected edges in $\mathcal{G}$ by Meek's rules
7: **end if**
8: **end for**
Output: Restricted CPDAG $\mathcal{G}$

In our implementation, we use two structure learning algorithms to construct the initial estimate of a CPDAG: the order-independent PC algorithm (Colombo and Maathuis, 2014) and the CCDr algorithm (Aragam and Zhou, 2015). The PC algorithm is a constraint-based method that learns a graphical structure by repeated conditional independence (CI) tests. The main procedure of this method is to first estimate a skeleton using CI tests, and then identify v-structures in the skeleton. Finally it applies the orientation rules in Meek (1995) to direct the remaining edges without introducing new conditional independence relations or directed cycles. The PC algorithm we use is implemented in the *bnlearn* package (Scutari, 2010), and the details of the algorithm can be found in Colombo and Maathuis (2014). One may use PC algorithm for non-Gaussian or non-linear dependencies by incorporating more general CI tests, such as the kernel-based CI test (Zhang et al., 2011). However, since our goal is to identify linear edges by the PC algorithm, we will use Gaussian CI tests in the numerical experiments. The CCDr algorithm is a score-based method that maximizes a regularized Gaussian likelihood under a concave penalty function. This algorithm is available in the R package *sparsebn* (Aragam et al., 2019) with algorithm details described in Aragam and Zhou (2015). The CCDr algorithm outputs a DAG, which we convert to a CPDAG by the function *cpdag* in the package *bnlearn*. We call these two implementations PC-NNCL and CCDr-NNCL, respectively.

We discuss briefly the intuition behind our algorithms. Assume that (1) the input initial PDAG $\mathcal{G}$ in Algorithm 2 is the CPDAG of the true DAG, and (2) there exists an undirected edge between X_i and X_j that is non-invertible. Algorithm 2 will first obtain residuals by regressing X_i and X_j on their respective identified parents $\mathbf{PA}_i$ and $\mathbf{PA}_j$. Since we assume that the parent effects are additive, the residuals will preserve a non-invertible relation, which reduces the problem to the bivariate case. Thus, we will be able to direct this non-invertible edge using Algorithm 2 when the sample size becomes large. After that, a repeated application of Meek's orientation rules (Line 9) will maximally orient the graph and recover the restricted CPDAG. It is possible that the initial PDAG $\mathcal{G}$ may not contain certain non-invertible edges. The additional search procedure in Algorithm 3 is designed to detect such missing edges.

4. Numerical experiments

In this section, we report numerical experiments on simulated data to verify the validity and demonstrate the effectiveness of our non-invertible nonlinear causal network learning method. We evaluate three different versions of our method: PC-NNCL and CCDr-NNCL, discussed above, and NNCL (Algorithm 3 with empty initial graph), so that we can see the usefulness of combining linear structure learning (PC and CCDr) with our nonlinear edge detection. We also compare them with PC and CCDr in Section 4.1, and another recent method for nonlinear DAG learning in Section 4.2.

4.1. Simulation study

We selected six different DAGs with graph size ranging from small to large from the Bayesian network repository (http://www.bnlearn.com/bnrepository/). We did not use the conditional distributions or edge weights associated with

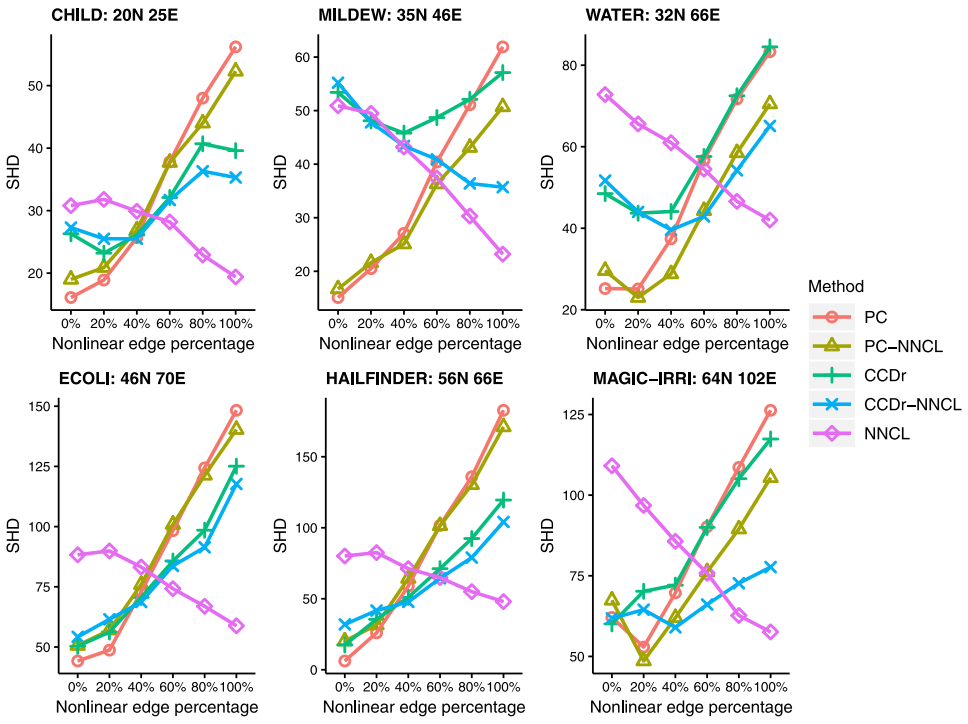


Fig. 6. SHD comparison among five algorithms on six networks. (N: number of nodes, E: number of edges).

these networks but only their DAG structures. For each network we simulated data with different percentages of nonlinear edges: 0%, 20%, 40%, 60%, 80%, 100%. Under each setting we generated 10 simulated data sets with sample size $n = 1,000$. The linear edges were generated using linear functions with random coefficients, while nonlinear edges were generated using different quadratic functions for the comparison in this section. For comparisons on more general nonlinear relations, see the simulation study reported in Section 4.2. The detailed simulation procedure is described in Supplementary Material, Section S2.

