Outline

- ☑ 背景知识
- ☑ 因果发现
  - 什么是因果发现
  - 经典方法：基于约束的方法、基于因果函数的方法
  - 研究进展：隐变量问题、非独立同分布问题
  - 应用探索：故障检测
- ☑ 因果性学习

什么是因果发现

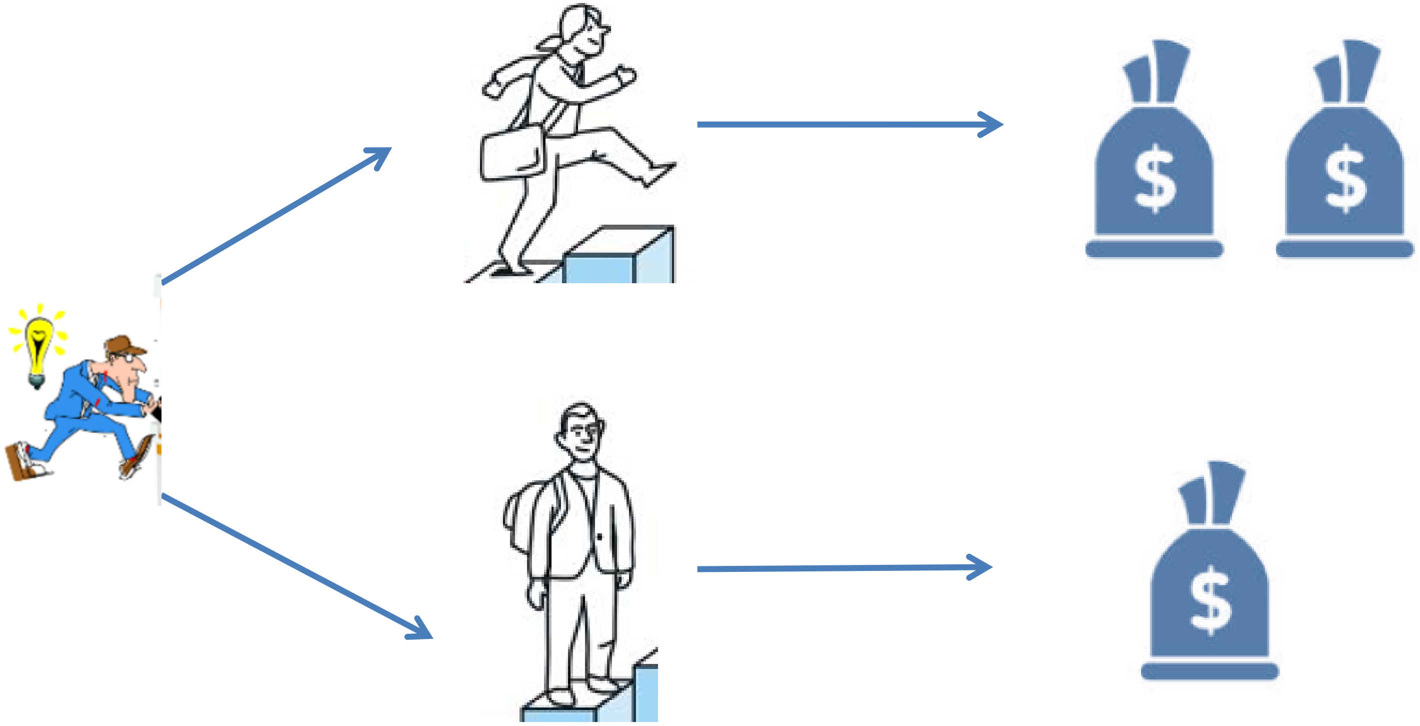
什么是因果关系发现?

☑ 因果发现：回答“为什么？”

为什么
苹果
会往
地上掉?

如何发现因果关系？基于干预实验的方法

☑ **基于实验的方法**：干预原因，结果会发生改变



如何发现因果关系？基于观察数据的方法

☑ **基于观察数据的方法**： 观察数据+因果假设⇒因果模型

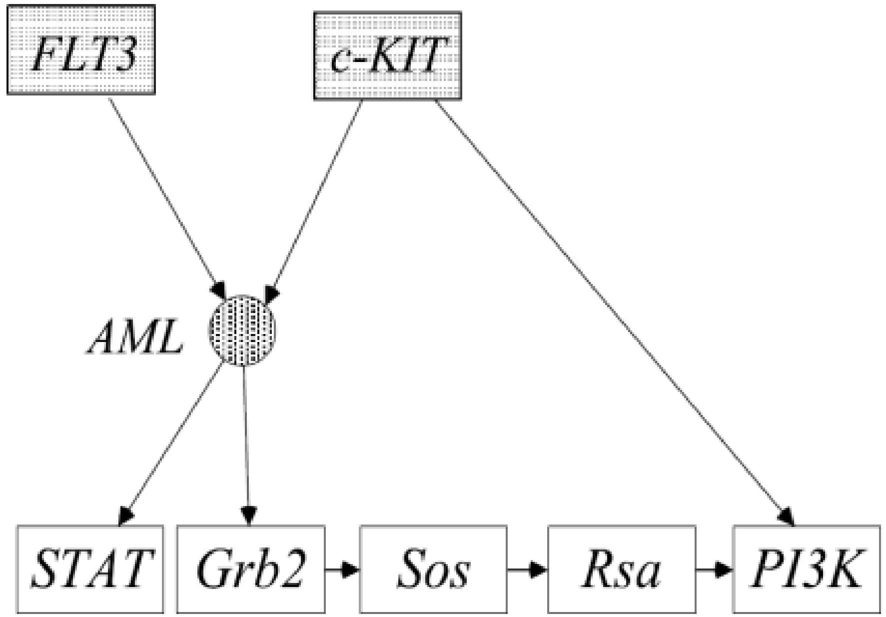
观察数据

	V1	V2	,,,	Vn
S1				
S2				
…				
…				
Sm				

\+

因果假设

因果发现



因果模型

经典方法

- ☑ 基于约束的因果发现方法
- ☑ 基于函数的因果发现方法
- ☑ 混合型因果发现方法

Structural Causal Models and Graphical Causal Models

☑ The structural causal model (**SCM**) is a framework that can be used for multivariate analysis, which can be used to describe the real-world related variables and their interactions.

☑ $X_i = f_i(pa_i, E_i), i = 1,2, \dots, n$

- pa_i : parents of X_i
- E_i : exogenous variables / errors / disturbances
- Each equation represents an autonomous mechanism
- Describes how nature assigns values to variables of interest

SCM

$X = \{X_1, X_2, X_3, X_4\}, E = \{E_1, E_2, E_3, E_4\}, F = \{f_3, f_4\}$

$$
\begin{aligned}
X_1 &= E_1 \\
X_2 &= E_2 \\
X_3 &= f_3(X_1, X_2, E_3) \\
X_4 &= f_4(X_3, E_4)
\end{aligned}
$$

Graphical Causal Model: Directed Acyclic Graphs (DAG)

$Nodes: \{X_1, X_2, X_3, X_4\}$

$Edges: \{X_1 \rightarrow X_3,$

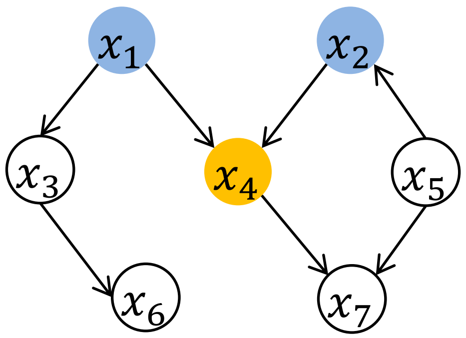
$X_2 \rightarrow X_3,$

$X_3 \rightarrow X_4\}$

$$
\begin{tikzcd}
X_1 \arrow[dr] & & X_2 \arrow[dl] \\
& X_3 \arrow[d] & \\
& X_4 &
\end{tikzcd}
$$

Constraint-based methods: assumptions

- ☑ **Causal Markov Assumption**: A variable X is independent of every other variable (except X's effects) conditional on all of its direct causes.



$$\Leftrightarrow \quad x_4 \perp\!\!\!\perp \{x_3, x_6, x_5\} \mid \{x_1, x_2\}$$

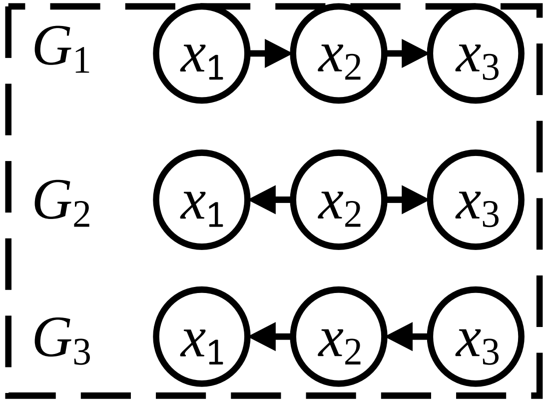
- ☑ **Causal Faithfulness Assumption**: for all observed variables, X_i is independent of X_j conditional on variables $\mathbf{Z}$ if and only if the Markov Assumption for $\mathcal{G}$ entails such conditional independencies.

Constraint-based methods

☑ (Conditional) Independence Test

$x_1 \perp\!\!\!\perp x_3 | x_2$

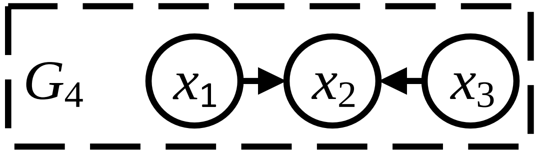
Inferring



Markov Equivalence Class

$x_1 \perp\!\!\!\perp x_3$
$x_1 \not\perp\!\!\!\perp x_3 | x_2$

Inferring



V-Structure

Constraint-based methods: PC

☑ PC (Peter-Clark from CMU)

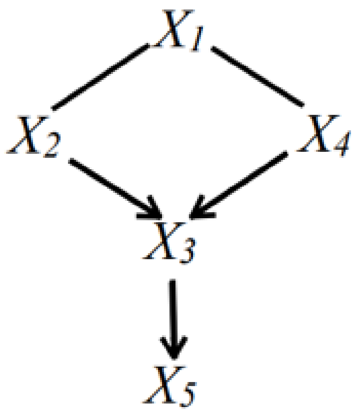
X_1	X_2	X_3	X_4	X_5
-1.1	1	1.3	0.2	-0.7
2.1	2	3.1	-1.3	-1.6
3.1	4.2	-2.6	0.6	2.1
2.3	-0.6	-3.5	0.8	2.3
1.3	-1.7	0.9	2.4	-1.4
-1.8	0.9	-1.3	0.9	0.7
…	…	…	…	…

(a) 数据

$$X_1 \perp\!\!\!\perp X_5 | X_3$$
$$X_2 \perp\!\!\!\perp X_4 | X_1$$
$$X_2 \perp\!\!\!\perp X_5 | X_3$$
$$X_4 \perp\!\!\!\perp X_5 | X_3$$
$$X_1 \perp\!\!\!\perp X_3 | \{X_2, X_4\}$$

(b) 条件独立性

?



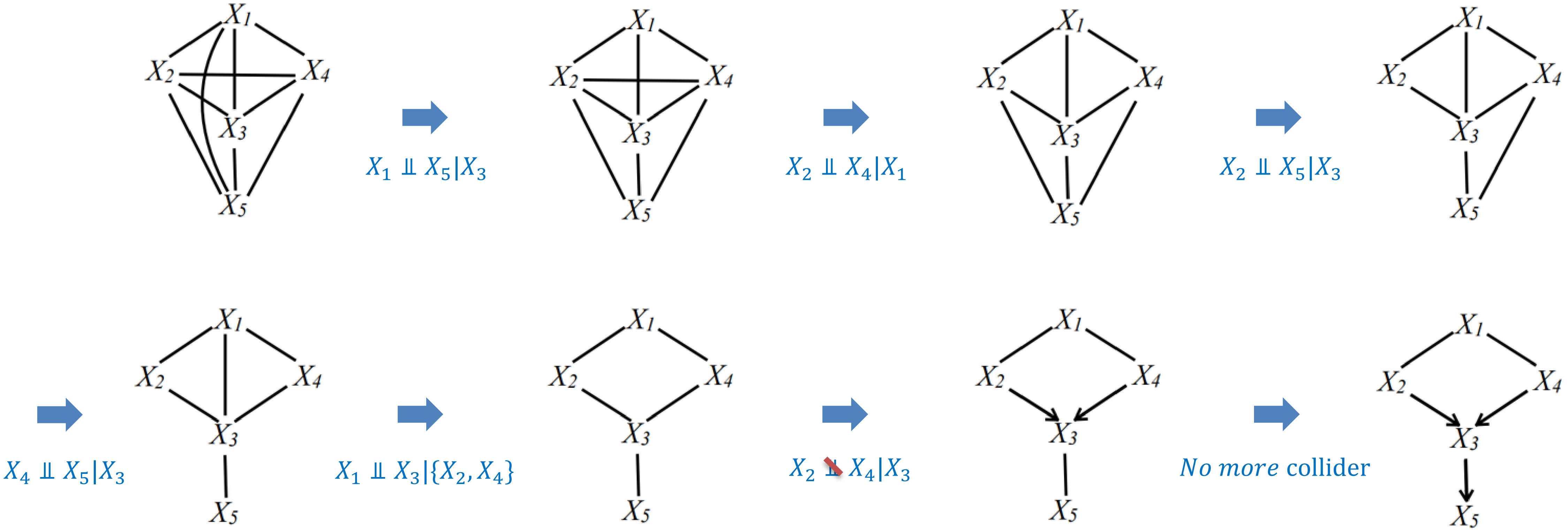
(c) 因果结构

Spirtes, P., Glymour, C. N., Scheines, R., & Heckerman, D. *Causation, prediction, and search*. MIT press, 2000.

Constraint-based methods: PC

☑ A toy example of PC (Peter-Clark)

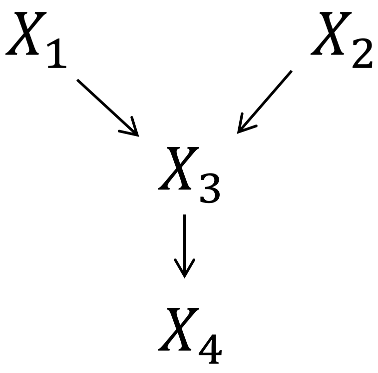
$$X_1 \perp\!\!\!\perp X_5 | X_3$$
$$X_2 \perp\!\!\!\perp X_4 | X_1$$
$$X_2 \perp\!\!\!\perp X_5 | X_3$$
$$X_4 \perp\!\!\!\perp X_5 | X_3$$
$$X_1 \perp\!\!\!\perp X_3 | \{X_2, X_4\}$$

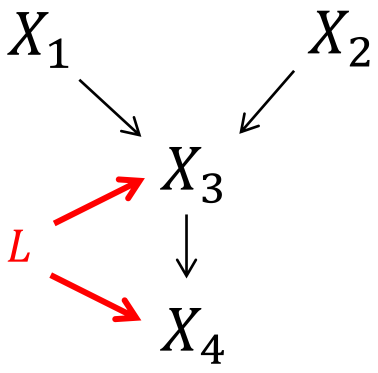


Constraint-based methods: FCI

☑ How to deal with latent confounders?

- if there is a latent confounder L behind X_3 and X_4





$$X_1 \perp\!\!\!\perp X_2$$
$$X_1 \perp\!\!\!\perp X_4 | X_3$$
$$X_2 \perp\!\!\!\perp X_4 | X_3$$

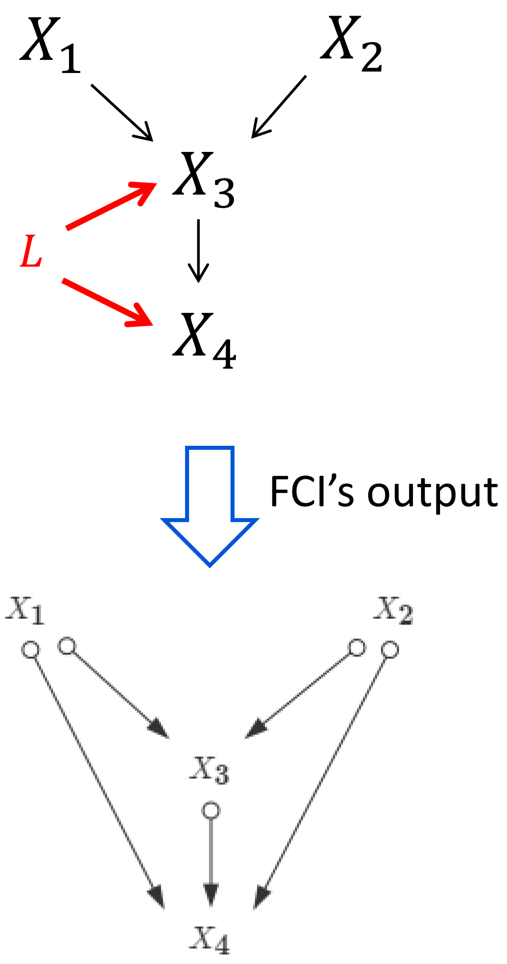
$$X_1 \perp\!\!\!\perp X_2$$
$$X_1 \not\perp\!\!\!\perp X_4 | X_3$$
$$X_2 \not\perp\!\!\!\perp X_4 | X_3$$

Spirtes, P., Glymour, C. N., Scheines, R., & Heckerman, D. *Causation, prediction, and search*. MIT press, 2000.

