

Causal Inference

Table of contents

- 1 Introduction
- 2 Theoretical overview
 - Introduction
 - Identifiability I
 - SCM graph
 - D -separation
 - Faithfulness
 - Markov property
 - Identifiability II
- 3 Methods
 - Conditional independence-based methods
 - Score-based methods
- 4 Inference
 - Do-calculus
 - Counterfactuals

Introduction

Given a dataset with observations on some variables, we might get interested in the causal structure of the data, i.e. which variables affect which, what the impact of the cause onto the effect is, and so on.

Problem I

Say we have some observations on variables $X_1, \dots, X_d$.
We need to describe the causal structure for the variables, i.e. we should understand what variables are causes and what are effects for a certain variable.

Problem II

Say we have a causal structure for variables $X_1, \dots, X_d$.
We need to measure the impact of each cause on the effect.

Trying to solve these problems we immediately face a few fundamental questions.

What does cause even mean?

How do we define causal structure?

How do we measure the impact of cause on effect?

Once defined at what extent can we be sure about the correctness of the causal structure?

...

Causal Inference is a theory within mathematical statistics developed in order to formulate the causality and related problems

Theoretical overview. Introduction

Let's start with only two variables – X and Y .

In real life we are given a number of observations $(x_1, y_1), \dots, (x_n, y_n)$, so let us assume some joint probability distribution $P_{X,Y}$.

It is well known that $P_{X,Y}$ admits the following representations

$$\mathbf{P}_{X,Y} = \mathbf{P}_X * \mathbf{P}_{Y|X}, \quad (1)$$

$$\mathbf{P}_{X,Y} = \mathbf{P}_Y * \mathbf{P}_{X|Y}. \quad (2)$$

If we imply a causal structure for X and Y it is natural to associate representation (1) with the statement “ X causes Y ”, and (2) with “ Y causes X ”.

Furthermore, (1) and (2) bring the following systems

$$\begin{cases} X = \xi_1 \\ Y = F_{Y|X}^{-1}(\xi_2), \end{cases} \quad \begin{cases} X = F_{X|Y}^{-1}(\eta_2) \\ Y = \eta_1 \end{cases},$$

where ξ_1 and η_1 are random variables with distributions P_X and P_Y respectively, ξ_2 and η_2 are random variables distributed uniformly on $[0,1]$, $\xi_1 \perp \xi_2$, $\eta_1 \perp \eta_2$.

The consideration above leads us to define the causal structure as a system

$$\begin{cases} X = \xi_1 \\ Y = f(X, \xi_2), \end{cases}$$

where f is a measurable function that characterizes causal link from X to Y (X is the *parent* of Y) and ξ_1, ξ_2 are some independent random variables.

This system is called Structural Causal Model, or SCM.

Theoretical overview. Introduction

Let's generalize SCM for the case of finitely many variables.

Suppose we have variables $X_1, \dots, X_d$, and each variable, X_j , is associated with a set of some other variables, $\mathcal{P}_j$, called *parents*.

Structural Causal Model, or **SCM**, is a system of equations

$$X_j = f_j(\mathcal{P}_j, \xi_j), \quad j \in \overline{\{1, d\}},$$

where f_j is a measurable function, $\mathcal{P}_j$ is a subset of $\{X_k\} \setminus X_j$ and $\{\xi_k\}$ are jointly independent random variables.

The idea of SCM is that each variable depends on its parents and some random effect.

Some examples of SCMs:

$$\begin{cases} X_1 = \xi_1 \\ X_2 = \xi_2 \\ X_3 = 2X_1 + 3X_2 + \xi_3 \\ X_4 = \xi_4 * X_3 \end{cases}$$

$$\begin{cases} X_1 = \xi_1 \\ X_2 = \sin(11X_1) \\ X_3 = 5e^{X_1} \end{cases}$$

$$\begin{cases} X_1 = \xi_1 \\ X_2 = 13X_1 + \xi_2 \\ X_3 = 17\xi_3 * X_2 \\ X_4 = X_3^{\xi_4} \\ X_5 = \cos(X_1 + X_2 + X_4) \end{cases}$$

Theoretical overview. Identifiability I

As we have seen in the two-variables case, $\mathbf{P}_{X_1, \dots, X_d}$ doesn't uniquely determine SCM, instead many SCMs correspond to $\mathbf{P}_{X_1, \dots, X_d}$. In fact, number of SCMs corresponding to $\mathbf{P}_{X_1, \dots, X_d}$ increases super-exponentially as d increases.