The ground-truth we compare against is the true restricted CPDAG. We used the function *cpdag* in R package *bnlearn* to transform the true DAG to its restricted CPDAG by specifying a white list of non-invertible edges in the DAG.

The Structural Hamming Distance (SHD) and Jaccard Index (JI) are used to evaluate the performance of the algorithms. The SHD measures the difference between the estimated graphs and the true graph. It is defined as the number of edge additions, deletions or orientation corrections in order to match two PDAGs. Here, orientation corrections include reversal of a directed edge and a change from a directed edge to an undirected one and vice versa. Thus, a lower SHD indicates a better performance. JI measures the similarity between two graphs. It is the percentage of correct edges among the union of edges in the estimated graph and the true graph. The higher the percentage is, the closer the two graphs are.

In the following simulation results, the test for causal direction was performed using the normal approximation approach with a p-value threshold of 0.01. The significance level of the conditional independence tests in Algorithm 2 and 3 was set at 0.01. For conditional independence tests in the PC algorithm, we used the predefined *gaussCItest* with significance level 0.01. The default settings were used for running the CCDr algorithm.

Figs. 6 and 7 show the SHD and JI comparisons among the five methods: PC, PC-NNCL, CCDr, CCDr-NNCL, and NNCL. The six panels in each figure report the results for the six networks from the Bayesian network repository. The colored curves correspond to different algorithms and are plotted against the percentage of nonlinear edges in the true DAG.

PC and CCDr showed higher accuracy in general when there were less than 40% nonlinear edges, except that CCDr algorithm did not perform as well as PC on the Mildew data sets. As we increased the percentage of nonlinear edges, there was a dramatic decrease in the accuracies of PC and CCDr estimates, reflected by both metrics. This demonstrates the difficulty of these baseline algorithms in learning nonlinear DAGs. The NNCL algorithm exhibited an opposite trend, having higher accuracy for DAGs with more nonlinear edges. The SHDs of NNCL estimates were generally the smallest when there were more than 60% of nonlinear edges. Linear-NNCL algorithms (i.e. PC-NNCL and CCDr-NNCL) showed great improvement over PC and CCDr, and the improvement became more substantial in settings with a higher percentage of nonlinear edges. The performance curves of the two linear-NNCL algorithms had a similar trend with the corresponding linear structure learning algorithms, because the nonlinear edge estimation was based on the initial graphs estimated by the linear algorithms. We observe moderate decrease in SHDs and significant increase in JI after the NNCL step, showing that more edges were correctly identified in the NNCL step. Overall, linear-NNCL algorithms showed the best performance

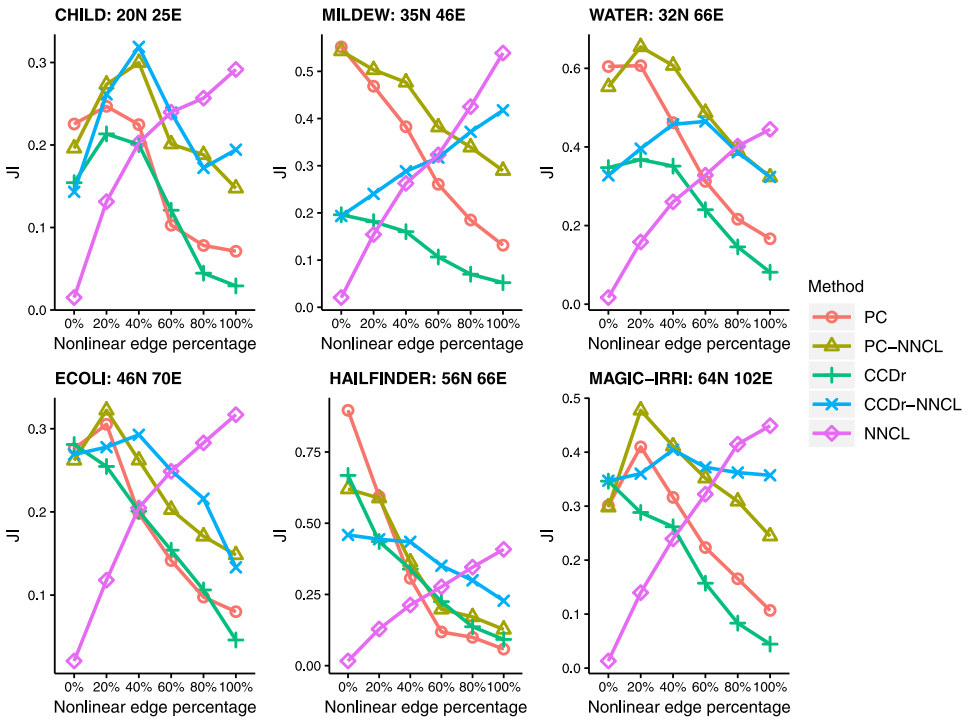


Fig. 7. JI comparison among five algorithms on six networks. (N: number of nodes, E: number of edges).

for a wide range of nonlinear edge percentages. They became inferior to NNCL only when the true DAG was mostly composed of nonlinear edges ($\geq$ 80%).