Constraint-based methods: FCI

☑ **FCI (Fast Causal Inference):** allows Confounders

- Results represented by PAGs (Partial Ancestral Graphs)



X_1	X_2	X_1 and X_2 are not adjacent
X_1 ∘⟶	X_2	X_2 is not an ancestor of X_1
X_1 ∘—∘	X_2	No set d-separates X_2 and X_1
X_1 ⟶	X_2	X_1 *is a cause of* X_2
X_1 ⟷	X_2	There is a latent common cause of X_1 *and* X_2

Spirtes, P., Glymour, C. N., Scheines, R., & Heckerman, D. *Causation, prediction, and search*. MIT press, 2000.

Constraint-based methods: limitations

☑ Limitations of constraint-based methods

$$x_1 \perp\!\!\!\perp x_3 | x_2 \xrightarrow{\text{Inferring}}$$

G_1: $x_1 \rightarrow x_2 \rightarrow x_3$

G_2: $x_1 \leftarrow x_2 \rightarrow x_3$

G_3: $x_1 \leftarrow x_2 \leftarrow x_3$

Markov Equivalence Class

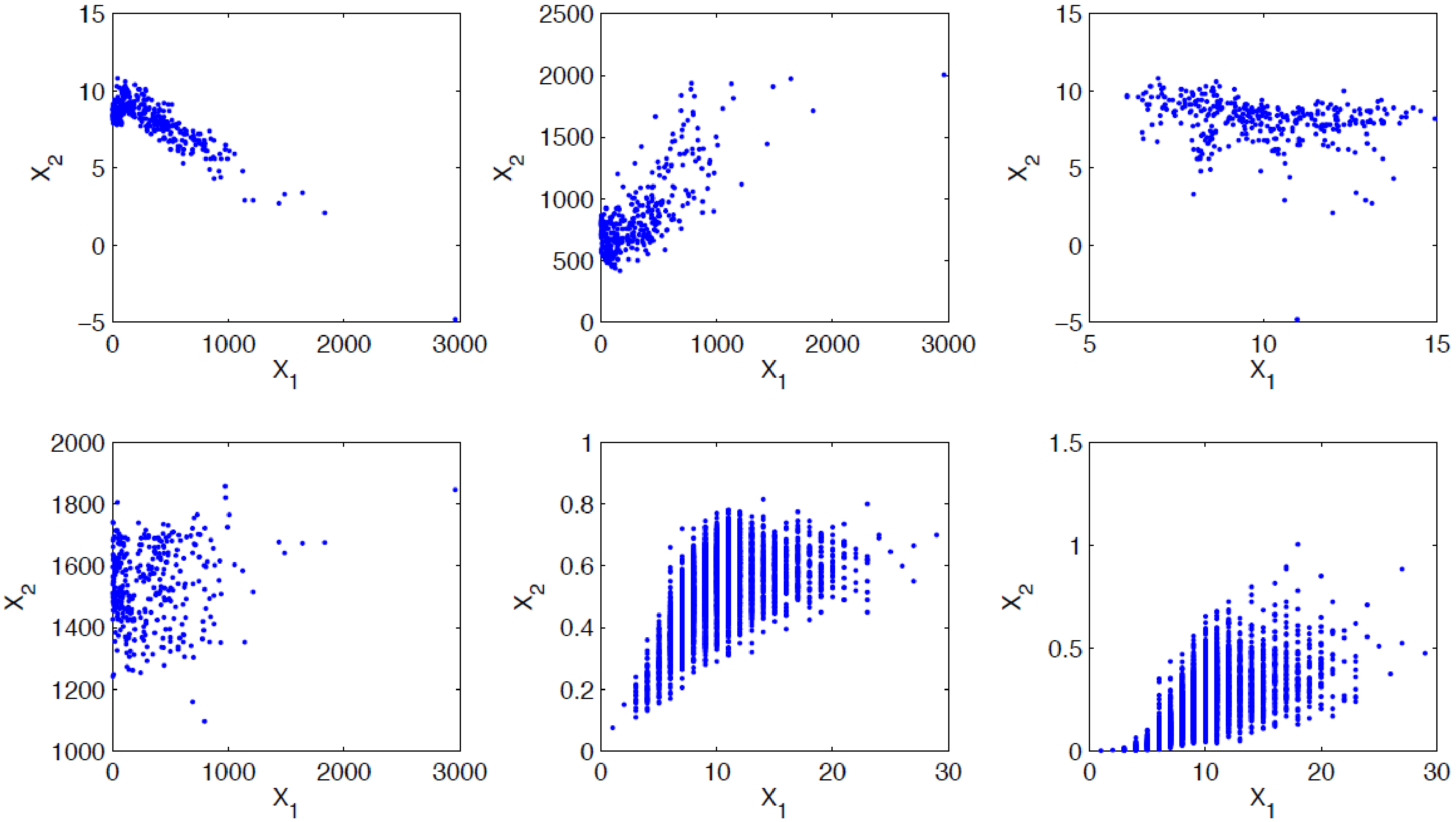
Problem: Cannot identify the structures belonging to the Markov Equivalence Class

Spirtes, P., Glymour, C. N., Scheines, R., & Heckerman, D. *Causation, prediction, and search*. MIT press, 2000.

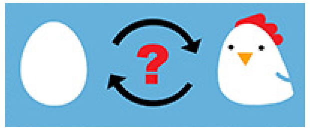
经典方法

- ☑ 基于约束的因果发现方法
- ☑ 基于函数的因果发现方法
- ☑ 混合型因果发现方法

Can we directly distinguish cause from effect?



$X_1 \rightarrow X_2$ or $X_2 \rightarrow X_1$?



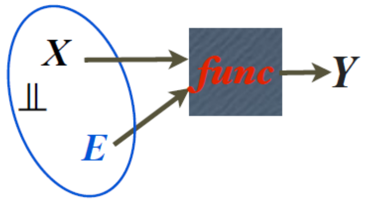
Causal Function based methods: assumptions

☑ **Considering the data generating process**, effect generated from causes and noises, represented with functional causal model:

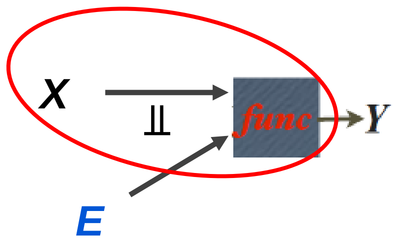
$$Y = f(X, E)$$

☑ **Introducing additional assumptions**

- Independent noise assumption: Independence between the causes X and noises E



- Independent mechanism assumption: independence between the causes X and process f

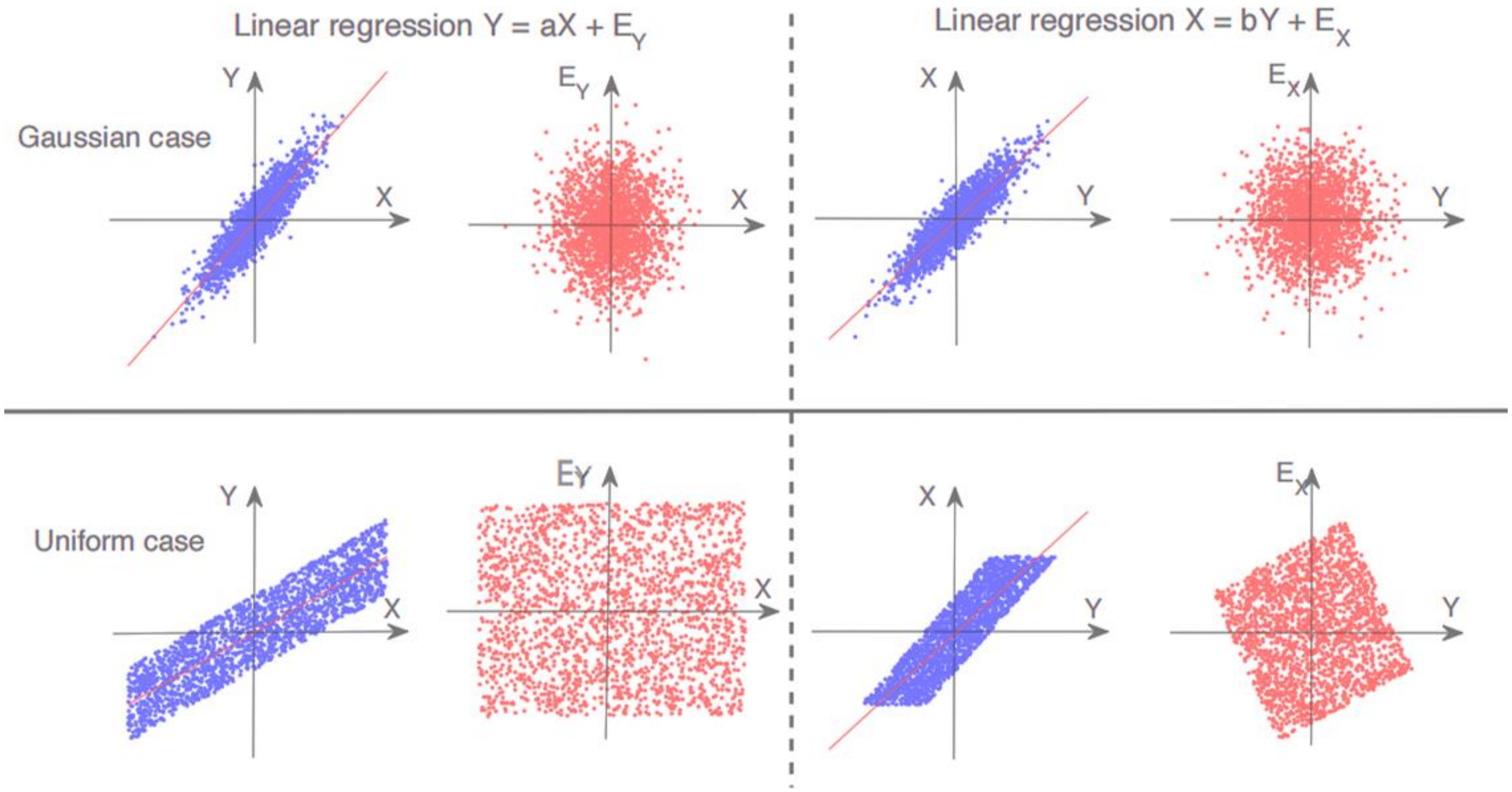


Causal Function based methods: independent Noise (IN) Condition

☑ Causal Asymmetry in the Linear non-Gaussian Case

- Data generated by $Y = aX + E$ $(i.e., X \rightarrow Y)$

☑ $(\mathbf{X}, Y)$ follows the IN condition iff regression residual $Y - \tilde{\omega}^{\mathrm{T}}\mathbf{X}$ is independent from $\mathbf{X}$



Causal Function based method: LiNGAM

☑ Under the above assumptions, the LiNGAM can be expressed as

$$\mathbf{X} = \mathbf{B}\mathbf{X} + \boldsymbol{E}$$

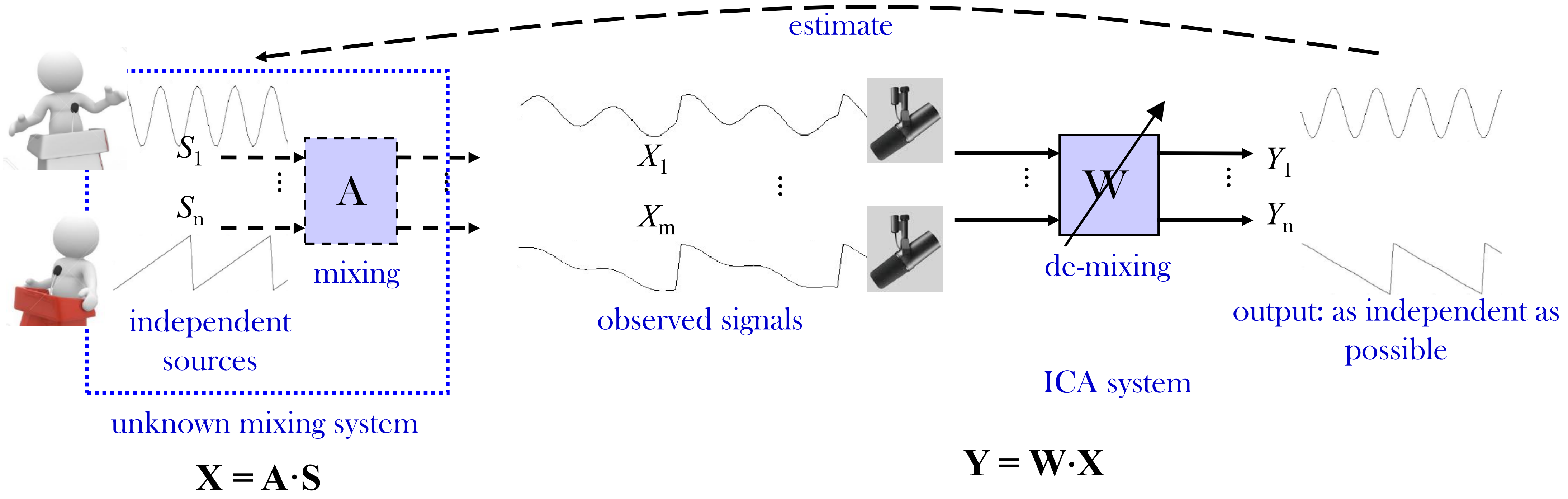
- $\boldsymbol{X}$ is a p-dimensional random vector, representing the observed variable.
- $\boldsymbol{B}$ is $p \times p$-dimensional matrix, which represents the connection weight between the observed variables.
- $\boldsymbol{E}$ is a p-dimensional non-Gaussian random noise variable.

☑ Because of the DAG assumption, there exists a permutation matrix $\boldsymbol{P} \in R^{m \times m}$ such that $\boldsymbol{B}'$ $= \boldsymbol{P}\boldsymbol{B}\boldsymbol{P}^T$ is a strict lower triangular matrix and diagonal elements are all 0

Shimizu S, Hoyer P O, Hyvärinen A, et al. *A linear non-Gaussian acyclic model for causal discovery*[J]. Journal of Machine Learning Research, 2006, 7(10).
Shimizu S. LiNGAM: *Non-Gaussian methods for estimating causal structures*[J]. Behaviormetrika, 2014, 41(1): 65-98.

LiNGAM: Independent Component Analysis

☑ ICA, **Independent Component Analysis**, can be used to solve the LiNGAM



☑ Assumptions in ICA

- At most one of S_i is Gaussian
- Size(X) >= Size(S), and **A** is of full column rank

Then **A** can be estimated up to column scale and permutation indeterminacies

Hyvärinen et al., *Independent Component Analysis*, 2001

LiNGAM: analysis by ICA

- ☑ LiNGAM:

$$\mathbf{X} = \mathbf{B}\mathbf{X} + \boldsymbol{E}$$

- ☑ ICA:

$$\boldsymbol{Y} = \boldsymbol{W}\boldsymbol{X}$$

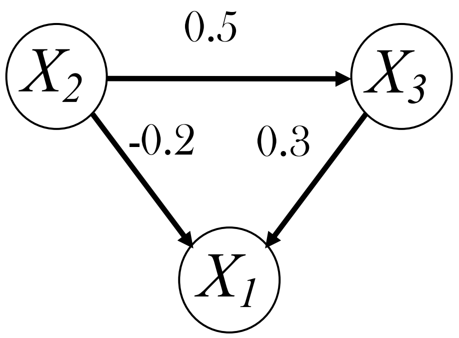
$$\boldsymbol{B} = \boldsymbol{I} - \boldsymbol{W}$$

- ☑ An example

$$\begin{bmatrix} E_1 \\ E_3 \\ E_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ -0.5 & 1 & 0 \\ 0.2 & -0.3 & 1 \end{bmatrix}}_{W} \cdot \begin{bmatrix} X_2 \\ X_3 \\ X_1 \end{bmatrix}$$

$$\Leftrightarrow \begin{cases} X_2 = E_1 \\ X_3 = 0.5X_2 + E_3 \\ X_1 = -0.2X_2 + 0.3X_3 + E_2 \end{cases}$$

So we have the causal relation:

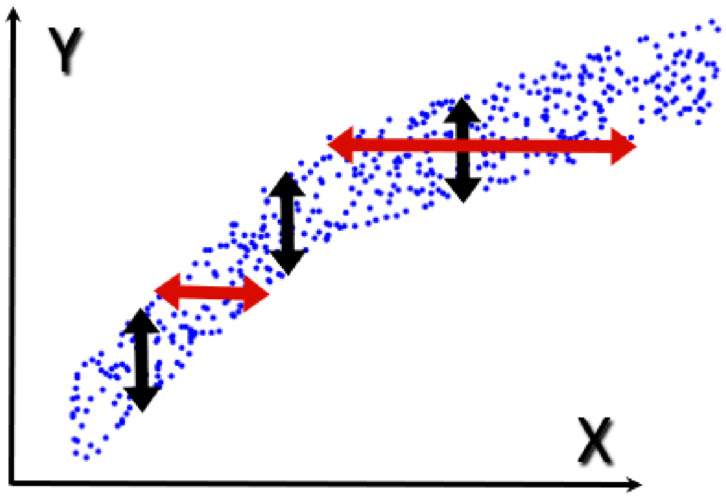


Shimizu S, Hoyer P O, Hyvärinen A, et al. *A linear non-Gaussian acyclic model for causal discovery*[J]. Journal of Machine Learning Research, 2006, 7(10).