The notion above means that given a number of observations on several variables, it is impossible to know their causal structure certainly.

How should we deal with that fact?

Let us again consider the two-variables cause, X and Y . So then the problem would be to determine Cause and Effect. In terms of SCM we need to come up with the model

$$\begin{cases} \text{Cause} = \xi_1 \\ \text{Effect} = f(\text{Cause}, \xi_2). \end{cases} \quad (1)$$

Since both X and Y can be the Cause, we need to make some assumptions about $\mathbf{P}_{\text{Effect}|\text{Cause}}$, $\mathbf{P}_{\text{Cause}}$, f or ξ_1 and ξ_2 . If our assumptions are “good” enough, we will be able to distinguish between Cause and Effect so that only X , or only Y , can be the Cause in the model (1).

Given a model and its assumptions, the problem of identifying whether one is able to surely distinguish between Causes and Effect is called **identifiability** problem.

Theoretical overview. Identifiability I

Linear non-Gaussian acyclic model,
or LiNGAM

Consider the model

$$\begin{cases} X = \xi_1 \\ Y = \alpha X + \xi_2, \end{cases} \quad (1)$$

where $\alpha \in \mathbb{R}$, $\xi_1 \perp \xi_2$.

Theorem ([1, th. 4.2])

There exist $\beta \in \mathbb{R}$, η_1, η_2 such that
 $\eta_1 \perp \eta_2$ and
$$\begin{cases} X = \beta Y + \eta_2 \\ Y = \eta_1 \end{cases},$$

if and only if ξ_1 and ξ_2 are Gaussian.

The theorem means that if at least one of random variables ξ_1, ξ_2 is non-Gaussian, the model (1) is identifiable, i.e. one is able distinguish between Cause and Effect.

Additive noise model, or ANM

Consider the model

$$\begin{cases} X = \xi_1 \\ Y = f(X) + \xi_2, \end{cases} \quad (2)$$

where f is measurable, $\xi_1 \perp \xi_2$.

There is a theorem [2] that states that in a very wide range of cases, "typically", that SCM is identifiable.

Theorem

Let p_{ξ_1}, p_{ξ_2} be strictly positive and f, p_{ξ_1}, p_{ξ_2} be three times differentiable. Then, if there exists a $y \in \mathbb{R}$, such that $(\log p_{\xi_2})''(x)(y - f(x))f'(x) \neq 0$ a.e., the set of log densities $\log p_{\xi_1}$ for which the model (2) is not identifiable is a three-parametric family of special functions.

The theorem means that if p_{ξ_1} is not of a very special kind of functions, then the model (2) is identifiable.

Post-nonlinear model, or PNLM

Consider the model

$$\begin{cases} X = \xi_1 \\ Y = g(f(X) + \xi_2), \end{cases} \quad (3)$$

where f, g are measurable, $\xi_1 \perp \xi_2$.

Theorem

Assume p_{ξ_1}, f, g are all three times differentiable. The model (3) is identifiable if and only if p_{ξ_1}, f, g are not connected to each other in the sense of a differential equation described in [3].

Theoretical overview. SCM graph

The identifiability results in many-variables case are much easier to formulate using graph terminology, which is very useful itself. So now we will adopt that terminology.

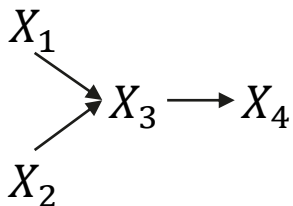
The graph G , or **SCM graph** G , corresponding to a SCM

$$X_j = f_j(\mathcal{P}_j, \xi_j), \quad j \in \overline{\{1, n\}},$$

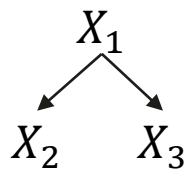
is a directed acyclic graph, or DAG, with vertices $\{X_k\}$ and some edges such that there is an edge from X_i to X_j if and only if X_i is in $\mathcal{P}_j$

Some examples of SCMs and its' graphs:

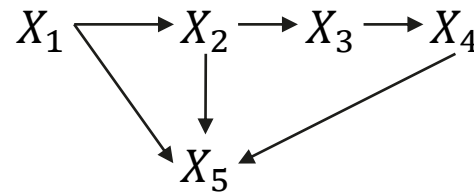
$$\begin{cases} X_1 = \xi_1 \\ X_2 = \xi_2 \\ X_3 = 2X_1 + 3X_2 + \xi_3 \\ X_4 = \xi_4 * X_3 \end{cases}$$



$$\begin{cases} X_1 = \xi_1 \\ X_2 = \sin(11X_1) \\ X_3 = 5e^{X_1} \end{cases}$$



$$\begin{cases} X_1 = \xi_1 \\ X_2 = 13X_1 + \xi_2 \\ X_3 = 17\xi_3 * X_2 \\ X_4 = X_3^{\xi_4} \\ X_5 = \cos(X_1 + X_2 + X_4) \end{cases}$$



Theoretical overview. SCM graph

SCM graph is not only a fancy way to illustrate the SCM causal structure. Under some additional assumptions it may as well be used to derive information about conditional probabilities within the model directly from graph.

It turns out that to define “conditioning” on graph, it is enough to consider three three-variables SCMs.

● Mediating, or chain

$$\begin{cases} X = \xi_1 \\ Y = f_2(X, \xi_2) \\ Z = f_3(Y, \xi_3) \end{cases} \quad \begin{array}{c} X \\ \downarrow \\ Y \\ \downarrow \\ Z \end{array}$$

Essential relationships:
 $X \not\perp Z \mid Y$

Mediating rule:
If two variables in a chain are separated by a third variable, then they are independent given that third variable.

● Confounding, or fork

$$\begin{cases} X = \xi_1 \\ Y = f_2(X, \xi_2) \\ Z = f_3(X, \xi_3) \end{cases} \quad \begin{array}{c} X \\ \swarrow \quad \searrow \\ Y \quad Z \end{array}$$

Essential relationships:
 $Y \not\perp Z$
 $Y \perp Z \mid X$

Confounding rule:
If two dependent variables in a fork share a common parent variable, they are independent given that parent.

● Colliding

$$\begin{cases} X = \xi_1 \\ Y = \xi_2 \\ Z = f_3(X, Y, \xi_3) \end{cases} \quad \begin{array}{c} X \quad Y \\ \swarrow \quad \searrow \\ Z \end{array}$$

Essential relationships:
 $X \perp Y$
 $X \not\perp Y \mid Z$

Colliding rule:
If two independent variables share the same descendant variable, they are dependent given that descendant.

Theoretical overview. *D*-separation

The operation of “conditioning” on SCM graph is based on mediating, confounding and colliding rules.

Given a SCM graph, an undirected path between two vertices is **blocked by a set** of vertices $\mathcal{S}$ if

- $\mathcal{S}$ contains either a mediator or a confounder,
- $\mathcal{S}$ doesn't contain nor colliders neither any colliders' descendants.

Given a SCM graph, two sets of vertices A and B are called ***d*-separated** by a third set of variables $\mathcal{S}$ if every undirected path between vertices from A and B is blocked by $\mathcal{S}$.

We will denote the fact that A and B are *d*-separated by $\mathcal{S}$ by $A \perp_d B \mid \mathcal{S}$.

So, given a SCM and a graph, we want to associate true conditioning with *d*-separation so that for each sets of variables (vertices) A , B and $\mathcal{S}$

$$A \perp B \mid \mathcal{S} \Leftrightarrow A \perp_d B \mid \mathcal{S}.$$

Such association is possible when the joint probability behind the SCM is *Markovian* and *faithful* with respect to the graph.