Besides the above overall accuracy metrics, we also report in the supplementary material the numbers of true positive (TP) and false positive (FP) edges in this comparison (Figure S3-1, S3-2)). True positive curves were similar to what we observed with the JI curves above. Adding NNCL step increased the TPs by 15.2% on average compared to the PC algorithm, and 20.8% on average compared to the CCDr algorithm. The FP curves of linear-NNCL algorithms were close to those of the linear algorithms. Note that the NNCL step orients undirected edges and does not delete any edges in the initial CPDAG estimated by linear structure learning. Therefore, a linear-NNCL algorithm will not decrease the FPs compared to its linear counterpart. Moreover, as seen in Figure S3-2, the NNCL algorithm showed a low number of FPs overall and had the lowest FPs among all the algorithms when there were more than 40% nonlinear edges. This indicates that our nonlinear edge detection does not lead to overfitting.

An alternative approach to learning a causal DAG with nonlinear edges is to first exhaustively search for nonlinear edges among all pairs of nodes by running Algorithm 1 repeatedly. Then we apply a linear structure learning algorithm with the detected nonlinear edges fixed. We call this approach NNCL-linear and present the results in the supplementary material. The curves of JI and TP (Figure S3-4 and S3-5) show that, in general, adding the linear step after NNCL helped improve the detection of true positive edges, especially in settings with a low percentage of nonlinear edges. In Figure S3-3, we observe moderate decrease in SHDs of the NNCL-linear algorithms when there were less than 60% nonlinear edges. However, we also observe a quite significant increase in SHDs of these algorithms comparing to NNCL when we increased the percentage of nonlinear edges to more than 60%. This is mainly due to the substantial increase in the FP edges in the linear step, especially for PC (see Figure S3-6). These observations suggest that when the DAG consists of mostly nonlinear edges, adding the linear step would be of little use. Overall, we found linear-NNCL algorithms more accurate and will stick to this approach in the following results.

4.2. Comparison with RESIT

Next, we compare our approach with a recent nonlinear causal learning algorithm called regression with subsequent independence test (RESIT) proposed by Peters et al. (2014). RESIT was developed based on additive noise models (ANM) (Hoyer et al., 2009), which is identifiable from observational data. The algorithm consists of two phases. The first phase yields a topological ordering by iteratively identifying and removing a sink node. In each step of this iterative procedure, each of the remaining variables is regressed on all the other remaining variables and the dependence between residuals and the other variables is measured. The variable with the least dependence is identified as a sink node and

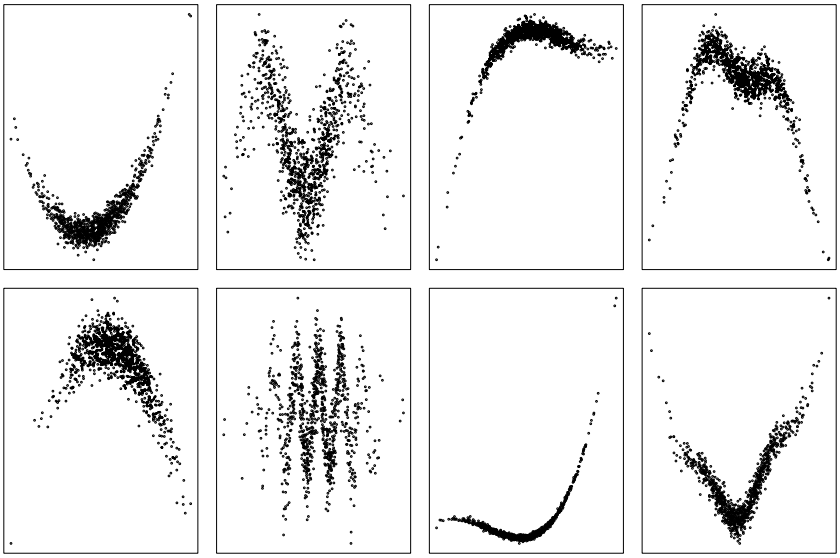


Fig. 8. Examples of various nonlinear patterns in simulated data.

removed. In the second phase, given the estimated ordering, superfluous edges are removed by further conditional independence tests. The RESIT algorithm was implemented in R and the code was obtained from the author's website. More details of the algorithm can be found in Peters et al. (2014).

Since RESIT does not handle a large number of nodes effectively and generally takes a long time to run, we compared our methods with RESIT on a small network *Asia* from the Bayesian network repository that has 8 nodes and 8 edges. Similarly, we simulated data sets with different percentages of nonlinear edges: 0%, 20%, 40%, 60%, 80%, 100%. Four types of nonlinear relationships listed in Section S2 were simulated for empirical performance evaluation and comparison. Fig. 8 shows examples of different nonlinear patterns in the simulation. Each column corresponds to one type of nonlinear functions, and the two rows were simulated with randomly chosen parameters.

The regression method in RESIT can be selected from linear regression, generalized additive model (gam) and Gaussian process regression. Here we ran RESIT with gam and the default HSIC independence test for dependence measure. We compared the two linear-NNCL algorithms and NNCL algorithm with RESIT. The p-value cutoff was set to 0.001 for all three of our algorithms.

Figs. 9 and 10 show the performances of the four algorithms in terms of SHD and JI under different simulation settings. The ground truth we compared our results against was the true restricted CPDAG. Since RESIT always outputs a DAG, the performance of RESIT was compared to the true DAG instead. True positive (TP) and false positive (FP) comparisons are provided in supplementary materials (Figure S3-7, S3-8).

We observe from the plots that in general linear-NNCL algorithms performed similarly and showed the best results across all four cases of nonlinear relations. In particular, both PC-NNCL and CCDr-NNCL outperformed RESIT substantially for data sets with $\leq$ 60% of nonlinear edges and showed comparable accuracy with RESIT for cases with $\geq$ 80% nonlinear edges. When there were fewer nonlinear edges, PC-NNCL algorithm (red lines) had lower SHDs and higher JI, which indicate better performance, while CCDr-NNCL algorithm (green lines) performed better in settings with a higher percentage of nonlinear edges. The NNCL algorithm (blue lines) showed lower accuracy when there were fewer nonlinear edges, but its performance improved greatly as the nonlinear percentage increased. The performance curves of RESIT (purple lines) exhibited similar trend as NNCL. RESIT was able to identify more correct edges than NNCL, however, at a cost of more superfluous edges (observed from the false positive curves) which led to a higher SHD between true and estimated graphs.