Causal Function based method: ANM

☑ Hoyer et al. proved that nonlinear functions can play a similar role to non-Gaussian models, which can be used to identify causal directions.

$$Y = f(X) + E \quad with$$
$$E \perp\!\!\!\perp X$$



(Hoyer et al., 2009)

Hoyer P O, Janzing D, Mooij J M, et al. *Nonlinear causal discovery with additive noise models*. NeurIPS. 2008.

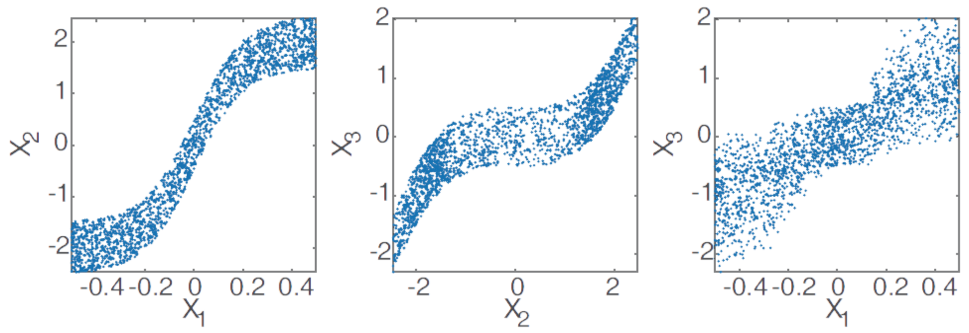
Causal Function based method: practical issue - hidden variables

☑ Motivation: Non-Transitivity of Nonlinear Causal Model

- Let the direct cause in $\boldsymbol{X_1} \rightarrow \boldsymbol{X_2} \rightarrow \boldsymbol{X_3}$ satisfy the additive noise model (ANM):

$$\begin{cases} X_1 \sim U(-0.5, 0.5) \\ X_2 = 2\tanh(5X_1) + N_2 \\ X_3 = \left(\dfrac{X_2}{2}\right)^3 + N_3 \end{cases}$$

- However, the causal influence $\boldsymbol{X_1} \rightarrow \boldsymbol{X_3}$ does not necessarily follow the additive noise model,



(a) (b) (c)

and we might not identify the causal by simply test the independence between cause and noise.

Causal Function based method: cascade Additive Noise Model (CANM)

☑ There exists a sequence of unmeasured intermediate variables Z between X and Y, where $N_1, N_2, N_3, \boldsymbol{\epsilon}$ are the additive noise at each direct cause.

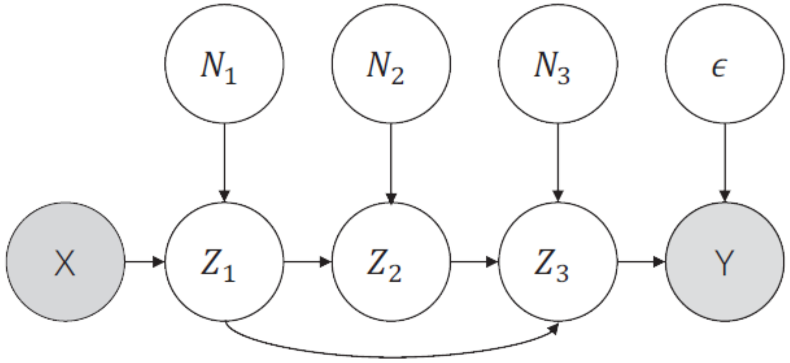


Figure: Illustration of the CANM

☑ Formally:

$$\begin{cases} Z_1 = f_1(X) + N_1 \\ Z_t = f_t(\mathbf{Z}_{pa(t)}) + N_t \\ Y = f_{T+1}\left(\mathbf{Z}_{pa(y)}\right) + \epsilon \end{cases}$$

☑ How to solve it?

Cai R, Qiao J, Zhang K, et al. *Causal discovery with cascade nonlinear additive noise models*. IJCAI, 2019.

Cascade Additive Noise Model (CANM)

☑ Given data set $\boldsymbol{\mathcal{D}} = \{\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}\}_{\boldsymbol{i=1}}^{\boldsymbol{m}}$,the marginal likelihood is given as follow:

$$log \prod_{i=1}^{m} \int p_\theta \left(x^{(i)}, y^{(i)}, z\right) dz \Leftrightarrow log \prod_{i=1}^{m} \int p_\theta \left(x^{(i)}, \epsilon^{(i)}, n\right) dn$$

① Decomposes the joint likelihood based on the Markov condition

② Applies the independence property between the cause and the noise, i.e.,

$$p(Z_t|\mathbf{Z}_{pa(t)}) = p(N_t = Z_t - f_t(\mathbf{Z}_{pa(t)})|\mathbf{Z}_{pa(t)}) \underline{\underline{\mathbf{Z}_{pa(t)} \perp N_t}}\, p(N_t = Z_t - f_t(\mathbf{Z}_{pa(t)}))$$

③ At the same time, we replace $d\boldsymbol{z} = d\boldsymbol{n}$ and rewrite function $f_{T+1}(\mathbf{Z}_{pa(y)})$ as $f(X, \mathbf{N})$.

☑ The unobserved intermediate variables z can be replaced by n.

Cai R, Qiao J, Zhang K, et al. *Causal discovery with cascade nonlinear additive noise models*. IJCAI, 2019.

CANM: variational solution

- ☑ Purpose: Find $q_\phi(\mathbf{N}|X,Y)$ to approximate $p_\theta(\mathbf{N}|X,Y)$
- ☑ Action: Optimize a variational lower bound (ELBO) of the marginal log-likelihood

$$\log\prod_{i=1}^{m}\int p_\theta\left(x^{(i)},\epsilon^{(i)},n\right)dn$$

$$\geq\sum_{i=1}^{m}\log p\left(x^{(i)}\right)-KL(q_\phi\left(n|x^{(i)},y^{(i)}\right)\|p_\theta(n))+E_{n\sim q_\phi(n|x^{(i)},y^{(i)})}\left[\log p\left(\epsilon^{(i)}=y^{(i)}-f\left(x^{(i)},n;\theta\right)\right)\right]$$

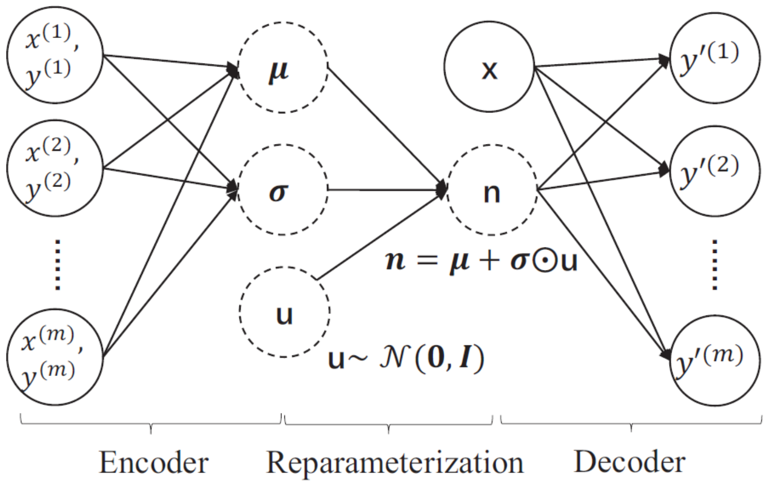
- ☑ The lower bound is tight at the maximum of ELBO i.e. $q_\phi(\mathbf{N}|X,Y)$ equal $p_\theta(\mathbf{N}|X,Y)$

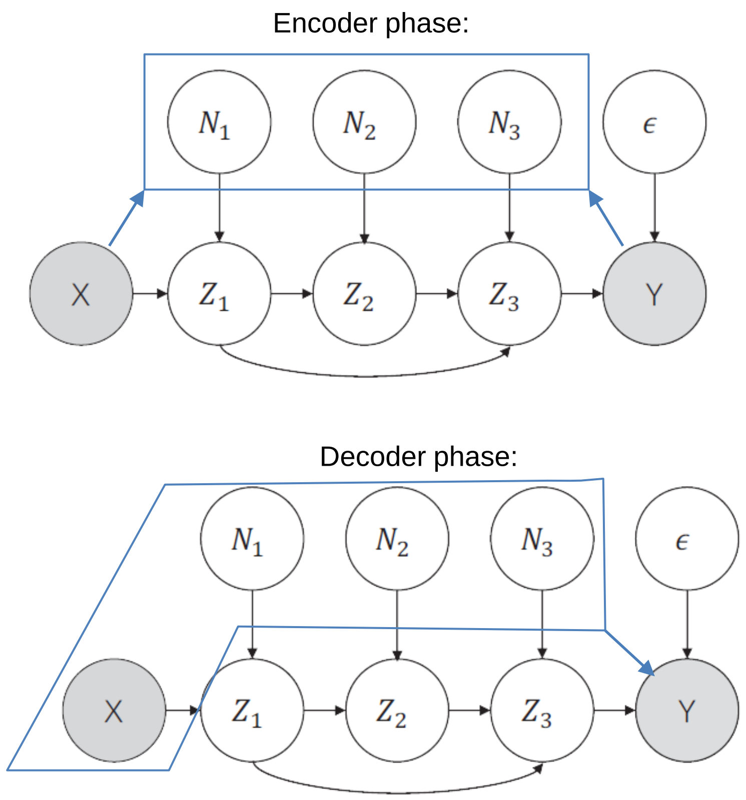
Cai R, Qiao J, Zhang K, et al. *Causal discovery with cascade nonlinear additive noise models*. IJCAI, 2019.

CANM: Encoder and Decoder Scheme

☑ Steps of VAE to optimize $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{X}, \boldsymbol{Y})$

1. Encode data into $\boldsymbol{\mu}, \boldsymbol{\sigma}$.
2. Sample the $\boldsymbol{n}$ through $\boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{u}$, where $\boldsymbol{u} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
3. Reconstruct $\boldsymbol{y}$ using $\mathbf{x}, \boldsymbol{n}$.





Cai R, Qiao J, Zhang K, et al. *Causal discovery with cascade nonlinear additive noise models*. IJCAI, 2019.

CANM: algorithm

- ☑ Model selection: select the best number of latent variables.
- ☑ To estimate the marginal likelihood to identify the causal relationship.

Algorithm 1 Inferring causal direction with CANM

Input: Data samples $\{(x^{(i)}, y^{(i)})\}_{i=1}^{m}$.
Output: The causal direction.

①
Split the data into training and test sets;
Choose the best number of latent variables by optimizing the variational lower bound (Eq. (3)) on the training set and evaluating the performance on the test set;

②
Optimize the lower bound in both directions with the best number of latent variables on the full dataset, obtaining $\mathcal{L}_{X\to Y}$ and $\mathcal{L}_{Y\to X}$ (see Eq. (4)), respectively.
if $\mathcal{L}_{X\to Y} > \mathcal{L}_{Y\to X} + \delta$, where δ is a pre-asigned small positive number, **then**,
    Infer $X \to Y$
else if $\mathcal{L}_{X\to Y} < \mathcal{L}_{Y\to X} - \delta$, **then**
    Infer $Y \to X$
else
    Non-identifiable
end if

Cai R, Qiao J, Zhang K, et al. *Causal discovery with cascade nonlinear additive noise models*. IJCAI, 2019.

CANM: identifiability

☑ Theorem 1. *Let $\boldsymbol{X} \rightarrow \boldsymbol{Y}$ follow the cascade additive noise model, while there exists a backward model following the same form, i.e.*

$$Y = f(X, N) + \epsilon, \qquad X, N, \text{and } \epsilon \text{ are independent,}$$
$$X = g(Y, \widehat{N}) + \hat{\epsilon}, \qquad Y, \widehat{N}, \text{and } \hat{\epsilon} \text{ are independent,}$$

then the noise distribution of the reverse direction $p_{\hat{\epsilon}}$ must be

$$p_{\hat{\epsilon}}(\hat{\epsilon}) = \int e^{2\pi i \hat{\epsilon}\cdot v} \frac{\int \int p(x) p(n) p_{\epsilon}(y - f(x,n)) e^{-2\pi i x \cdot v} dn dx}{p(y) \int p(\hat{n}) e^{-2\pi i g(y,\hat{n})\cdot v} d\hat{n}} dv,$$

where $\boldsymbol{f}, \boldsymbol{g}$ denote the function implied by the cascade process.

- **The main result of our main theorem is that, If the model is non-identifiable, one strict condition must hold:**

$$\forall y_1, y_2, \int e^{2\pi i \hat{\epsilon}\cdot v} \frac{\int \int p(x) p(\mathbf{n}) p_{\epsilon}(y_1 - f(x,\mathbf{n})) e^{-2\pi i x \cdot v} d\mathbf{n} dx}{p(y_1) \int p(\widehat{\mathbf{n}}) e^{-2\pi i g(y_1,\widehat{\mathbf{n}})\cdot v} d\widehat{\mathbf{n}}} dv$$
$$= \int e^{2\pi i \hat{\epsilon}\cdot v} \frac{\int \int p(x) p(\mathbf{n}) p_{\epsilon}(y_2 - f(x,\mathbf{n})) e^{-2\pi i x \cdot v} d\mathbf{n} dx}{p(y_2) \int p(\widehat{\mathbf{n}}) e^{-2\pi i g(y_2,\widehat{\mathbf{n}})\cdot v} d\widehat{\mathbf{n}}} dv$$

Intuitively, such a condition holds only in restrictive cases.

Cai R, Qiao J, Zhang K, et al. *Causal discovery with cascade nonlinear additive noise models*. IJCAI, 2019.

CANM: experiments - synthetic data

- ☑ Depth: as the depth increases, the accuracy of CANM is stable and around 90% accuracy with a slight decrease, while the performance of the rest methods decreases rapidly as the depth grows.
- ☑ Sample Size: as the sample size grows the accuracy increase, and even in the small sample size, CANM still outperforms the other methods.
- ☑ Fixed Structure: the variance of the likelihood decreases and the accuracy increases as the sample size grows.

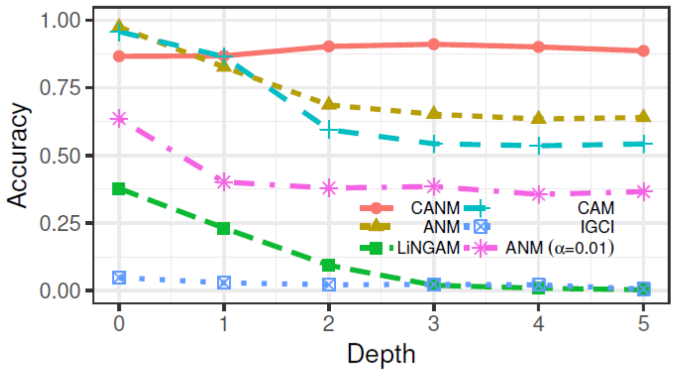


Figure 4: Sensitivity to Depth.

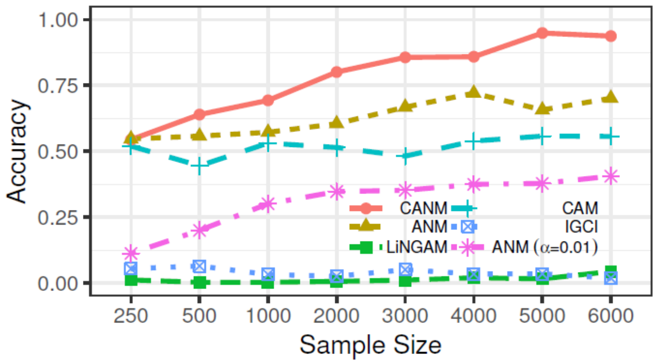


Figure 5: Sensitivity to Sample.

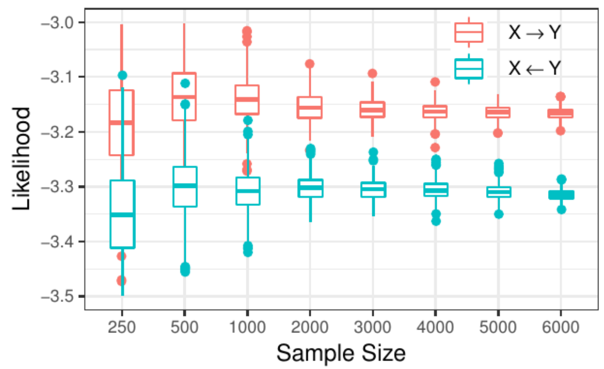


Figure 6: Sensitivity to Sample in a Fixed Structure.

Cai R, Qiao J, Zhang K, et al. *Causal discovery with cascade nonlinear additive noise models*. IJCAI, 2019.

CANM: experiments - real world data

☑ The electricity consumption dataset

- Hour of day → Temperature → electricity load
  - This data might exist more than one unmeasured variables because in the same hour of day there have different electricity load and the reason could be the season.
  - CANM successfully catch this latent variable because the prediction of electricity load, the red point, separating into both upper and lower parts.