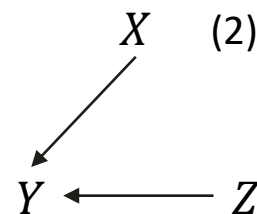
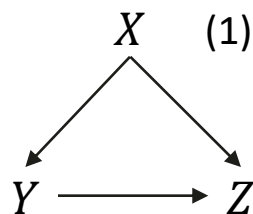
Theoretical overview. Faithfulness

Given a DAG G of vertices $X_1, \dots, X_d$ and a joint probability $P_{X_1, \dots, X_d}$, we say that $P_{X_1, \dots, X_d}$ is **faithful** with respect to G if

$$A \perp B \mid \mathcal{S} \Rightarrow A \perp_d B \mid \mathcal{S}.$$

The concept of faithfulness is nicely illustrated in the following example.

$$\begin{cases} X = \xi_1 \\ Y = aX + \xi_2 \\ Z = bX + cY + \xi_3 \end{cases}$$



It seems that the graph (1) suits the SCM above in the sense that $\mathbf{P}_{X,Y,Z}$ is faithful with respect to (1), but from the equations we get

$$Z = bX + cY + \xi_3 = bX + c(aX + \xi_2) + \xi_3 = (ac + b)X + c\xi_2 + \xi_3.$$

Let $ac + b = 0$. Then obviously $X \perp Z$ whereas $X \not\perp_d Z$, so $\mathbf{P}_{X,Y,Z}$ is not faithful with respect to (1).

One can show that the SCM we are considering is equivalent to the SCM below that is faithful to the graph (2).

$$\begin{cases} X = \xi_1 \\ Y = \tilde{a}X + \tilde{b}Z + \tilde{\xi}_2 \\ Z = \tilde{\xi}_3 \end{cases}$$

Theoretical overview. Markov property

Given a DAG G of vertices $X_1, \dots, X_d$ and a joint probability $P_{X_1, \dots, X_d}$, we say that

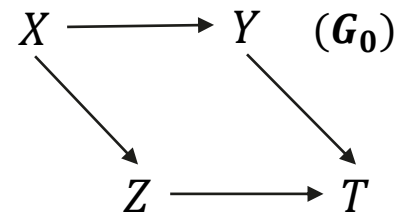
- $P_{X_1, \dots, X_d}$ meets the **global Markov property** with respect to G if
$$A \perp B \mid \mathcal{S} \Leftarrow A \perp_d B \mid \mathcal{S},$$
- $P_{X_1, \dots, X_d}$ meets the **local Markov property** with respect to G if each of $X_1, \dots, X_d$ is independent of its non-descendants given its parents (parental relations are derived from G).

For DAGs the local Markov property is equivalent to the global Markov property so will refer to any of them simply as the Markov property.

Given a DAG G of vertices $X_1, \dots, X_n$, all distributions $P_{X_1, \dots, X_n}$ that are Markovian with respect to G are denoted by $\mathcal{M}(G)$. For example, two following SCMs generate distributions that are both in $\mathcal{M}(G_0)$.

$$\begin{cases} X = \xi_1 \\ Y = f_2(X, \xi_2) \\ Z = f_3(X, \xi_3) \\ T = Y + Z + \xi_4 \end{cases}$$

$$\begin{cases} X = \xi_1 \\ Y = f_2(X, \xi_2) \\ Z = f_3(X, \xi_3) \\ T = Y * Z + \xi_4 \end{cases}$$

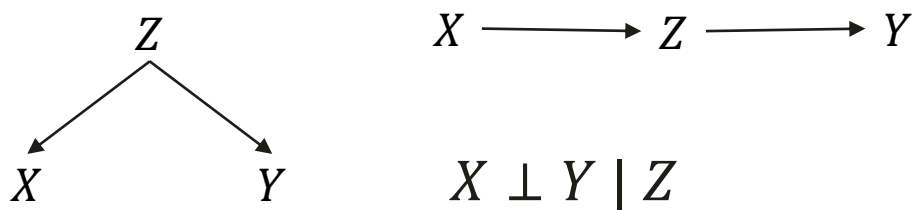


Theoretical overview. Markov property

A DAG G generates a list of d -separations (e.g., $X_1 \perp_d X_2 \mid \{X_3, X_4\}$, $X_2 \perp_d X_4 \mid \{X_5\}$).

On the one hand, all it takes for a distribution to be Markovian is to have the same list of conditional independencies. As we have seen, there are in fact an infinite number of distributions that are Markovian with respect to G .

On the other hand, it turns out that we cannot surely recover DAG from a list of d -separations. The reason for this is that chains and forks generate the same list of d -separations:



Thus we cannot recover DAG from a list of distributions that are Markovian with respect to that DAG as well. However, we can think of a class of all suitable DAGs.