The above results also confirm that the proposed NNCL algorithms were able to handle different types of nonlinear data. For instance, the nonlinear patterns in the second and fourth columns in Fig. 8 are obviously composed of multiple segments, yet our method had no problem detecting such non-invertible relationships using two-piece approximations as in Eq. (2). Fig. 11 illustrates the detection of such complex non-invertible relationships by a two-piece linear model. The red dashed line in the figure is the estimated cut point of X_1 being the parent (i.e. $X_1 \to X_2$), and the blue dashed line is the estimated cut point of X_2 being the parent (i.e. $X_2 \to X_1$). The solid lines represent the fitted two-piece linear functions. We observe from Fig. 11a that the two pieces of functions captured the significant change in the nonlinear pattern. However, model fitting in the other direction $X_2 \to X_1$ (Fig. 11b) failed to do so and resulted in a much smaller goodness of fit statistic $\bar{R}^2$. Therefore, a simple two-piece linear model allows us to identify more complex non-invertible relationships by capturing a single significant change in the pattern. Of course, there are drawbacks using this simple

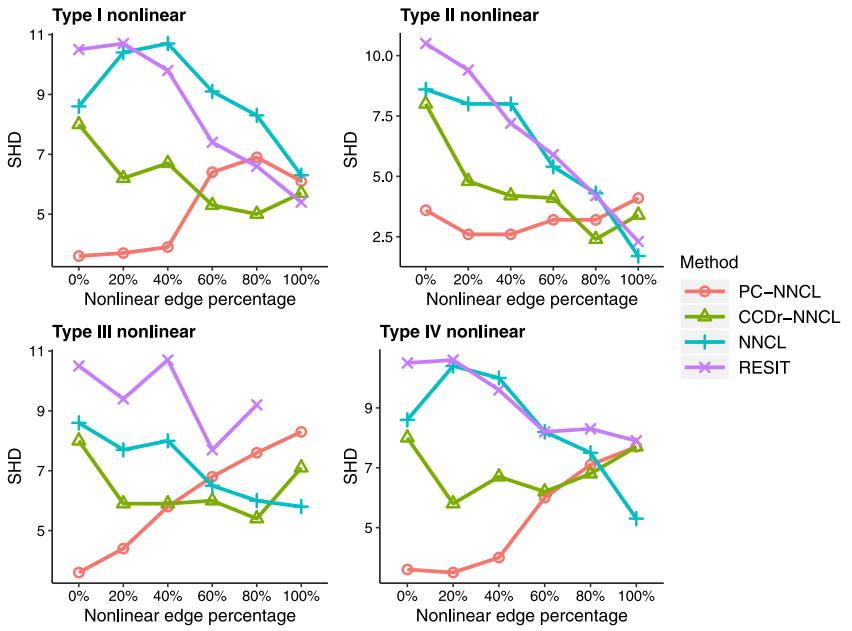


Fig. 9. SHD comparison among four algorithms on different types of nonlinear models (Note: RESIT result missing for type III with 100% nonlinear edges due to an error in their code).

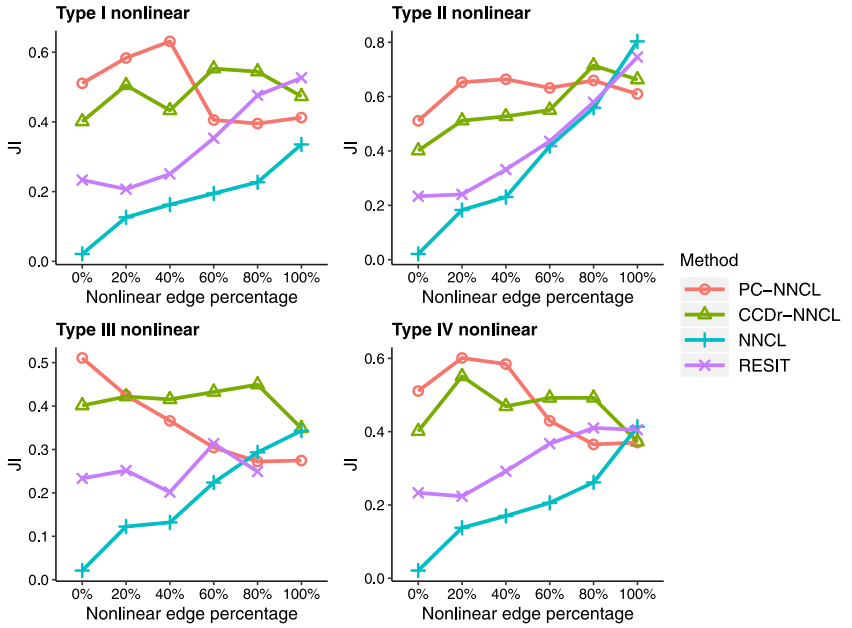
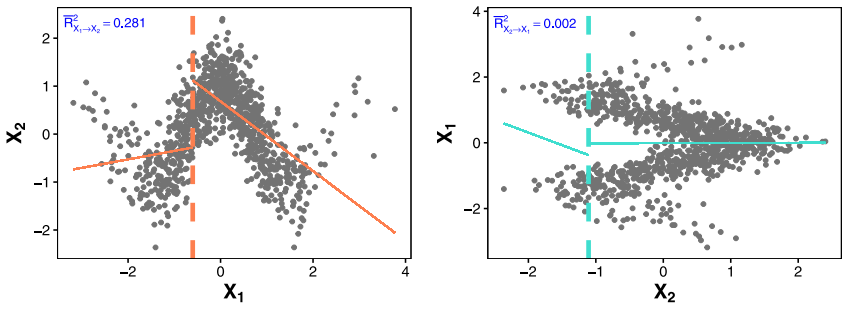


Fig. 10. JI comparison among four algorithms on different types of nonlinear models (Note: RESIT result missing for type III with 100% nonlinear edges due to an error in their code).

procedure. Although we are able to successfully detect a non-invertible edge, the residuals obtained from the two pieces of linear models would be inaccurate and could affect the following detection if there are other undirected edges between X_2 and its neighbors. In such cases, a multiple-piece or more general nonlinear model fitting procedure is expected to be more powerful.