☑ Stock Market Dataset

- Hutchison → Cheung Kong → Sun Hung Kai.
  - The fitted intermediate variable from CANM has a high correction ($\rho = 0.54$) with the ground truth (Cheung Kong)

Electricity Consumption Dataset

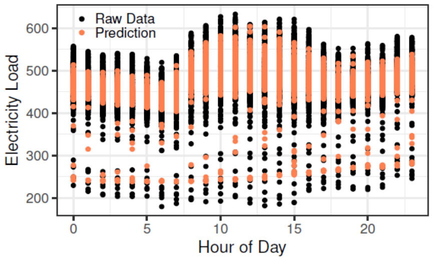


Figure 7: Hour of Day Against Electricity Load.

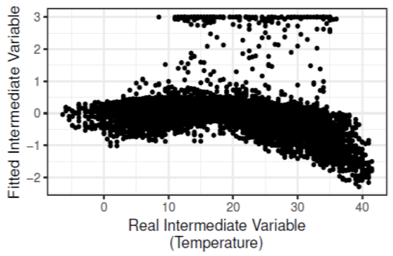


Figure 8: Temperature Against Fitted Intermediate Variable.

Stock Market Dataset

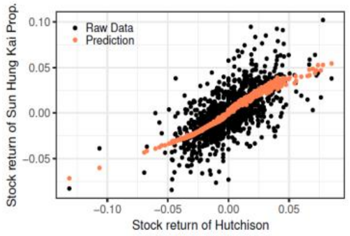
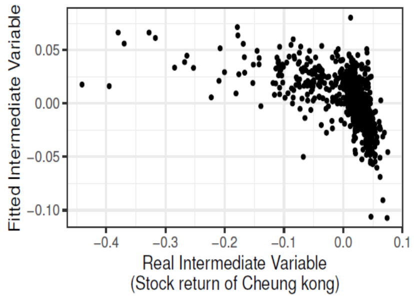


Figure 9: Stock return of Hutchison Against Stock return of Sun Hung Kai Prop.



Cai R, Qiao J, Zhang K, et al. *Causal discovery with cascade nonlinear additive noise models*. IJCAI, 2019.

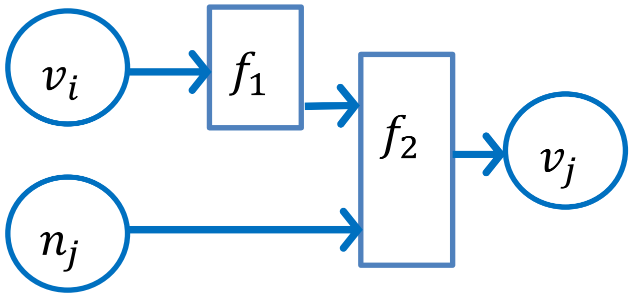
Causal Function based method: PNL

☑ LiNGAM algorithms can only solve linear problems. For non-linear problems,Zhang et al. proposed PNL, the post-NonLinear method.

☑ In the PNL model, assuming that there is a causal relationship $v_i \to v_j$, it can be expressed as

$$v_j = f_2(f_1(v_i) + n_j) \quad (1)$$

- v_i and n_j are independent of each other
- f_1 is an unconstant smooth function
- f_2 is a reversible smooth function and $f'_2 \neq 0$



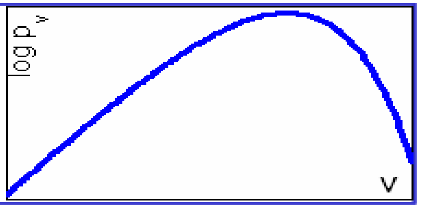
Zhang K, Hyvarinen A. *On the identifiability of the post-nonlinear causal model*. UAI, 2009.

PNL: All non-identifiable cases

☑ Causal direction is generally identifiable if the data were generated according to

$$X_2 = f_2(f_1(X_1) + E).$$

$(\log p_v)' \to c \ (c \neq 0)$,
as $v \to -\infty$ or as $v \to +\infty$



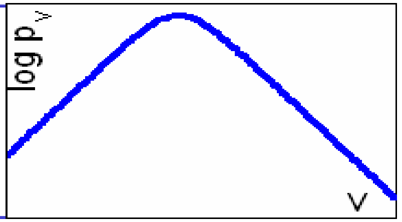
Log-mixed-linear-and-exponential:
$\log p_v = c_1 e^{c_2 v} + c_3 v + c_4$

Table 1: All situations in which the PNL causal model is not identifiable.

	p_{e_2}	p_{t_1} $(t_1 = g_2^{-1}(x_1))$	$h = f_1 \circ g_2$	Remark
I	Gaussian	Gaussian	linear	h_1 also linear
II	log-mix-lin-exp	log-mix-lin-exp	linear	h_1 strictly monotonic, and $h_1' \to 0$, as $z_2 \to +\infty$ or as $z_2 \to -\infty$
III	log-mix-lin-exp	one-sided asymptotically exponential (but not log-mix-lin-exp)	h strictly monotonic, and $h' \to 0$, as $t_1 \to +\infty$ or as $t_1 \to -\infty$	—
IV	log-mix-lin-exp	generalized mixture of two exponentials	Same as above	—
V	generalized mixture of two exponentials	two-sided asymptotically exponential	Same as above	—

$p_v \propto (c_1 e^{c_2 v} + c_3 e^{c_4 v})^{c_5}$

$(\log p_v)' \to c_1 \ (c_1 \neq 0)$,
as $v \to -\infty$ and
$(\log p_v)' \to c_2 \ (c_2 \neq 0)$,
as $v \to +\infty$



Zhang K, Hyvarinen A. *On the identifiability of the post-nonlinear causal model*. UAI, 2009.

Causal Function based methods

☑ If we add new assumptions to the PNL model, we will get the LiNGAM and ANM.

PNL

$x_j = f_2(f_1(x_i) + n_j)$

x_i and n_j is independent

f_1 and f_2 is linear

n_j is non-Gaussian

→

LiNGAM

$x_j = b_{ji} x_i + n_j$

x_i and n_j is independent

b_{ji} is the connection weight of x_j and x_i

f_1 is non-linear

f_2 is linear

→

ANM

$x_j = f(x_i) + n_j$

x_i and n_j is independent

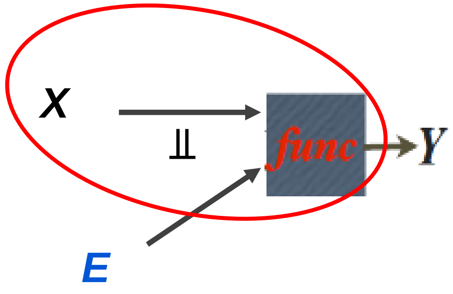
Zhang K, Hyvarinen A. *On the identifiability of the post-nonlinear causal model*. UAI, 2009.

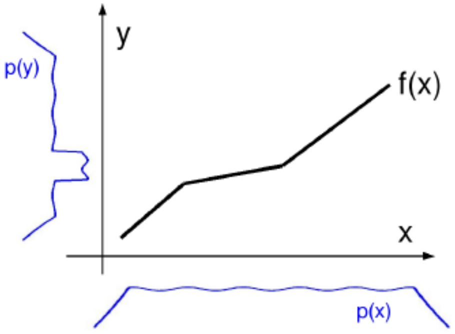
Causal Function based methods: in noiseless case

☑ Problem: how to infer whether $Y = f(X)$ or $X = f^{-1}(X)$ is the right causal model?

☑ An example of independent mechanism

$$Y = f(X), \qquad X \perp\!\!\!\perp f(X)$$





☑ Asymmetry holds

- In the causal direction: independence between X and $f(X)$ holds
- In the anti-causal direction: independence between Y and $f(Y)$ does not hold

Zhang K, Hyvarinen A. *On the identifiability of the post-nonlinear causal model*. UAI, 2009.

IGCI: inference rule

☑ If $X \rightarrow Y$ then

$$\int \log|f'(x)|\, p(x)dx \leq \int \log\left|f^{-1'}(y)\right| p(y)dy$$

☑ empirical estimator

$$\hat{C}_{X\rightarrow Y} := \frac{1}{m}\sum_{j=1}^{m} \log\left|\frac{y_{j+1}-y_j}{x_{j+1}-x_j}\right| \approx \int \log|f'(x)|\, p(x)dx$$

☑ infer $X \rightarrow Y$ whenever

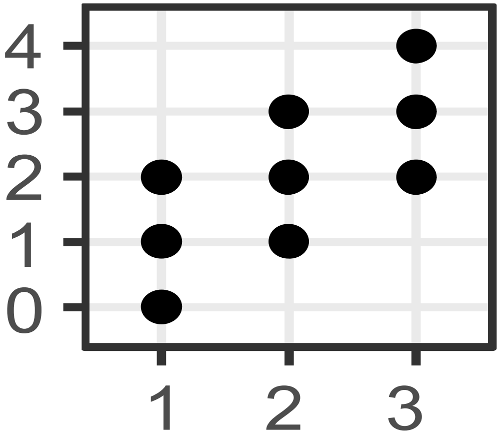
$$\hat{C}_{X\rightarrow Y} < \hat{C}_{Y\rightarrow X}$$

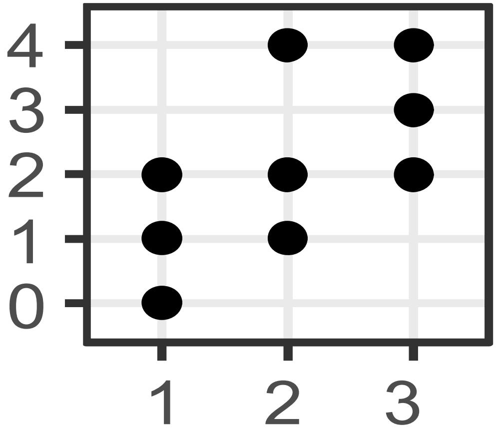
Janzing D, Mooij J, Zhang K, et al. *Information-geometric approach to inferring causal directions*. Artificial Intelligence, 2012.

Causal Function based method: practical issue - categorical variables

☑ Additive noise model for categorical variables

$$Y = g(X) + E, X \perp E$$



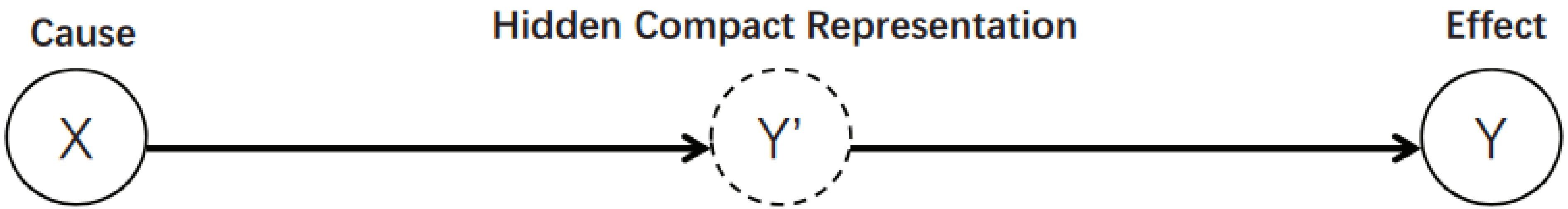


☑ Problem of existing method: how to fit the discrete data? e.g. male, female, wood, iron, steel…

Cai R, Qiao J, Zhang K, et al. *Causal discovery from discrete data using hidden compact representation*. NeurIPS, 2018.

Causal Function based method: HCR

☑ Categorical data: A Hidden Compact Representation Model



Stage 1: deterministic mapping
from cause (X) to hidden representation (Y')

Stage 2: probabilistic mapping
from hidden representation (Y') to effect (Y)

X \ Y'	Poisonous	Not Poisonous
Poisonous Mushroom	1	0
Mushroom	0	1
Rice	0	1
Poisonous Fish	1	0

Y' \ Y	Food Poisoning	Stomach Flu	Normal
Poisonous	0.85	0.10	0.05
Not Poisonous	0.03	0.07	0.90

Cai R, Qiao J, Zhang K, et al. *Causal discovery from discrete data using hidden compact representation*. NeurIPS, 2018.

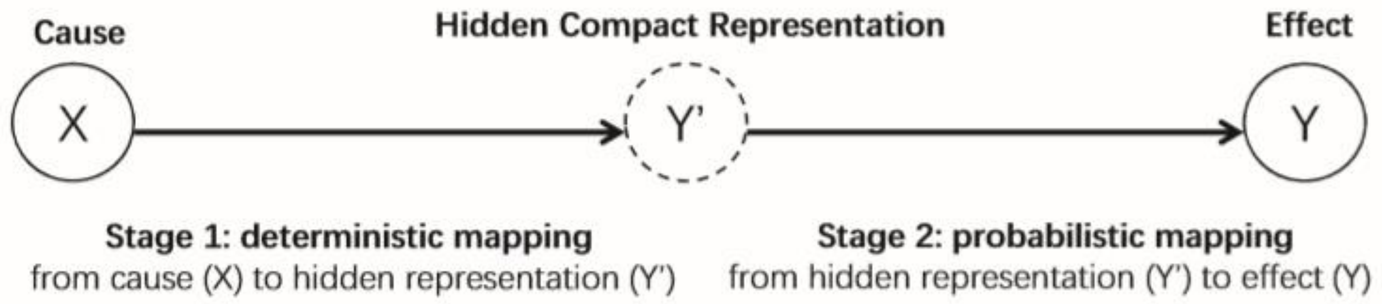
HCR: algorithm

☑ Step 1: Estimate the model $M: X \to Y' \to Y$, $\widehat{M}: Y \to X' \to X$ by maximizing BIC $L^*(M; D), L^*(\widehat{M}; D)$ respectively.

☑ Step 2: If $L^*(M; D) > L^*(\widehat{M}; D)$, infer "$X \to Y$"

If $L^*(M; D) < L^*(\widehat{M}; D)$, infer "$X \to Y$"

If $L^*(M; D) = L^*(\widehat{M}; D)$, infer "$\text{non} - \text{identifiable}$"



$$L^*(M; D) = \prod_{i=1}^{m} P(X = x_i) P(Y = y_i | Y' = f(x_i)) - \frac{d}{2}\log(m)$$

$$= \sum_x n_x \log\left(\frac{n_x}{\sum_x n_x}\right) + \sum_{y'} \sum_y n_{y',y} \log\left(\frac{n_{y',y}}{\sum_y n_{y',y}}\right) - \frac{d}{2}\log(m)$$

HCR: Identifiability

☑ **Theorem 2**. *Assume that in the causal direction there exists the transformation* $Y' = f(X)$ *such that* $P(Y \mid X) = P(Y \mid Y')$, *where* $|Y'| < |X|$, *and assumption A1 holds. Then to produce the same distribution* $P(X, Y)$, ***the reverse direction must involve more effective number of parameters in the model than the causal direction***.

X	$P(X)$
Poisonous Mushroom	0.1
Mushroom	0.2
Rice	0.6
Poisonous Fish	0.1

Y' \ Y	Food Poisoning	Stomach Flu	Normal	
Poisonous Mushroom	0.85	0.10	0.05	Poisonous
Mushroom	0.03	0.07	0.90	Not Poisonous
Rice	0.03	0.07	0.90	Not Poisonous
Poisonous Fish	0.85	0.10	0.05	Poisonous

Y	$P(Y)$
Food Poisoning	0.276
Stomach Flu	0.079
Normal	0.645

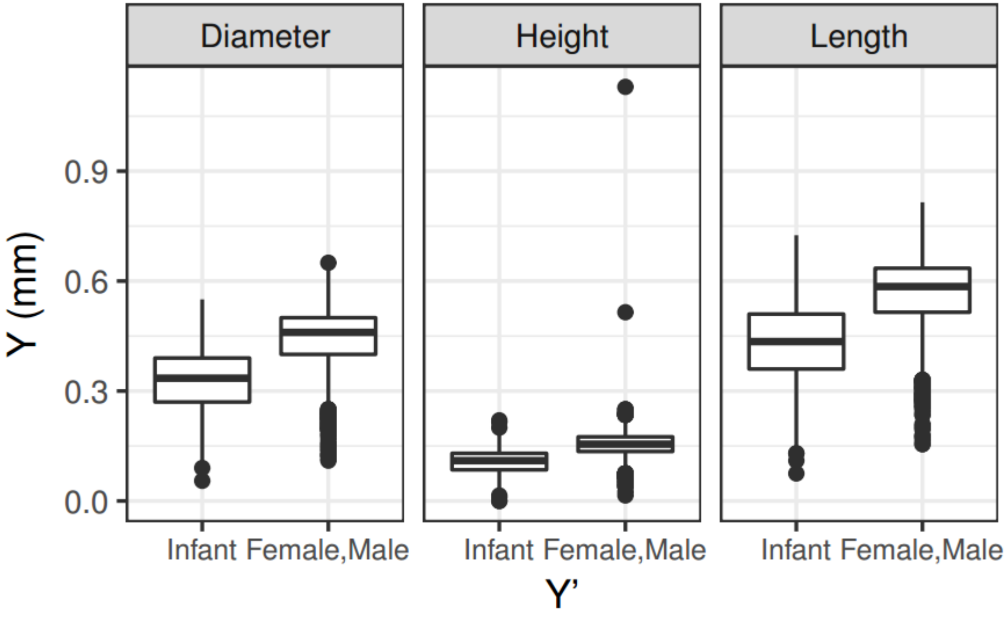
X' \ X	Poisonous Mushroom	Mushroom	Poisonous Fish	
Poisonous Mushroom	0.308	0.127	0.008	Poisonous Mushroom
Mushroom	0.616	0.253	0.016	Mushroom
Rice	0.065	0.532	0.837	Rice
Poisonous Fish	0.011	0.089	0.140	Poisonous Fish

Cai R, Qiao J, Zhang K, et al. *Causal discovery from discrete data using hidden compact representation*. NeurIPS, 2018.