Two DAGs G_1 and G_2 are **Markov equivalent**, or belong to the same **Markov equivalent class**, if $\mathcal{M}(G_1) = \mathcal{M}(G_2)$.

Theorem

Two DAGs G_1 and G_2 are Markov equivalent if and only if they

- have the same skeleton,
- have the same collidings.

Theoretical overview. Identifiability II

Given a SCM or a joint probability and a DAG, we can associate true conditioning with d -separation in the DAG in the way that

$$A \perp B \mid \mathcal{S} \Leftrightarrow A \perp_d B \mid \mathcal{S} \quad (1)$$

if and only if the joint probability is Markovian and faithful with respect to the DAG.

The proposition above makes us about other DAGs for which the equality (1) holds. We already know that it holds for all DAGs from the same Markov equivalence class – it turns out that this is the only possibility for Markovian and faithful distributions.

Theorem

Assume that $\mathbf{P}_{X_1, \dots, X_d}$ is Markovian and faithful with respect to $\mathbf{G}_0$.

Then

- for each $\mathbf{G} \in \mathcal{M}(\mathbf{G}_0)$ there exists SCM that generates $\mathbf{P}_{X_1, \dots, X_n}$,
- there is no $\mathbf{G} \notin \mathcal{M}(\mathbf{G}_0)$ such that $\mathbf{P}_{X_1, \dots, X_d}$ is Markovian and faithful with respect to $\mathbf{G}$.

Now as we have formulated the theorem above, we are able to reformulate the problem of identifiability.

Given a SCM, the problem of identifying whether a joint probability generated by the SCM entails a unique Markov equivalence class of DAGs or a DAG itself is called the problem of **identifiability**.

Theoretical overview. Identifiability II

Linear Gaussian model with homoscedastic errors

Consider the model

$$X_j = \sum_{k \in \mathcal{P}_j} \alpha_{jk} X_k + \xi_j, \quad (1)$$

where $\alpha_{jk} \in \mathbb{R} \setminus \{0\}$, $\{\xi_j\}$ are Gaussian i.i.d. random variables.

Theorem ([4])

A DAG associated with the model (1) is identifiable if $\{\xi_j\}$ have the same variance.

Linear non-Gaussian acyclic model, LiNGAM

Consider the model

$$X_j = \sum_{k \in \mathcal{P}_j} \alpha_{jk} X_k + \xi_j, \quad (2)$$

where $\alpha_{jk} \in \mathbb{R} \setminus \{0\}$, $\{\xi_j\}$ are non-Gaussian i.i.d. random variables.

Theorem ([5])

A DAG associated with the model (2) is identifiable if $\{\xi_j\}$ have strictly positive densities.

Nonlinear Gaussian models with additive noise

Consider the model

$$X_j = f_j(\mathcal{P}_j) + \xi_j, \quad (3)$$

where $\{\xi_j\}$ are Gaussian i.i.d. random variables.

Theorem ([6])

A DAG associated with the model (3) is identifiable if $\{f_j\}$ are three times differentiable and non-linear with respect to each of its' arguments.

Conditional independence-based methods

Conditional independence-based methods assume that the joint probability distribution is Markovian and faithful with respect to the underlying graph. That is, there one-to-one correspondence between d -separations in the graph and conditional independencies in the joint probability. One can use then conditional independence tests to estimate the Markov equivalence class of DAGs based on the following statements.

Theorem

Given a DAG G ,

- two vertices X, Y are adjacent if and only if they cannot be d -separated by any set of other vertices, (1)
- if two vertices X, Y aren't adjacent, they are d -separated by either $\mathcal{P}_X$ or $\mathcal{P}_Y$. (2)

The **IC** and **SGS** algorithms are based on the (1). For each pair of vertices, these methods consider all possible sets of other vertices and test whether the vertices of the pair are d -separated given some set of vertices. After all tests, two vertices are adjacent if and only if there is no set that d -separates these vertices.

The **PC** algorithm is based on the (2). It starts with the fully connected undirected graph and step-by-step increases the size of the condition set, starting with the empty set. At iteration k , checking whether vertices X and Y are d -separated it considers sets of size k that are subsets either of neighbors of X or the neighbors of Y .