We also compared the computing time among the four algorithms in Table 1 on the data sets in this subsection. All the methods were run on a MacBook Pro with 2 GHz dual-core. One sees that RESIT was quite time consuming for even a small network. The average computing time of RESIT was almost three times that of the other algorithms.



(a) Model fitting for $X_1 \to X_2$ (b) Model fitting for $X_2 \to X_1$

Fig. 11. Example of complex non-invertible functions detected by our method with two-piece linear approximation.

Table 1
Comparison on computing time.

Method	PC-NNCL	CCDr-NNCL	NNCL	RESIT
Computing time in seconds	10.059	11.944	12.699	35.043

5. Application to ChIP-Seq data

Although linear SEMs are commonly used in learning causal network structures, real-world data rarely satisfy a perfectly linear causal relationship. Therefore, assuming nonlinearity will help identify causal relationships from data. For example, some causal relations in biological data are expected to be nonlinear, exhibiting a piecewise trend. A gene X may regulate gene Y with a nonlinear functional relationship. When the expression level of X is low, X may have no effect on the expression of Y; but if the expression level X passes certain threshold, it shows a strong positive regulation on Y. Similarly, the binding of transcription factors (TFs) to DNA may also show nonlinear causality. Transcription factors are a class of proteins that bind DNA in order to activate or suppress a downstream gene. The binding of one TF may stimulate the binding of another TF under a nonlinear dependence. Thus, it is an interesting and important problem to identify the causal network among the bindings of multiple TFs that work together in gene regulation.

In this section, we apply our methods to the ChIP-Seq data generated by Chen et al. (2008). The data set contains the DNA binding sites of 12 transcription factors in mouse embryonic stem cells: *Smad1, Stat3, Sox2, Pou5f1, Nanog, Esrrb, Tcfcp2l1, Klf4, Zfx, E2f1, Myc,* and *Mycn*. For each transcription factor, an association strength score, which is the weighted sum of the corresponding ChIP-Seq signal strength, was calculated for each of the 18,936 genes (Ouyang et al., 2009). Roughly speaking, this score can be understood as a measure of the binding strength of a TF to a gene. The genes with zero association scores were removed from our analysis. Accordingly, our observed data matrix, of size $n \times p = 8462 \times 12$, contains the association scores of 12 TFs over 8,462 genes. We aim to build a causal network that reveals how these 12 TFs might affect each other's binding to genes.

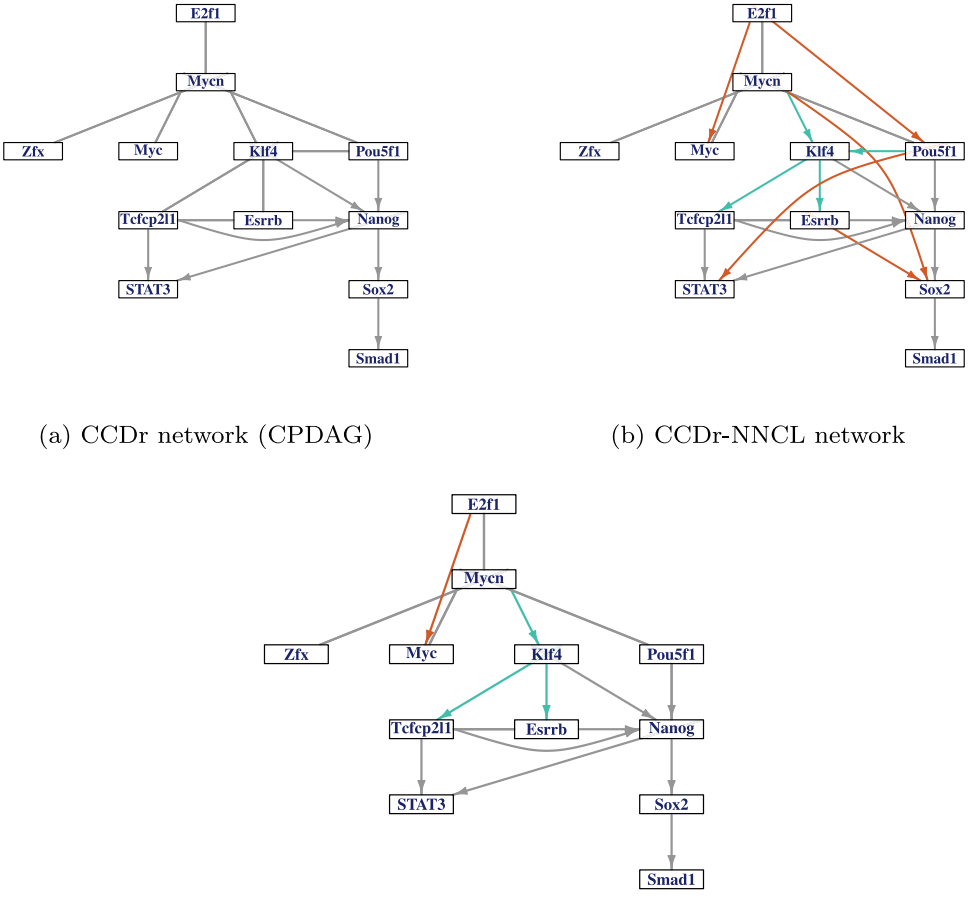
Since there is no ground-truth for comparison, ten-fold cross validation was used to evaluate the performance of our methods. We first split the data into training and test sets, and ran a network learning method to obtain an estimated graph and associated parameters from training data. Then given an estimated network structure and the parameters, we calculated the likelihood of the test data set. Since the estimated graph was a PDAG, we extended the PDAG to an arbitrary DAG in the restricted equivalence class without creating any directed cycle or additional v-structures, and then used this DAG for estimating model parameters from training data and calculating test data likelihood. For simplicity, we postulated a quadratic function for each identified nonlinear edge, i.e. $f_{ki}(x) = a_k x + b_k x^2$ in Eq. (6), so that parameter estimation can be done by least-squares. The likelihood of test data was evaluated based on Gaussian error distributions. Note that the real causal relations among the variables in this data are unknown and could be any nonlinear functions.