HCR: results

☑ The causal mechanism behind the abalone data set： Adult/Infant → size

Abalone	X	Y'	Y
Sex→Length	Infant	1	0.43 ± 0.1
	Female, Male	2	0.57 ± 0.96
Sex→Diameter	Infant	1	0.33 ± 0.088
	Female, Male	2	0.45 ± 0.079
Sex→Height	Infant	1	0.11 ± 0.032
	Female, Male	2	0.15 ± 0.037



Result on Abalone data set.

Cai R, Qiao J, Zhang K, et al. *Causal discovery from discrete data using hidden compact representation*. NeurIPS, 2018.

经典方法

- ☑ 基于约束的因果发现方法
- ☑ 基于函数的因果发现方法
- ☑ 混合型因果发现方法

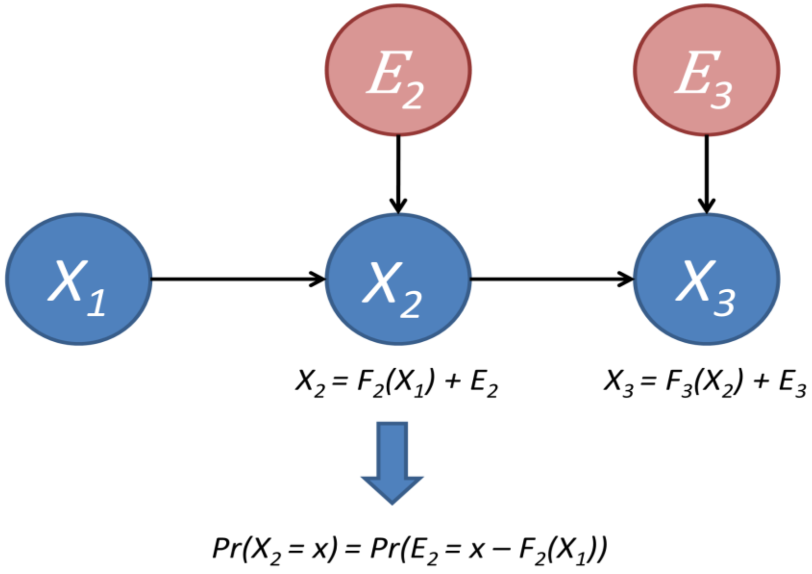
How to handle the high dimensional data?

☑ Motivation: complimentary of Constraints based methods and Causal function based methods

	Causal Function based	Constraints based
High Dimensionality	NO (Pairwise)	Yes
Discovery Ability	Yes	NO (Markov Equivalence)

SELF: model

☑ Embedding the functional causal assumption into the likelihood framework



$$\Pr\left(X_i = o_{j,i} \mid X_{P_i} = o_{j,P_i}\right)$$

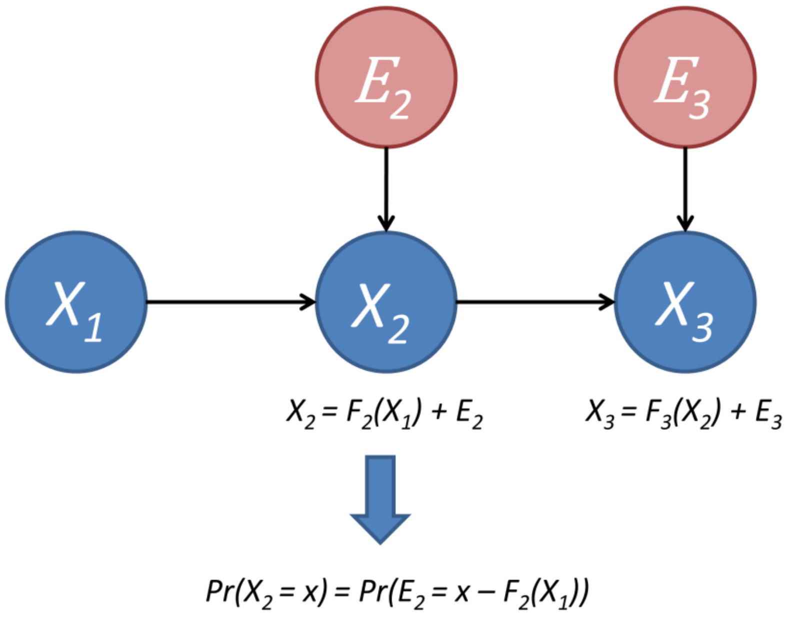
$$\underline{\underline{X_i = F_i(X_{P_i}) + E_i}} \; \Pr\left(E_i = o_{j,i} - F_i(o_{j,P_i}) | X_{P_i}\right)$$

$$\underline{\underline{X_{P_i} \perp\!\!\!\perp E_i}} \; \Pr\left(E_i = o_{j,i} - F_i(o_{j,P_i})\right)$$

Cai R, Qiao J, Zhang Z, et al. *Self: structural equational likelihood framework for causal discovery*. AAAI. 2018.

SELF: framework

☑ Embedding the functional causal assumption into the likelihood framework



Likelihood

$$\mathcal{L}(G;O) = \sum_{j=1}^{m}\sum_{i=1}^{n} \log\left(\Pr(X_i = o_{j,i} \mid X_{P_i} = o_{j,P_i})\right)$$

Structural equation

$$\boxed{\mathcal{L}(S;O) = \sum_{j=1}^{m}\sum_{i=1}^{n} \log\left(\Pr(E_i = o_{j,i} - F_i(o_{j,P_i}))\right)}$$

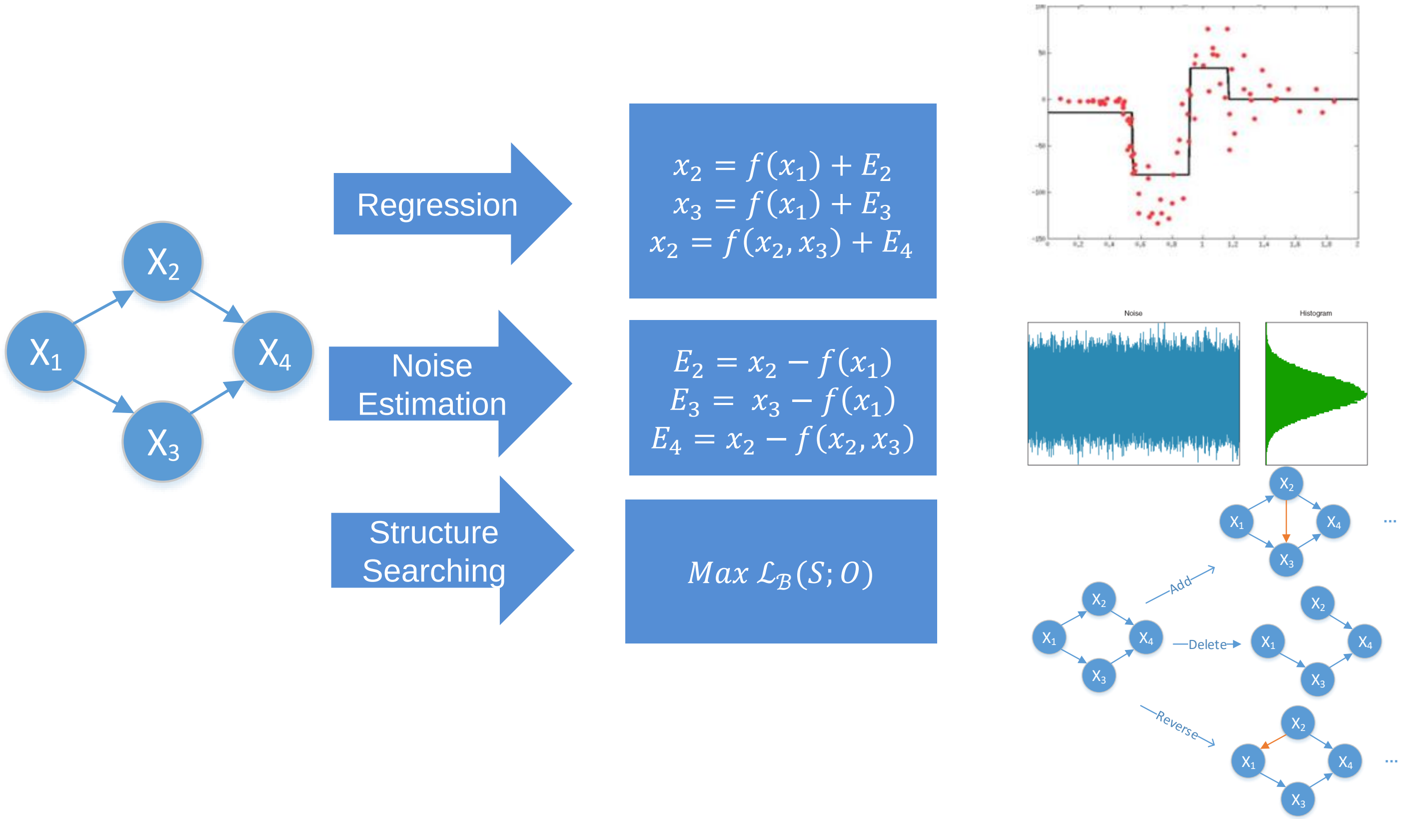
BIC criteria

$$\mathcal{L}_{\mathcal{B}}(S;O) = \sum_{i=1}^{n}\left(\sum_{j=1}^{m} \log(\Pr(E_i = o_{j,i} - F_i(o_{j,P_i}))) - \frac{d_i log(m)}{2}\right)$$

Cai R, Qiao J, Zhang Z, et al. *Self: structural equational likelihood framework for causal discovery*. AAAI. 2018.

SELF: algorithm

- ☑ Step 1: For each nodes, fit F_i with a nonlinear or linear regression (e.g. XGBoost, OLS)
- ☑ Step 2: Estimate the noise distribution with kernel density function
- ☑ Step 3: Maximize



Cai R, Qiao J, Zhang Z, et al. *Self: structural equational likelihood framework for causal discovery*. AAAI. 2018.

SELF: results

☑ SOTA results, out of box

Table 3: Results on real world structure with nonlinear data.

Data	F1		Recall		Precision	
	SELF	ANM	SELF	ANM	SELF	ANM
Child	**0.71**	0.26	**0.60**	0.40	**0.88**	0.19
Alarm	**0.79**	0.53	**0.74**	0.59	**0.85**	0.48
Win95pts	**0.77**	0.47	**0.71**	0.49	**0.86**	0.45
Pathfinder	**0.88**	0.15	**0.90**	0.08	0.86	**1.00**

Table 2: Results on real world structure with linear data.

Dataset	F1				Recall				Precision			
	SELF	LiNGAM	DLiNGAM	HCBN	SELF	LiNGAM	DLiNGAM	HCBN	SELF	LiNGAM	DLiNGAM	HCBN
Child	**0.98**	**0.98**	0.95	0.58	**1.00**	**1.00**	**1.00**	0.65	**0.96**	0.95	0.92	0.52
Alarm	**0.98**	0.43	0.94	0.52	0.99	0.76	**1.00**	0.64	**0.96**	0.31	0.88	0.44
Win95pts	**0.95**	0.56	0.88	0.80	0.97	0.88	**1.00**	0.91	**0.93**	0.42	0.79	0.71
Pathfinder	**0.91**	0.86	0.85	0.73	0.95	**0.96**	**0.96**	0.83	**0.87**	0.77	0.76	0.64

Cai R, Qiao J, Zhang Z, et al. *Self: structural equational likelihood framework for causal discovery*. AAAI. 2018.
Codes: https://github.com/DMIRLAB-Group/CDMIR

研究进展

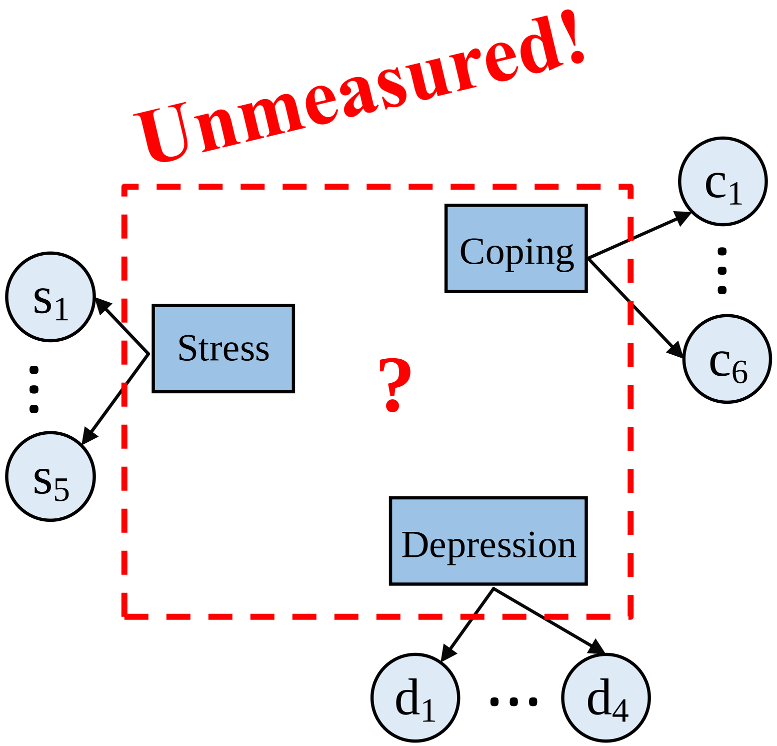
☑ 隐变量问题

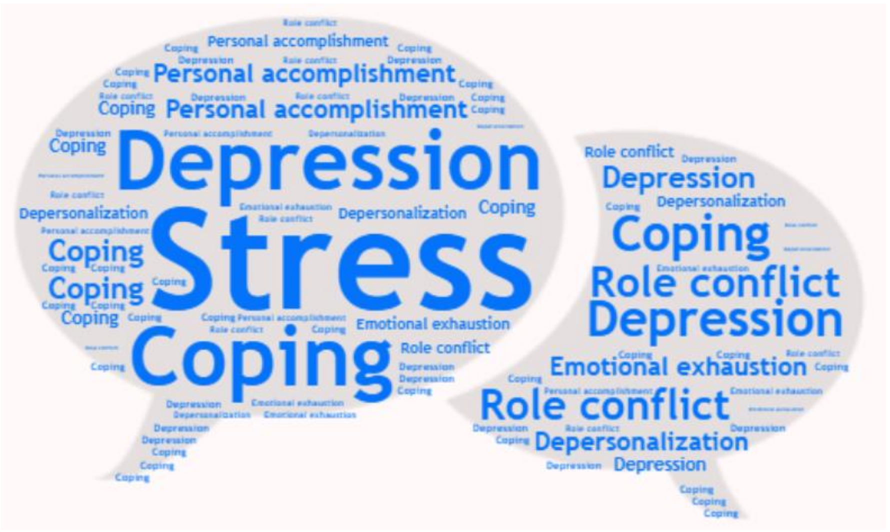
☑ 非独立同分布问题

Causal discovery among latent variables

☑ Discovering causal relations among latent variables is important in many domains

- In psychotherapy: what is the causal relations among role conflict, depersonalization, personal accomplishment?





Q_1: What is the level of stress you are experiencing?

Q_2: How often do you feel depressed?

How to handle latent variable?

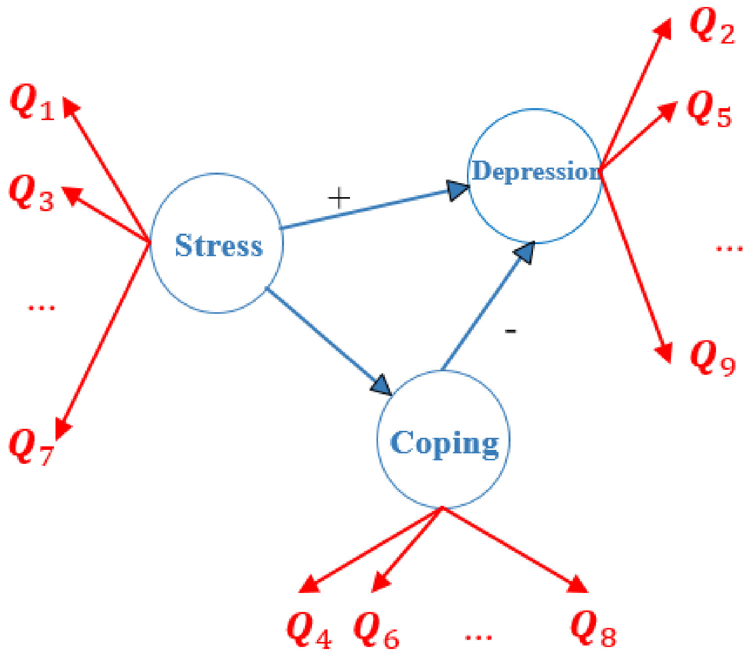
☑ Linear latent variable model

- Measurement model
- Structural model

S_1				
……				
S_{20}.				
C_1				
……				
C_{20}				
D_1				
……				
D_{20}				

Observed data

How to?