Conditional independence-based methods

IC (Inductive Causation) algorithm

1. For each pair of vertices (X, Y) , search for a set $\mathcal{S}$ such that $X \perp Y \mid \mathcal{S}$.
Construct an undirected link between X and Y if and only if $\mathcal{S}$ is not found.
2. For each pair of non-adjacent vertices (X, Y) with a common adjacent vertex Z test if $X \perp Y \mid Z$.
If $X \not\perp Y \mid Z$, then Z is a collider for X and Y .
3. In the resulting partially directed graph, orient as many undirected edges as possible subject to two conditions:
 - the orientation should not create new colliders,
 - the orientation should not create a directed cycle.

One can see that the IC algorithm may suffer from high computational cost due to searching through all possible sets of variables.

The PC is more efficient than the IC because it doesn't consider all possible sets of variables. It is perceived as the main method for identifying the causality.

PC (Peter-Clark) algorithm

Let d be the number of variables.

0. Construct a fully connected undirected graph with vertices corresponding to the set of variables. For each pair of vertices (X, Y) , test if $X \perp Y$. If $X \perp Y$, then remove a link between X and Y .
1. For each pair of connected vertices (X, Y) , for each Z such that Z is either a neighbor of X or a neighbor of Y , test if $X \perp Y \mid Z$. If there is a test showing that $X \perp Y \mid Z$, then remove a link between X and Y .
- ...
- d-2. For each pair of connected vertices (X, Y) , for each $Z_1, \dots, Z_{d-2}$ such that $Z_1, \dots, Z_{d-2}$ are either neighbors of X or neighbors of Y , test if $X \perp Y \mid Z_1, \dots, Z_{d-2}$. If there is a test showing that $X \perp Y \mid Z$, then remove a link between X and Y .
- d-1. For each pair of non-adjacent vertices (X, Y) with a common adjacent vertex Z test if $X \perp Y \mid Z$.
If $X \perp Y \mid Z$, then Z is a collider for X and Y .
- d. In the resulting partially directed graph, orient as many undirected edges as possible subject to two conditions:
 - the orientation should not create new colliders,
 - the orientation should not create a directed cycle.

Conditional independence-based methods

Conditional independence-based methods strongly depend on the tests of independence. That is why they may suffer from the problems associated with such tests.

● Order of testing

As one can see, resulting graph depends on the order according to which tests are performed. So as the number of variables increases, the algorithm becomes more unstable. A few modifications were proposed to avoid the problem.

PC-stable ([7])	Conservative PC, or CPC ([8])	Fast Casual Inference, or FCI ([9])
Exploiting the idea of waiting until all tests for a vertex are performed before removing any edges associated with the vertex, the algorithm makes a DAG's skeleton more order-independent	Changing the rule of making up colliders in Step N-1 of the PC algorithm, makes the set of resulting colliders more order-independent	Adding some additional tests of independence for non-adjacent nodes, tries to handle the influence of possible hidden variables

● Estimating the regression functions

Mathematically $X \perp Y \mid Z$ is just the same as $X - f(Z) \perp Y - g(Z)$, where functions f and g are regression functions and deliver the best Z -approximation of X and Y respectively in terms of the L_2 -error (mean squared error). So to perform a test of conditional independence, one should find f and g , subtract $f(Z)$ and $g(Z)$ from X and Y respectively and then perform a test of independence on the resulting variables.

However in practice it is impossible to search for f and g through the set of all appropriate functions. That is why usually f and g are assumed to be linear, polynomial, etc. As it is known, in the gaussian case linear assumption is totally fair. But generally the assumption means that we can't be sure that the test's result corresponds with the reality.

Conditional independence-based methods

● Multiple hypothesis testing problem

Say, we need to test whether variables X and Y are independent. Then we have a null hypothesis $H_0: X \text{ and } Y \text{ are independent}$ and an alternative hypothesis $H_1: X \text{ and } Y \text{ are dependent}$.

Type I error, denoted as α_I , is the probability to reject H_0 given that H_0 is true.

Type II error, denoted as α_{II} , is the probability to not reject given that H_1 is true.

Significance level α is the desirable value of α_I . So if $\alpha = 0.05$, it means that we are willing to reject true H_0 only in 5% of all cases and try to come up with a test for which $\alpha_I = \alpha$, or at least $\alpha_I < \alpha$.