From the simulation results in previous section, we find that CCDr-NNCL tends to have the best overall performance in different nonlinear settings. Therefore CCDr-based algorithms were chosen for this data analysis. The significance levels of the hypothesis tests and conditional independence tests in the NNCL steps were all set to 0.001. Table 2 reports the results for CCDr and CCDr-NNCL averaging over 10 folds of cross validations. We see that the NNCL steps indeed identified on average 6.3 nonlinear edges and increased test data likelihoods compared to CCDr.

The networks learned on the full data set are shown in Fig. 12. Fig. 12a is the CPDAG of CCDr estimated network and Fig. 12b is the network estimated by CCDr-NNCL. The green edges are directed among the reversible edges in the structure, and the red ones are nonlinear edges detected outside the skeleton using Algorithm 3. In the CCDr-NNCL network, we conducted 58 causal direction tests in total, each at significance level 0.001, and six of them were rejected. Another three undirected edges were later oriented using Meek's rules. Therefore, the expected false discovery rate was around 0.01 for our nonlinear edge detection in this problem.

Table 2
Ten-fold cross validation results on ChIP-Seq data.

Method	CCDr	CCDr-NNCL
Average test data log-likelihood	−12 081.1	−11 901.0
Average number of edges	19.0	22.1
Average number of nonlinear edges	NA	6.3



(a) CCDr network (CPDAG)

(b) CCDr-NNCL network

(c) Consensus network

Fig. 12. TF binding causal networks estimated from ChIP-Seq data. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

To improve the stability of our estimated graph, a consensus network was constructed via bootstrap. Let $\mathcal{G}$ denote the estimated graph (PDAG) in Fig. 12b by CCDr-NNCL and λ be the tuning parameter used in the CCDr algorithm. First, we resampled the full data set with replacement 100 times, and ran CCDr-NNCL with the same CCDr tuning parameter λ on each bootstrap sample to obtain 100 estimated graphs $\{\mathcal{G}'_1, \ldots, \mathcal{G}'_{100}\}$. Second, we calculated a weighted adjacency matrix $W = (w_{ij})_{p\times p}$, where each entry of the adjacency matrix w_{ij} recorded the percentage of the edge $i \to j$ appeared in the 100 estimated graphs $\{\mathcal{G}'_b, b = 1, \ldots, 100\}$. Finally, we constructed a consensus network using the weight matrix by the following rules. A directed edge $i \to j$ in $\mathcal{G}$ was kept if the weight $w_{ij} \geq 0.6$; a directed edge $i \to j$ in $\mathcal{G}$ was kept but changed to an undirected edge $i - j$ if $w_{ij} < 0.6$ and $w_{ij} + w_{ji} \geq 0.6$; a directed edge $i \to j$ was deleted if neither of the above two conditions were satisfied. An undirected edge $i - j$ in $\mathcal{G}$ was kept if $w_{ij} + w_{ji} \geq 0.6$ and was deleted otherwise. The graph in Fig. 12c is the consensus network so constructed with the same color code in Fig. 12b.

It is well-known that two or more TFs may cooperate to regulate target genes. The work in Ouyang et al. (2009) suggests that *E2f1, Myc, Mycn, Zfx* form one group of TFs (group I) that work together, and *Pou5f1, Nanog, Sox2, Smad1, Stat3, Tcfcp2l1, Esrrb* form another group (II). We observe from the consensus network in Fig. 12c that the group I TFs are more closely connected, and similarly group II TFs are also closely connected, consistent with their findings. *Mycn* appears

to be a point of junction in group I, and *Nanog* seems to be an important connecting point in group II. Four nonlinear edges (green edges) were discovered from the CCDr skeleton and three of them were preserved in the consensus network. Only one edge out of the five edges detected outside the CCDr skeleton was preserved in the consensus network. Obviously the nonlinear edges detected within the skeleton were more stable. The results provide clues for nonlinear causal relations among TF binding events. Such pairs of TFs include *E2f1* $\rightarrow$ *Myc*, *Mycn* $\rightarrow$ *Klf4*, *Klf4* $\rightarrow$ *Tcfcp2l1* and *Klf4* $\rightarrow$ *Esrrb*, in which *Klf4* appears to interact with other TFs mostly in a nonlinear way. It would be interesting to further study the regulation roles of these TFs that showed nonlinear interactions. Another observation from the consensus network is that *Mycn* and *Pou5f2* are the root causes of the binding of all group II TFs, while *Stat3* and *Smad1*, both in group II, are identified as sink nodes in all three estimated graphs.