Causal structure of latent variables

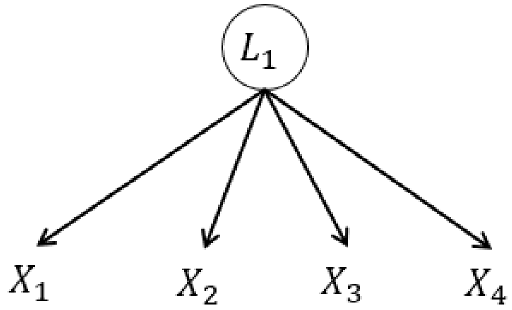
Tetrad Condition

☑**Tetrad Condition**: for X_1, X_2 and any one of X_3, X_4, three quadratic constraints (tetrad constraints) on the covariance matrix are implied: e.g., for X_4,

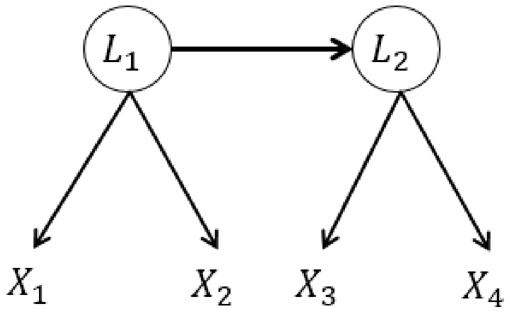
$$\rho_{12}\rho_{34} = \rho_{14}\rho_{23} = \rho_{13}\rho_{24},$$

where ρ_{12} is the correlation between X_1, X_2, etc.

(Note that any two of the three vanishing tetrad differences above entails the third.)



$$\rho_{14}\rho_{23} - \rho_{13}\rho_{24} = 0$$
$$\rho_{12}\rho_{34} - \rho_{13}\rho_{24} = 0$$
$$\rho_{12}\rho_{34} - \rho_{14}\rho_{23} = 0$$



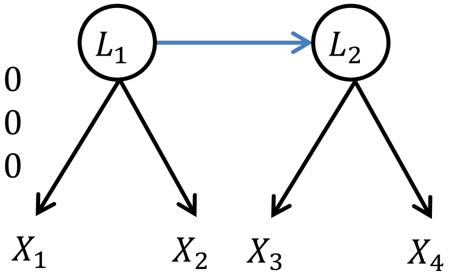
$$\rho_{14}\rho_{23} - \rho_{13}\rho_{24} = 0$$
$$\rho_{12}\rho_{34} - \rho_{13}\rho_{24} \neq 0$$
$$\rho_{12}\rho_{34} - \rho_{14}\rho_{23} \neq 0$$

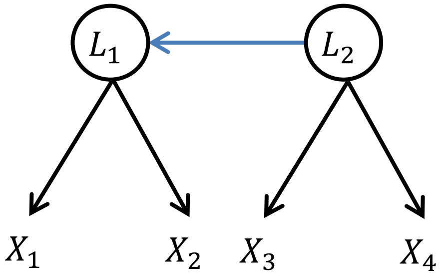
Silva R, Scheines R, Glymour C, Spirtes P, Chickering DM. *Learning the structure of linear latent variable models*, JMLR 2006.

Tetrad Condition: Limitation

☑ Cannot ***identify*** the causal direction: $L_1 \rightarrow L_2$ or $L_2 \rightarrow L_1$?

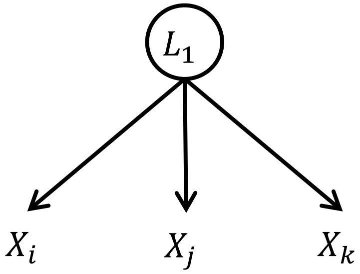
$$\rho_{14}\rho_{23} - \rho_{13}\rho_{24} = 0$$
$$\rho_{12}\rho_{34} - \rho_{13}\rho_{24} \neq 0$$
$$\rho_{12}\rho_{34} - \rho_{14}\rho_{23} \neq 0$$

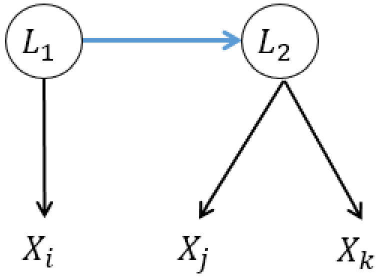


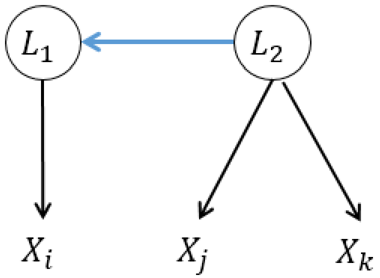


$$\rho_{14}\rho_{23} - \rho_{13}\rho_{24} = 0$$
$$\rho_{12}\rho_{34} - \rho_{13}\rho_{24} \neq 0$$
$$\rho_{12}\rho_{34} - \rho_{14}\rho_{23} \neq 0$$

☑ Cannot detect the latent variables when we only have 3 measured variables







How to solve this problem?

Intuition

☑ What kind of information is helpful to distinguish them?

Second moment information (covariance)

Non-Gaussianity

Higher moment information (independence of noise)

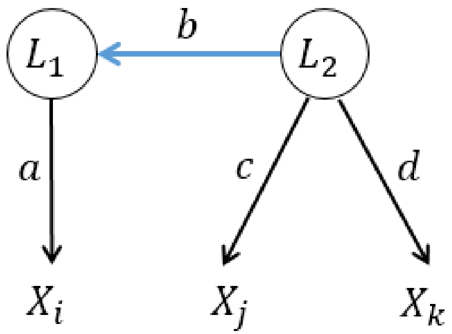
Tetrad Condition

? Condition

Non-Gaussian information may help

Triad: the asymmetry of Triad condition

☑ Consider the following two causal structures:

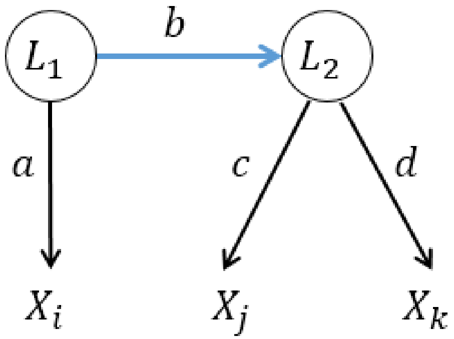


$$X_i = abL_2 + aL_1 + \varepsilon_i$$

$$\boxed{\begin{aligned} Cov(X_i, X_k) &= abd \\ Cov(X_j, X_k) &= cd \\ \frac{Cov(X_i, X_k)}{Cov(X_j, X_k)} &= \frac{ab}{c} \end{aligned}}$$

$$\left(X_i - \frac{Cov(X_i, X_k)}{Cov(X_j, X_k)} X_j\right) \perp X_k$$

Asymmetry



$$X_i = aL_1 + \varepsilon_i$$

$$\boxed{\begin{aligned} Cov(X_i, X_k) &= abd \\ Cov(X_j, X_k) &= (b^2+1)cd \\ \frac{Cov(X_i, X_k)}{Cov(X_j, X_k)} &= \frac{ab}{(b^2+1)c} \end{aligned}}$$

$$\left(X_i - \frac{Cov(X_i, X_k)}{Cov(X_j, X_k)} X_j\right) \not\perp X_k$$

Cai, R., Xie, F. et al. *Triad constraints for learning causal structure of latent variables*. NeurIPS 2019.

Triad condition

☑ **Triad Constraints (our proposed)** In a linear latent model, suppose $\{X_i, X_j\}$ and X_k are distinct and correlated variables and that all noise variables are non-Gaussian. Define the pseudo-residual of $\{X_i, X_j\}$ relative to X_k , which is called a reference variable, as

$$E_{(i,j|k)} := X_i - \frac{Cov(X_i, X_k)}{Cov(X_j, X_k)} \cdot X_j.$$

☑ We say that $\{X_i, X_j\}$ and X_k satisfy Triad constraint if and only if $E_{(i,j|k)} \perp X_k$, i.e., $\{X_i, X_j\}$ and X_k violate the Triad constraint if and only if $E_{(i,j|k)} \perp X_k$.

Residual

R1= $X_i - \boxed{\frac{Cov(X_i, X_j)}{Cov(X_j, X_j)}} X_j$

"normal-residual"

VS

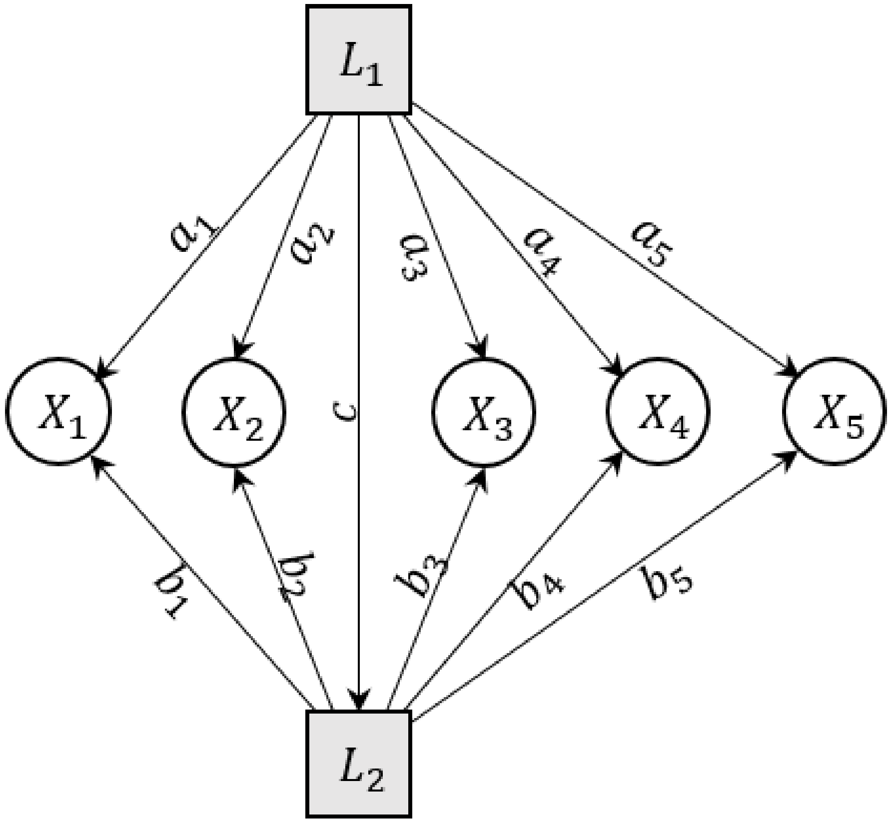
R2= $X_i - \boxed{\frac{Cov(X_i, X_k)}{Cov(X_j, X_k)}} X_j$

"pseudo-residual"

Cai, R., Xie, F. et al. *Triad constraints for learning causal structure of latent variables*. NeurIPS 2019.

Triad Condition: Limitation

☑ Cannot applied into the multiple shared latent variables case:



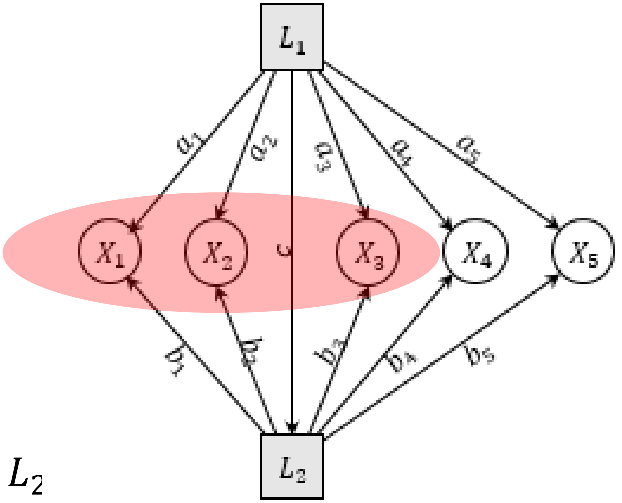
☑ Can we extend the surrogate strategy to multiple shared latent variables?

Generalized Independent Noise (GIN) Condition

Consider $\boldsymbol{Y} = \{X_1, X_2, X_3\}$, we have

$$\underbrace{\begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix}}_{\mathbf{Y}} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} + \underbrace{\begin{bmatrix} \varepsilon_{X_1} \\ \varepsilon_{X_2} \\ \varepsilon_{X_3} \end{bmatrix}}_{\mathbf{E_Y}}$$

$\boldsymbol{E}_{\mathbf{Y}}$ is independent from L_1 and L_2.



However, we don't have access to L_1 and L_2

Can we still use some measured variables as surrogate variables?

Similarly, use Measured Variables $\boldsymbol{Z} = (X_4, X_5)^{\mathrm{T}}$ as Surrogate Variables, we have

$$\begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} \cdot \frac{1}{a_5 b_4 - a_4 b_5} \begin{bmatrix} b_5 & b_4 \\ a_5 & a_4 \end{bmatrix} \begin{bmatrix} X_4 \\ X_5 \end{bmatrix} - \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} \cdot \frac{1}{a_5 b_4 - a_4 b_5} \begin{bmatrix} b_5 & b_4 \\ a_5 & a_4 \end{bmatrix} \begin{bmatrix} \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} + \begin{bmatrix} \varepsilon_{X_1} \\ \varepsilon_{X_2} \\ \varepsilon_{X_3} \end{bmatrix}$$

$\exists$ nonzero vector $\boldsymbol{\omega}$ s.t. $\boldsymbol{\omega}^{\mathrm{T}} \mathbf{Cov}(\boldsymbol{Y}, \boldsymbol{Z}) = \mathbf{0}$

$$\boldsymbol{\omega} = [a_2 b_3 - b_2 a_3, b_1 a_3 - a_1 b_3, a_1 b_2 - b_1 a_2]^T$$

$\Rightarrow \boldsymbol{\omega}^{\mathrm{T}} \boldsymbol{Y}$ is independent from L.

Xie, F., **Cai, R.**, Huang, B., Glymour, C., Hao, Z., & Zhang, K. *Generalized Independent Noise Condition for Estimating Latent Variable Causal Graphs*. NeurIPS 2020.

GIN: Generalized Independent Noise Condition

Consider $\boldsymbol{Y} = \{X_1, X_2, X_3\}$, we have $\begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} + \begin{bmatrix} \varepsilon_{X_1} \\ \varepsilon_{X_2} \\ \varepsilon_{X_3} \end{bmatrix}$ $\Rightarrow \boldsymbol{Y} = \boldsymbol{AL} + \boldsymbol{E} \Rightarrow \boldsymbol{E} = \boldsymbol{Y} - \boldsymbol{AL}$

$\exists$ nonzero vector $\boldsymbol{\omega}$ s.t. $\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{A} = \mathbf{0}$ $\Rightarrow$ $\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{Y} = \boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{E}$ is independent from L.

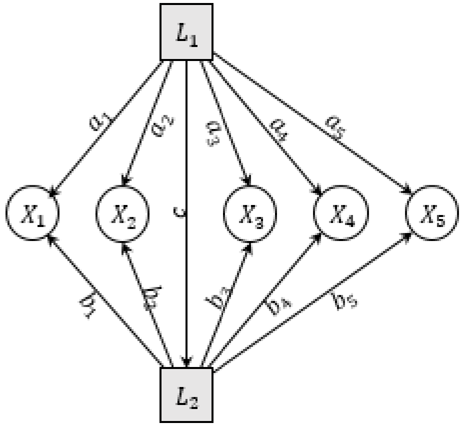
$\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{A} = \mathbf{0}$ $\Leftrightarrow$ $\boldsymbol{\omega}^{\mathrm{T}} \cdot \mathbf{Cov}(\boldsymbol{Y}, \boldsymbol{L}_{1,2}) = \mathbf{0}$

$\boldsymbol{\omega}^{\mathrm{T}} \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} = \mathbf{0}$ $\Leftrightarrow$ $\boldsymbol{\omega}^{\mathrm{T}} \cdot \mathbf{Cov}(\boldsymbol{Y}, \boldsymbol{L}_{1,2}) = \mathbf{0}$

use $\boldsymbol{Z} = (X_4, X_5)^{\mathrm{T}}$ instead $\boldsymbol{L}_{1,2}$

$\boldsymbol{\omega}^{\mathrm{T}} \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix} \begin{bmatrix} a_4 & a_5 \\ b_4 & b_5 \end{bmatrix} = \mathbf{0}$ $\Leftrightarrow$ $\boldsymbol{\omega}^{\mathrm{T}} \cdot \mathbf{Cov}(\boldsymbol{Y}, \boldsymbol{L}_{1,2})\mathbf{Cov}(\boldsymbol{L}_{1,2}, \boldsymbol{Z}) = \mathbf{0}$

$\boldsymbol{\omega}^{\mathrm{T}} \begin{bmatrix} a_1a_4 + b_1b_4 & a_1a_5 + b_1b_5 \\ a_2a_4 + b_2b_4 & a_2a_5 + b_2b_5 \\ a_3a_4 + b_3b_4 & a_3a_5 + b_3b_5 \end{bmatrix} = \mathbf{0}$ $\Leftrightarrow$ $\boldsymbol{\omega}^{\mathrm{T}} \cdot \mathbf{Cov}(\boldsymbol{Y}, \boldsymbol{Z}) = \mathbf{0}$



$\exists$ nonzero vector $\boldsymbol{\omega}$ s.t. $\boldsymbol{\omega}^{\mathrm{T}} \cdot \mathbf{Cov}(\boldsymbol{Y}, \boldsymbol{Z}) = \mathbf{0}$ $\Rightarrow \omega^{\mathrm{T}}\boldsymbol{Y}$ is independent from L.