In the PC algorithm we have to conduct many tests at once. Let us show how the probability to reject at least one true hypothesis behave as the number of test M increases. For the sake of simplicity, assume that all null hypotheses are true and tests are independent.

$$\begin{aligned} \mathbf{P}\{\text{reject at least one true hypothesis}\} &= 1 - \mathbf{P}\{\text{not reject all true hypothesis}\} \\ &= 1 - \mathbf{P}\{\text{not reject the 1st true hypothesis}\} \cdot \dots \cdot \mathbf{P}\{\text{not reject the } M\text{th true hypothesis}\} = 1 - (1 - \alpha)^M \rightarrow 1 \text{ as } M \rightarrow \infty. \end{aligned}$$

For the common case, conclusion is the same – the probability to reject a true hypothesis increases in the number of tests, that is why one need to somehow adjust α .

Score-based methods

Like independence-based methods, score-based methods assume that the joint probability distribution is Markovian and faithful with respect to the underlying graph. These methods search for a suitable DAG over a set of DAGs using a measure of how well a DAG fits the data. This measure is called *score*. It is implied that a DAG encoding the wrong conditional probabilities will yield a low score.

Given a number of observations over variables X , a set of DAGs $\mathbf{G}$ and a score $\mathbf{S}$, find

$$\operatorname{argmax}_{G \in \mathbf{G}} \mathbf{S}(X, G).$$

If a parametric model with a set of parameters Θ is given as well, then find

$$\operatorname{argmax}_{G \in \mathbf{G}, \theta \in \Theta} \mathbf{S}(X, G, \theta).$$

Most common scores include

● Log-likelihood

Consider

$$\mathbf{S}(X, G, \theta) = \log p(X|G, \theta).$$

A DAG and a parametric model define a family of probability distributions. Assuming that this family has probability densities, given a number of observations, we are able to maximize log-likelihood and find a DAG and a set of parameters that best match the data.

Even if the model is identifiable, we still need to think about the consistency of the maximum log-likelihood estimator.

Score-based methods

● Penalized log-likelihood – Bayesian information criterion (BIC)

Consider

$$\mathbf{S}(X, G, \theta) = \log p(X|G, \theta) - \frac{\log n \cdot \text{Length of } \theta}{2}.$$

Here we add BIC penalty to the log-likelihood.

Due to the consistence of BIC, the score gives a consistent DAG. It means that a DAG derived from the model is getting closer to the true one as $n \rightarrow \infty$.

● Penalized log-likelihood – Akaike information criterion (AIC)

Consider

$$\mathbf{S}(X, G, \theta) = \log p(X|G, \theta) - \text{Length of } \theta.$$

Here we add AIC penalty to the log-likelihood.

Due to the efficiency of BIC, the score gives an efficient DAG. It means that a DAG derived from the model has the mean squared error that is getting closer to the true one's as $n \rightarrow \infty$.

● Bayesian scores

Consider

$$\mathbf{S}(X, G, \theta) = \log p(G) + \log \int p(X|G, \theta) p(\theta|G) d\theta.$$

Given prior probabilities $p(G)$ and $p(\theta)$, the score simply corresponds to the posterior probability $p(G|X)$.

Score-based methods

Let us show how log-likelihood is used as a score for the Additive Noise Models.

Say, we have observations $\mathbf{X}^1, \dots, \mathbf{X}^n$ over the variables $\{X_j\}$.

Common case

Let

$$X_j = f_j(\mathcal{P}_j) + \xi_j, \quad \text{or } X = f(X) + \xi,$$

where $\{\xi_j\}$ are Gaussian i.i.d. random variables, $\xi \sim N(\mathbf{0}, \Sigma)$.

Then the DAG and $\{f_j\}$ are derived from

$$\operatorname{argmax}_f \sum_{i=1}^n \left(\mathbf{X}^i - f(\mathbf{X}^i), \Sigma^{-1} (\mathbf{X}^i - f(\mathbf{X}^i)) \right)$$

Linear case

Let

$$X_j = \sum_{k \in \mathcal{P}_j} \alpha_{jk} X_k + \xi_j, \quad \text{or } X = AX + \xi,$$

Where $\alpha_{jk} \in \mathbb{R} \setminus \{0\}$, $\{\xi_j\}$ are Gaussian i.i.d. random variables, $\xi \sim N(\mathbf{0}, \Sigma)$.