6. Discussion

Causal discovery from observational data is a crucial step to understanding causality in real world applications, especially when experiments are limited or infeasible. In this paper, we have demonstrated that non-invertible causal relationships can be identified from observational data. We started from the bivariate case, where the task was to decide the cause between two variables, and designed a test-based procedure to determine the causal direction. Furthermore, we extended the work to multivariate case and proposed an efficient algorithm which incorporates both linear structure learning and non-invertible SEMs to estimate the structure of a causal DAG.

We have tested and applied our methods on both simulated and real-world data sets. The simulation results indicate that by applying our NNCL algorithm, we can identify the causal directions of nonlinear edges with non-invertible relationships, and thus further reduce the Markov equivalence class estimated by traditional constraint-based or score-based DAG learning methods. Extensive numerical comparisons show that our linear-NNCL algorithms are able to handle different types of nonlinear relationships and outperform the RESIT algorithm in most cases. The application to ChIP-Seq data highlights the utility of incorporating nonlinear SEMs in learning causal networks.

Several topics will be studied in future work. The two-piece linear model will lead to a loss of accuracy when fitting more complex nonlinear causal relations. For example, using residuals from the two-piece model may result in false negatives in non-invertibility tests. Generalization of our model from two pieces to multiple pieces can help improve model fitting and edge detection of our algorithms for more complicated data. The hypothesis test for causal direction we proposed in this paper is based on sample correlation coefficients. Other possible statistics await to be explored in the future. Finally, more theoretical work can be developed to study the large-sample properties of our methods.

Acknowledgment

This work was supported by US NSF grants IIS-1546098 and DMS-1952929.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.csda.2020.107141. Technical details and supplementary figures.

References

Aragam, Bryon, Gu, Jiaying, Zhou, Qing, 2019. Learning large-scale Bayesian networks with the sparsebn package. J. Stat. Softw. 91 (11), 1–38. http://dx.doi.org/10.18637/jss.v091.i11.

Aragam, Bryon, Zhou, Qing, 2015. Concave penalized estimation of sparse Gaussian Bayesian networks. J. Mach. Learn. Res. 16 (69), 2273–2328.

Blöbaum, Patrick, Janzing, Dominik, Washio, Takashi, Shimizu, Shohei, Schölkopf, Bernhard, 2018. Cause-effect inference by comparing regression errors. In: Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics. In: Proceedings of Machine Learning Research, vol. 84, pp. 900–909.

Chen, Jie, Gupta, Arjun K., 2014. Parametric Statistical Change Point Analysis: With Applications to Genetics, Medicine, and Finance. Birkhauser Boston, ISBN: 0817648003, http://dx.doi.org/10.1007/978-0-8176-4801-5.

Chen, Xi, Xu, Han, Yuan, Ping, Fang, Fang, Huss, Mikael, Vega, Vinsensius, Wong, Eleanor, Orlov, Yuriy, Zhang, Weiwei, Jiang, Jianming, Loh, Yuin-Han, Yeo, Hock, Yeo, Zhen, Narang, Vipin, Govindarajan, Kunde, Leong, Bernard, Shahab, Atif, Ruan, Yijun, Bourque, Guillaume, Ng, Huck-Hui, 2008. Integration of external signaling pathways with the core transcriptional network in embryonic stem cells. Cell 133, 1106–1117.

Chickering, David Maxwell, 1996. Learning equivalence classes of Bayesian-network structures. In: Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence. pp. 150–157.

Chickering, David Maxwell, 2003. Optimal structure identification with greedy search. J. Mach. Learn. Res. (ISSN: 1532-4435) 3, 507–554.

Colombo, Diego, Maathuis, Marloes H., 2014. Order-independent constraint-based causal structure learning. J. Mach. Learn. Res. 15, 3921–3962.

Fu, Fei, Zhou, Qing, 2013. Learning sparse causal Gaussian networks with experimental intervention: Regularization and coordinate descent. J. Amer. Statist. Assoc. 108 (501), 288–300.

Gao, Bin, Cui, Yuehua, 2015. Learning directed acyclic graphical structures with genetical genomics data. Bioinformatics (Oxf. Engl.) 31, http://dx.doi.org/10.1093/bioinformatics/btv513.

Garvey, Myles D., Carnovale, Steven, Yeniyurt, Sengun, Garvey, M.D., Carnovale, S., Yeniyurt, S., 2015. An analytical framework for supply network risk propagation: A Bayesian network approach. European J. Oper. Res. 243 (2), 618–627. http://dx.doi.org/10.1016/j.ejor.2014.10.034.

Ghoshal, Asish, Honorio, Jean, 2018. Learning linear structural equation models in polynomial time and sample complexity. In: Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics. In: Proceedings of Machine Learning Research, vol. 84, PMLR, pp. 1466–1475.