$\boldsymbol{\omega} = [a_2b_3 - b_2a_3, b_1a_3 - a_1b_3, a_1b_2 - b_1a_2]^T$

GIN: view measured and latent variables in a unified way

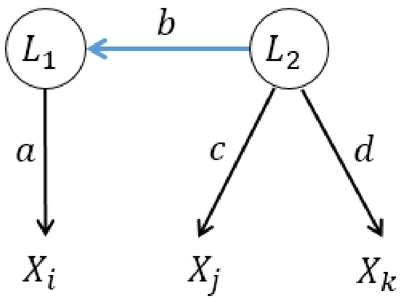
Xie, F., **Cai, R.**, Huang, B., Glymour, C., Hao, Z., & Zhang, K. *Generalized Independent Noise Condition for Estimating Latent Variable Causal Graphs*. NeurIPS 2020.

GIN: Generalized Independent Noise Condition

☑ **Generalized Independent Noise(GIN) condition**： $(\boldsymbol{Z}, \boldsymbol{Y})$ follows the GIN condition iff there exists non-zeros ω such that $\omega^{\mathrm{T}}\mathbb{E}[\boldsymbol{Y}\boldsymbol{Z}^{\mathrm{T}}] = 0$ and $\omega^{\mathrm{T}}\boldsymbol{Y}$ is independent from $\boldsymbol{Z}$**.**

☑ Triad condition can be seen as a special case of the GIN condition.

- For example, $(\{X_k\}, \{X_i, X_j\})$ satisfy Triad condition iff $(\{X_k\}, \{X_i, X_j\})$ satisfy GIN condition



$$\left(X_i - \frac{Cov(X_i, X_k)}{Cov(X_j, X_k)} X_j\right) \perp X_k$$

Triad condition

$\exists$ nonzero vector $\boldsymbol{\omega}$ s.t. $\boldsymbol{\omega}^{\mathrm{T}}\mathbf{Cov}\left(\{X_i, X_j\}, \{X_k\}\right) = \mathbf{0}$

$$\boldsymbol{\omega} = [cd, -abd]^T$$

$\Rightarrow\ \boldsymbol{\omega}^{\mathrm{T}}\begin{bmatrix} X_i \\ X_j \end{bmatrix}$ is independent from L_2.

GIN condition

Xie, F., **Cai, R.**, Huang, B., Glymour, C., Hao, Z., & Zhang, K. *Generalized Independent Noise Condition for Estimating Latent Variable Causal Graphs*. NeurIPS 2020.

GIN: Application in LiNGLaM

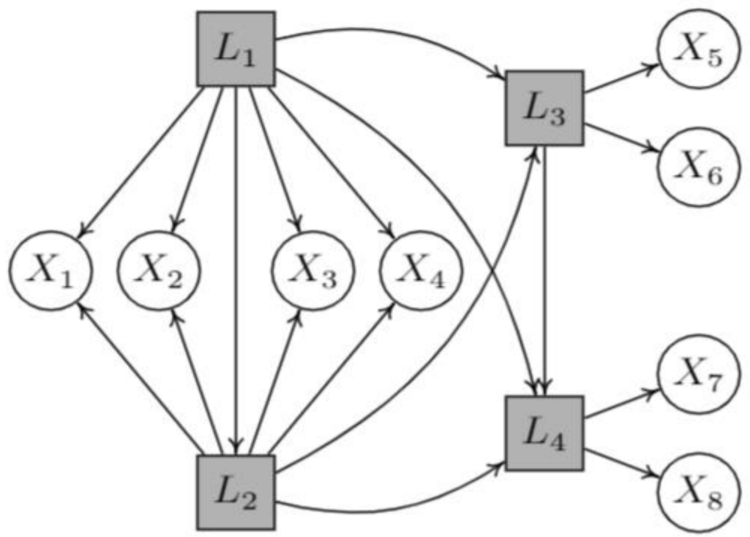
Linear Non-Gaussian Latent Variable Model (LiNGLaM)

☑ A1. [**Measurement Assumption**] There is no observed variable in $\mathbf{X}$ being an ancestor of any latent variables in $\mathbf{L}$.

☑ A2. [**Non-Gaussianity Assumption**] The noise terms are non-Gaussian.

☑ A3. [**Double-Pure Child Variable Assumption**] Each latent variable set $\mathbf{L}'$, in which every latent variable directly causes the same set of observed variables, has at least $2\mathrm{Dim}(\mathbf{L}')$ pure measurement variables as children.

☑ A4. [**Purity Assumption**] There is no direct edge between observed variables.



A simple structure that satisfies LiNGLaM

Xie, F., **Cai, R.**, Huang, B., Glymour, C., Hao, Z., & Zhang, K. *Generalized Independent Noise Condition for Estimating Latent Variable Causal Graphs*. NeurIPS 2020.

GIN: Application in LiNGLaM

☑ We proposed a *two-steps* algorithm.

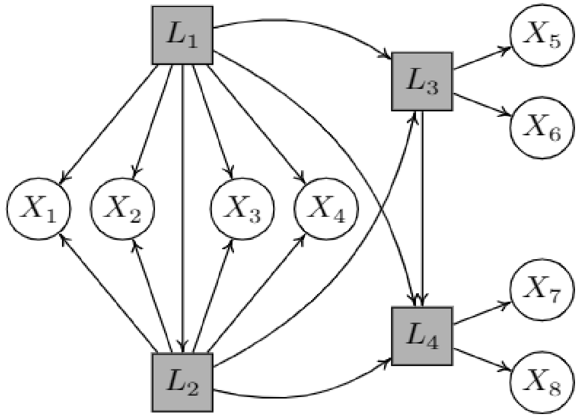
- Step 1: find *causal clusters* (variables sharing the same latent variables as parents);
- Step 2: determine *causal order* of the latent variables;
- Estimate the coefficients if needed

Xie, F., **Cai, R.**, Huang, B., Glymour, C., Hao, Z., & Zhang, K. *Generalized Independent Noise Condition for Estimating Latent Variable Causal Graphs*. NeurIPS 2020.

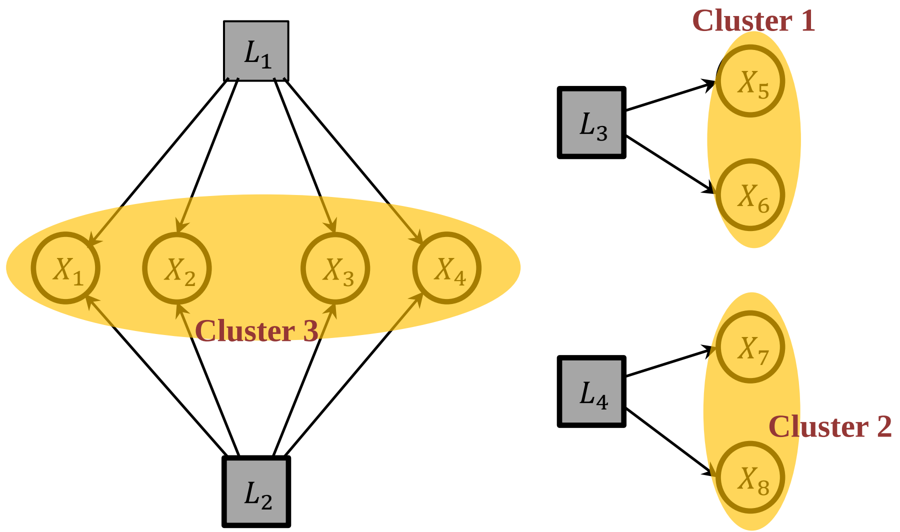
GIN: Application in LiNGLaM

☑ We proposed a *two-steps* algorithm.

- Step 1: find *causal clusters* (variables sharing the same latent variables as parents);
- Step 2: determine *causal order* of the latent variables;
- Estimate the coefficients if needed



Ground-truth graph



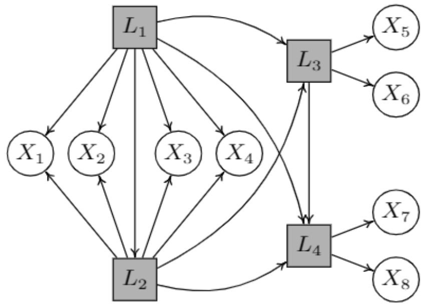
E.g., Test |latent|=1, we have
$(\{X_1, \dots, X_4, X_7, X_8\}, \{X_5, X_6\})$ satisfies GIN. Thus, $\{X_5, X_6\}$ is a cluster.
Similarly,
$(\{X_1, \dots, X_4, X_5, X_6\}, \{X_7, X_8\})$ satisfies GIN. Thus, $\{X_7, X_8\}$ is a cluster.

The variables in a cluster share the same GIN conditions.

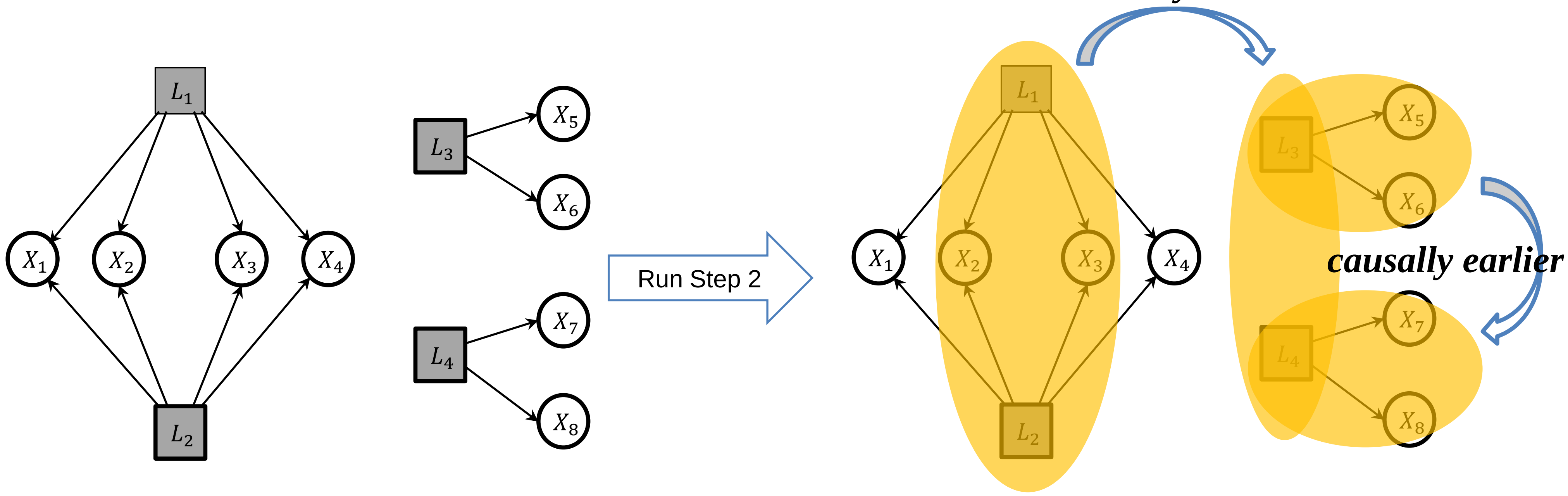
GIN: Application in LiNGLaM

☑ We proposed a *two-steps* algorithm.

- Step 1: find *causal clusters* (variables sharing the same latent variables as parents);
- Step 2: determine *causal order* of the latent variables;
- Estimate the coefficients if needed



Ground-truth graph



E.g.,
($\{X_3, X_4, X_5\}, \{X_1, X_2\}$) satisfies GIN, and
($\{X_3, X_4, X_7\}, \{X_1, X_2\}$) satisfies GIN.
We have $\{L_1, L_2\}$ is ***causally earlier*** than L_3 and L_4.
Similarly, we have L_3 is causally earlier than L_4.

GIN: Application in LiNGLaM

- ☑ We simulate data following the LiNGLaM, including 4 cases, with different DAG structures for and measurement variables and latent variables.
- ☑ Goal: find clusters (determine the location of latent variables)?
  - ***Latent oimission***: measure omitted latent variables
  - ***Latent commission***: measure falsely detected latent variables
  - ***Mismeasurements***: measure the misclassification of observed variables

The lower the better

		Latent omission				Latent commission				Mismeasurements			
Algorithm		GIN	LSTC	FOFC	BPC	GIN	LSTC	FOFC	BPC	GIN	LSTC	FOFC	BPC
Case 1	500	0.00(0)	0.00(0)	1.00(10)	0.50(10)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)
	1000	0.00(0)	0.00(0)	1.00(10)	0.50(10)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)
	2000	0.00(0)	0.00(0)	1.00(10)	0.50(10)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)
Case 2	500	0.10(2)	0.20(4)	0.9(10)	0.50(10)	0.00(0)	0.05(1)	0.00(0)	0.00(0)	0.12(2)	0.12(4)	0.00(0)	0.20(10)
	1000	0.05(1)	0.15(3)	1.00(10)	0.50(10)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.04(1)	0.12(3)	0.00(0)	0.20(10)
	2000	0.00(0)	0.00(0)	1.00(10)	0.50(10)	0.00(0)	0.02(2)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.20(10)
Case 3	500	0.20(3)	0.20(3)	0.13(9)	0.10(1)	0.00(0)	0.03(3)	0.00(0)	0.00(0)	0.19(3)	0.17(3)	0.00(0)	0.00(0)
	1000	0.06(2)	0.13(2)	0.16(10)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.06(2)	0.00(0)	0.00(0)	0.00(0)
	2000	0.00(0)	0.00(0)	0.50(10)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.00(0)
Case 4	500	0.13(4)	0.40(6)	0.90(10)	0.63(10)	0.00(0)	0.23(5)	0.00(0)	0.00(0)	0.04(2)	0.15(6)	0.02(2)	0.06(4)
	1000	0.10(3)	0.26(6)	0.93(10)	0.66(10)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.05(3)	0.11(2)	0.01(1)	0.02(2)
	2000	0.03(1)	0.32(6)	1.00(10)	0.70(10)	0.00(0)	0.00(0)	0.00(0)	0.00(0)	0.04(1)	0.11(3)	0.00(10)	0.00(0)

Note: The number in parentheses indicates the number of occurrences that the current algorithm *cannot* correctly solve the problem.

Our proposed algorithm is more efficient and can find all latent variables!

GIN: Results

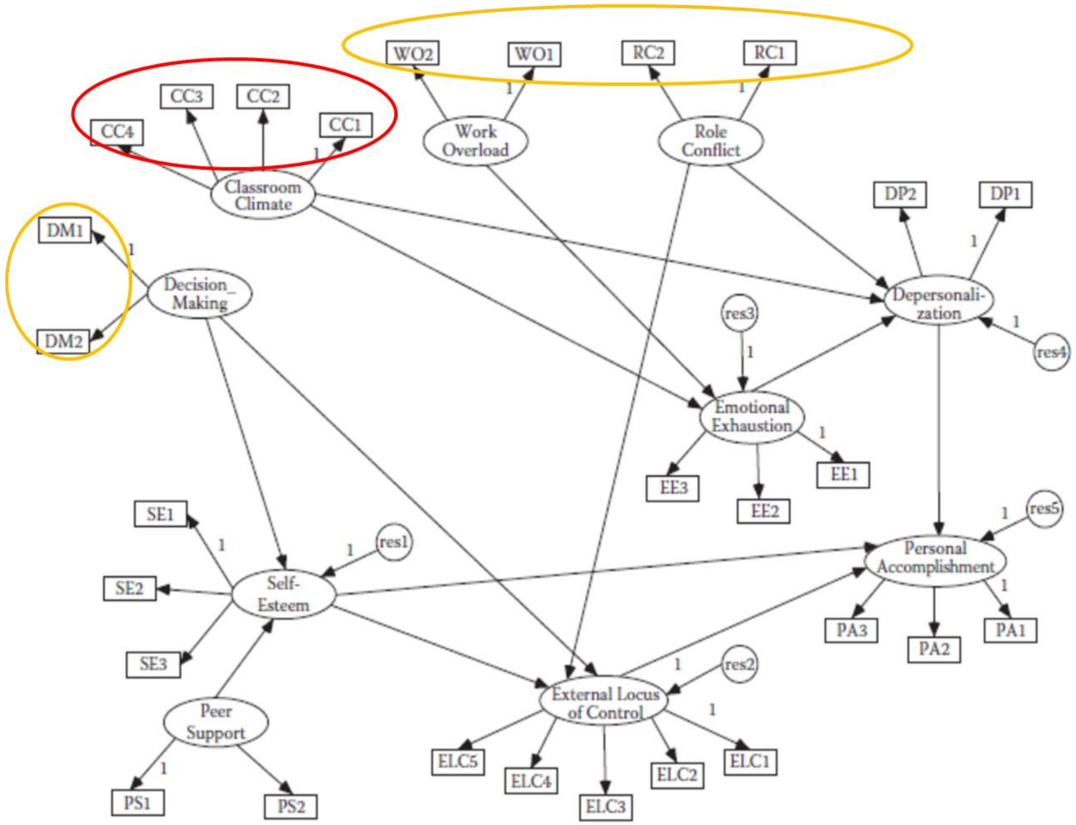
☑ We apply our algorithm to discover the underlying causal structure behind the Teacher's Burnout, and we are developing a tool with the help of psychologist.