Then the DAG and $\{\alpha_{jk}\}$ are derived from

$$\operatorname{argmax}_A \sum_{i=1}^n \|(A^T A)^{-1} A \mathbf{X}^i\|_2^2$$

Score-based methods

The number of DAGs grows super-exponentially in the number of vertices (OEIS A003024, [10]), and generally the problem of finding the optimal scoring DAG is NP-hard. That is why often it is infeasible to minimize a score over all DAGs.

● Greedy search

One can rely on greedy search technique instead of searching over all DAGs (for linear Gaussian SCMs see [11]). At each step of such algorithm there is a set of neighboring graphs, and one of them that fits the score the best becomes a new candidate for the causal graph. Neighbors may be defined using graph edit distance.

● Dynamic programming

Based on the density factorization, we have

$$\log p(X|G, \theta) = \sum_j \sum_{i=1}^n \log p(X_j^i | \mathcal{P}_j(G), \theta),$$

which can be interpreted as a sum of local scores. This decomposition can be used to solve the complexity problem at least partially using the methods of dynamic programming ([12], [13], [14]).

● Integer linear programming (ILP)

One can treat a DAG as a vector so that a score becomes a function defined on a multi-dimensional vector Space (for example, $\{0,1\}^d$). Then existing methods of solving NP-hard ILP problems may be applied ([15], [16]).

Bibliography

- [1] J. Peters, D. Janzing, and B. Schölkopf (2017). Elements of Causal Inference: Foundations and Learning Algorithms, The MIT Press, 288 p.
- [2] A. Hyvärinen, S. Shimizu, and P. Hoyer (2008). Causal modelling combining instantaneous and lagged effects: An identifiable model based on non-Gaussianity, Proceedings of the 25th International Conference on Machine Learning (ICML), p. 424–431.
- [3] K. Zhang and A. Hyvärinen (2009). On the identifiability of the post-nonlinear causal model. Proceedings of the 25th Annual Conference on Uncertainty in Artificial Intelligence (UAI), p. 647–655.
- [4] J. Peters and P. Bühlmann (2014). Identifiability of Gaussian structural equation models with equal error variances. Biometrika, 101 (1), p. 219–228.
- [5] S. Shimizu, P. O. Hoyer, A. Hyvärinen, and A. J. Kerminen (2006). A linear non-Gaussian acyclic model for causal discovery. Journal of Machine Learning Research, 7, p. 2003–2030.
- [6] J. Peters, J. M. Mooij, D. Janzing, and B. Schölkopf (2014). Causal discovery with continuous additive noise models. Journal of Machine Learning Research, 15, p. 2009–2053.
- [7] D. Colombo and M. H. Maathuis (2014). Order-independent constraint-based causal structure learning. Journal of Machine Learning Research, 15, p. 3741–3782.
- [8] J. Ramsey, J. Zhang, and P. Spirtes (2006). Adjacency-faithfulness and conservative causal inference. Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI-2006), AUAI Press, Arlington.
- [9] P. Spirtes (2001). An anytime algorithm for causal inference. Proceedings of the 8th International Conference on Artificial Intelligence and Statistics (AISTATS), p. 213–221.
- [10] [A003024 – OEIS](#)
- [11] D. M. Chickering (2002). Optimal structure identification with greedy search. Journal of Machine Learning Research, 3, p. 507–554.
- [12] T. Silander and P. Myllymak (2006). A simple approach for finding the globally optimal Bayesian network structure. Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI), p. 445–452.
- [13] M. Koivisto and K. Sood (2004). Exact Bayesian structure discovery in Bayesian networks. Journal of Machine Learning Research, 5, p. 549–573.
- [14] M. Koivisto (2006). Advances in exact Bayesian structure discovery in Bayesian networks. Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI), p. 241–248.
- [15] T. Jaakkola, D. Sontag, A. Globerson, and M. Meila (2010). Learning Bayesian network structure using LP relaxations. Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS), p. 358–365.
- [16] J. Cussens (2011). Bayesian network learning with cutting planes. In Proceedings of the 27th Annual Conference on Uncertainty in Artificial Intelligence (UAI), p. 153–160.