Glymour, Clark, Zhang, Kun, Spirtes, Peter, 2019. Review of causal discovery methods based on graphical models. Front. Genet. (ISSN: 1664-8021) 10, 524.
Greenland, Sander, Pearl, Judea, Robins, James M., 1999. Causal diagrams for epidemiologic research. Epidemiology (ISSN: 10443983) 10 (1), 37–48.
Gretton, Arthur, Herbrich, Ralf, Smola, Alexander, Bousquet, Olivier, Scholkopf, Bernhard, 2005. Kernel methods for measuring independence. J. Mach. Learn. Res. 6, 2075–2129.
Heckerman, David, Geiger, Dan, Chickering, David, 1995. Learning Bayesian networks: The combination of knowledge and statistical data. Mach. Learn. 20, 197–243. http://dx.doi.org/10.1007/BF00994016.
Heckerman, David, Meek, Chris, Cooper, Gregory, 2006. A Bayesian approach to causal discovery. In: Computation, Causation, and Discovery, Computation, Causation, and Discovery ed. AAAI Press, ISBN: 978-3-540-30609-2, pp. 141–166.
Hoyer, Patrik O., Janzing, Dominik, Mooij, Joris M, Peters, Jonas, Schölkopf, Bernhard, 2009. Nonlinear causal discovery with additive noise models. In: Advances in Neural Information Processing Systems 21. Curran Associates, Inc., pp. 689–696.
Joffe, Michael, Gambhir, Manoj, Chadeau-Hyam, Marc, Vineis, Paolo, 2012. Causal diagrams in systems epidemiology. Emerg. Themes Epidemiol. 9, 1. http://dx.doi.org/10.1186/1742-7622-9-1.
Meek, Christopher, 1995. Causal inference and causal explanation with background knowledge. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. pp. 403–410.
Monti, Ricardo P., Zhang, Kun, Hyvärinen, Aapo, 2019. Causal Discovery with general non-linear relationships using non-linear ICA. In: Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019, Conference on Uncertainty in Artificial Intelligence.
Mooij, Joris M., Peters, Jonas, Janzing, Dominik, Zscheischler, Jakob, Schölkopf, Bernhard, 2016. Distinguishing cause from effect using observational data: Methods and benchmarks. J. Mach. Learn. Res. 17 (32), 1–102.
Mooij, Joris M., Stegle, Oliver, Janzing, Dominik, Zhang, Kun, Schölkopf, Bernhard, 2010. Probabilistic latent variable models for distinguishing between cause and effect. In: Advances in Neural Information Processing Systems 23. pp. 1687–1695.
Ouyang, Zhengqing, Zhou, Qing, Wong, Wing, 2009. ChIP-Seq of transcription factors predicts absolute and differential gene expression in embryonic stem cells. Proc. Natl. Acad. Sci. USA 106, 21521–21526.
Pearl, Judea, 2009. Causality. Cambridge University Press.
Peters, Jonas, Bühlmann, Peter, 2014. Identifiability of Gaussian structural equation models with equal error variances. Biometrika (ISSN: 00063444) 101 (1), 219–228.
Peters, Jonas, Bühlmann, Peter, Meinshausen, Nicolai, 2016. Causal inference by using invariant prediction: identification and confidence intervals. J. R. Stat. Soc. Ser. B Stat. Methodol. 78 (5), 947–1012.
Peters, Jonas, Mooij, Joris M., Janzing, Dominik, Schölkopf, Bernhard, 2014. Causal discovery with continuous additive noise models. J. Mach. Learn. Res. (ISSN: 1532-4435) 15 (1), 2009–2053.
Pettitt, A.N., 1979. A non-parametric approach to the change-point problem. J. R. Stat. Soc. C (ISSN: 00359254) 28 (2), 126–135, 14679876.
Reeves, Jaxk, Chen, Jien, Wang, Xiaolan L., Lund, Robert, Lu, Qi Qi, 2007. A review and comparison of changepoint detection techniques for climate data. J. Appl. Meteorol. Climatol. 46 (6), 900–915.
Sachs, Karen, Perez, Omar, Pe'er, Dana, Lauffenburger, Douglas A., Nolan, Garry P., 2005. Causal protein-signaling networks derived from multiparameter single-cell data. Science (ISSN: 0036-8075) 308 (5721), 523–529. http://dx.doi.org/10.1126/science.1105809.
Scutari, Marco, 2010. Learning Bayesian networks with the bnlearn R package. J. Stat. Softw. Artic. (ISSN: 1548-7660) 35 (3), 1–22.
Shimizu, Shohei, Hoyer, Patrik O., Hyvärinen, Aapo, Kerminen, Antti, 2006. A linear non-Gaussian acyclic model for causal discovery. J. Mach. Learn. Res. 7, 2003–2030.
Shimizu, Shohei, Inazumi, Takanori, Sogawa, Yasuhiro, Hyvärinen, Aapo, Kawahara, Yoshinobu, Washio, Takashi, Hoyer, Patrik O., Bollen, Kenneth, 2011. Directlingam: A direct method for learning a linear non-Gaussian structural equation model. J. Mach. Learn. Res. 12, 1225–1248.
Spirtes, Peter, 2010. Introduction to causal inference. J. Mach. Learn. Res. (ISSN: 1532-4435) 11, 1643–1662.
Spirtes, Peter, Glymour, Clark, 1991. An algorithm for fast recovery of sparse causal graphs. Soc. Sci. Comput. Rev. 9, 62–72.
Spirtes, Peter, Glymour, Clark, Scheines, Richard, 2000. Causation, Prediction, and Search, second ed. MIT press.
Velikova, Marina, van Scheltinga, Josien Terwisscha, Lucas, Peter J.F., Spaanderman, Marc, 2014. Exploiting causal functional relationships in Bayesian network modelling for personalised healthcare. Internat. J. Approx. Reason. (ISSN: 0888-613X) 55 (1, Part 1), 59–73. http://dx.doi.org/10.1016/j.ijar.2013.03.016.
Verma, Thomas, Pearl, Judea, 1990. Equivalence and synthesis of causal models. In: Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence. In: UAI '90, Elsevier Science Inc., pp. 255–270.
Zhang, Kun, Hyvärinen, Aapo, 2008. Distinguishing causes from effects using nonlinear acyclic causal models. In: Proceedings of the 2008th International Conference on Causality: Objectives and Assessment - Volume 6. pp. 157–164.
Zhang, Kun, Hyvärinen, Aapo, 2009. On the identifiability of the post-nonlinear causal model. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. pp. 647–655.
Zhang, Kun, Peters, Jonas, Janzing, Dominik, Schölkopf, Bernhard, 2011. Kernel-based conditional independence test and application in causal discovery. In: Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence. In: UAI'11, AUAI Press, ISBN: 9780974903972, pp. 804–813.