- Discovered clusters and causal order of the latent variables:

Causal Clusters	Observed variables
$\mathcal{S}_1$ (1)	$RC_1, RC_2, WO_1, WO_2, DM_1, DM_2$
$\mathcal{S}_2$ (1)	CC_1, CC_2, CC_3, CC_4
$\mathcal{S}_3$ (1)	PS_1, PS_2
$\mathcal{S}_4$ (1)	$ELC_1, ELC_2, ELC_3, ELC_4, ELC_5$
$\mathcal{S}_5$ (2)	$SE_1, SE_2, SE_3, EE_1, EE_2, EE_3, DP_1, PA_3$
$\mathcal{S}_6$ (3)	DP_2, PA_1, PA_2

$$L(\mathcal{S}_1) \succ L(\mathcal{S}_2) \succ L(\mathcal{S}_3) \succ L(\mathcal{S}_5) \succ L(\mathcal{S}_4) \succ L(\mathcal{S}_6).$$

(from root to leaf)

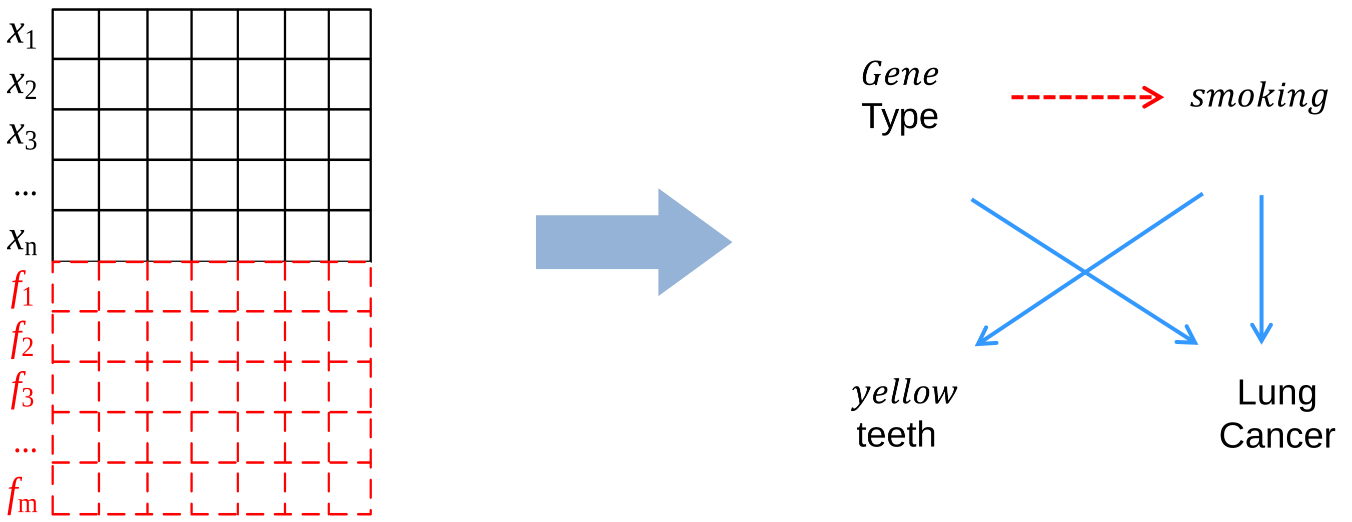


Hypothesized model by experts [Byrne, 2010]

Xie, F., **Cai, R.**, Huang, B., Glymour, C., Hao, Z., & Zhang, K. *Generalized Independent Noise Condition for Estimating Latent Variable Causal Graphs*. NeurIPS 2020.

FRITL: More general setting

☑ How to learn causal structure with latent variables from observed data?



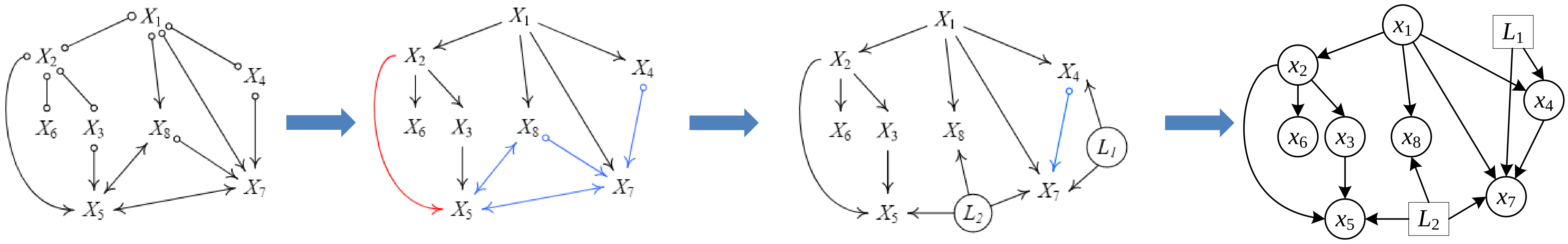
☑ **Challenges:**

- how to efficiently decompose a large global graph into local small structures *without introducing new latent confounders*?
- how to *recover local structures accurately* in the presence of latent confounders?

Chen W, Zhang K, **Cai R**, et al. *FRITL: A Hybrid Method for Causal Discovery in the Presence of Latent Confounders*. Submitted to JMLR, arXiv preprint arXiv:2103.14238, 2021.

FRITL: Algorithm

- ☑ Step 1: Construct a PAG: based on conditional independence test
- ☑ Step 2: Infer local causal structures: IN condition
- ☑ Step 3: Detect shared latent confounders
- ☑ Estimate remaining undetermined local causal structures if needed



Chen W, Zhang K, **Cai R**, et al. *FRITL: A Hybrid Method for Causal Discovery in the Presence of Latent Confounders*. Submitted to JMLR, arXiv preprint arXiv:2103.14238, 2021.

前沿进展

☑ 隐变量问题

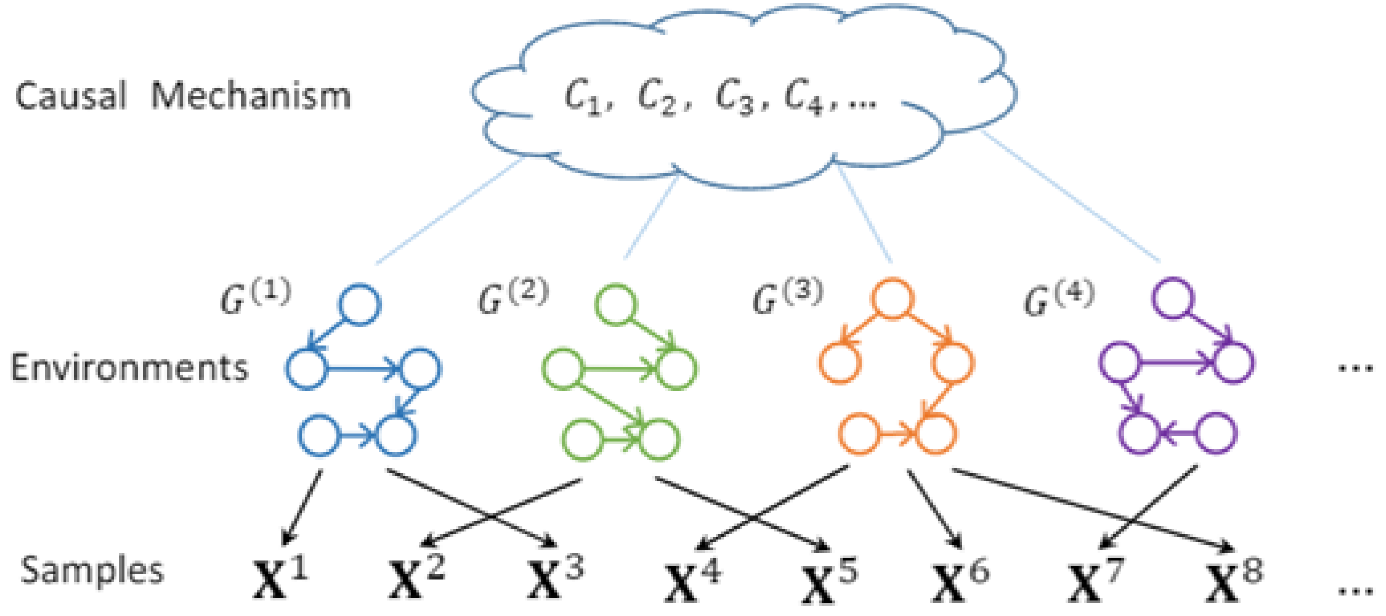
☑ 非独立同分布问题

- Multi-domain
- Multi closely related domain

How about data from multiple environments?

☑ If data are generated from multiple environments, how to discover the causal structures from data?

☑ Data is non-i.i.d.

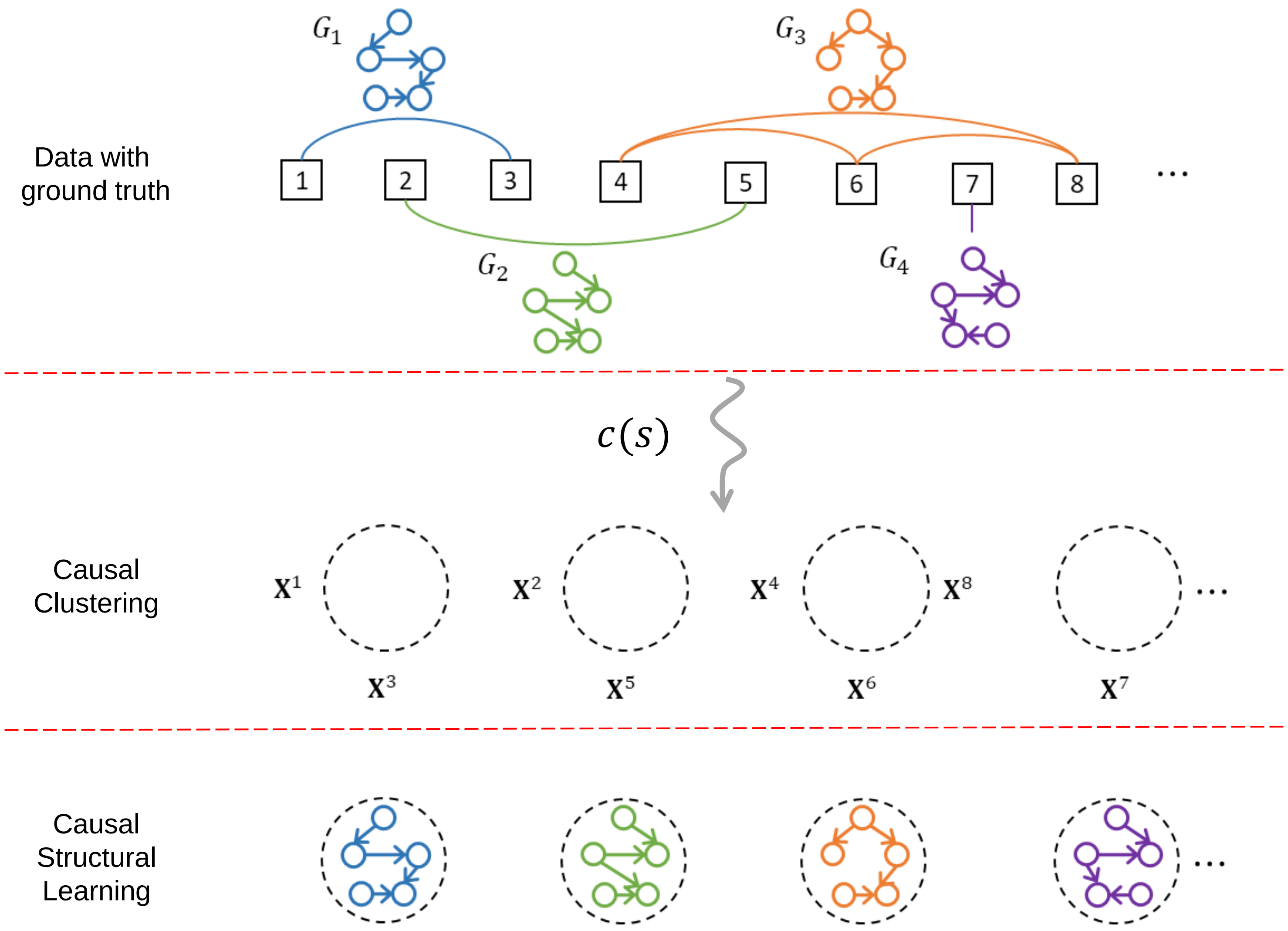


☑ Challenges:

- how to group the subjects that are implied the same causal structure?
- how to recover the causal structure?

Chen, W., Wu, Y., **Cai, R.**, et al. *CCSL: A Causal Structure Learning Method from Multiple Unknown Environments*. arXiv, 2022.

CCSL: Causal Clustering Structure Learning



Grouping subjects

causality

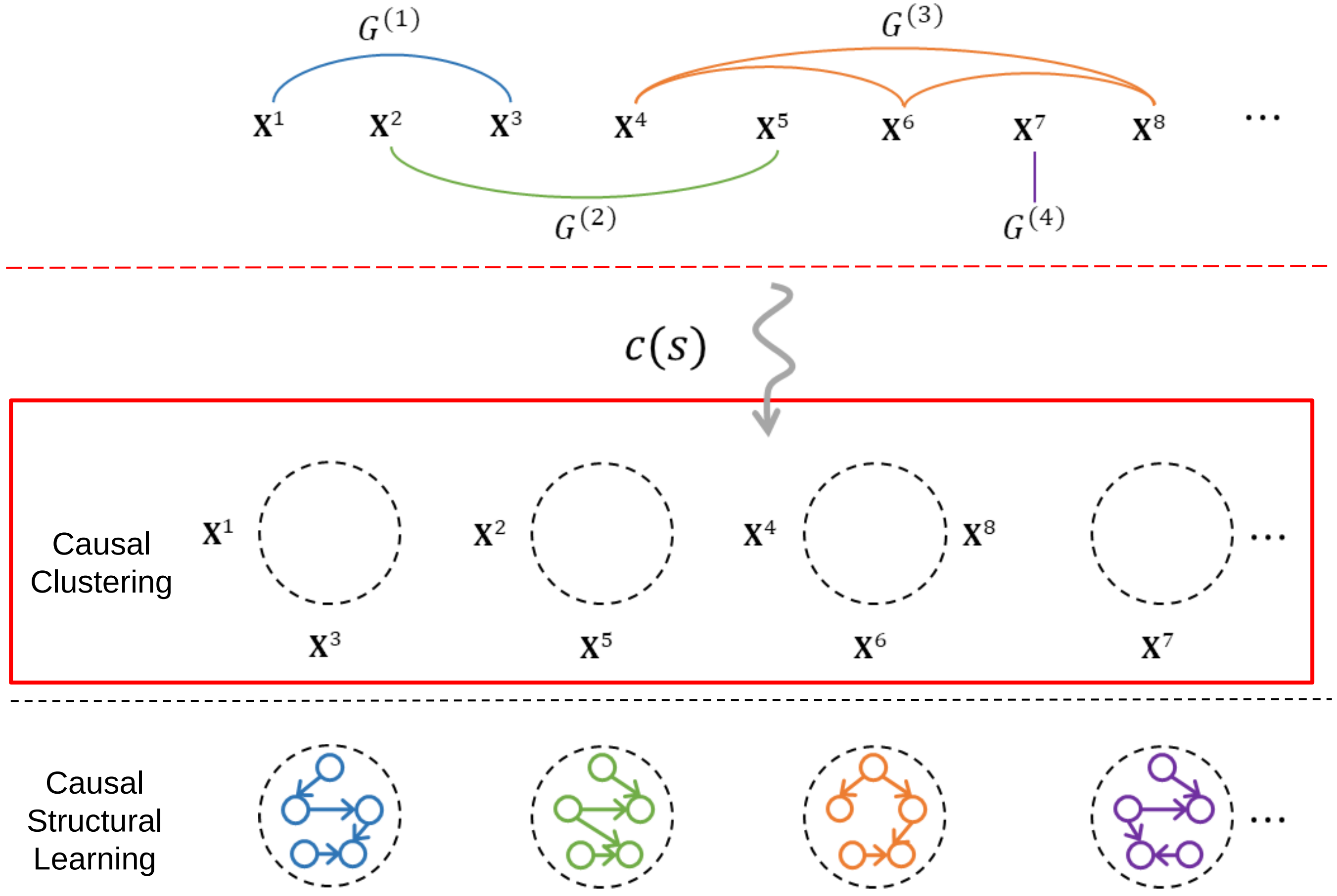
Recovering causal structures

Chen, W., Wu, Y., **Cai, R.**, et al. *CCSL: A Causal Structure Learning Method from Multiple Unknown Environments*. arXiv, 2022.

CCSL: causal clustering

☑ Challenge 1: how to group the subjects that are implied the same causal structure?

☑ Causal CRP: Chinese Restaurant Process



$$P\left(c_s = k | c_{1:(s-1)}, \alpha\right) \propto \begin{cases} P(c_s = k | \mathbf{X}^s) & k \leq K, \\ \alpha & k = K+1. \end{cases}$$

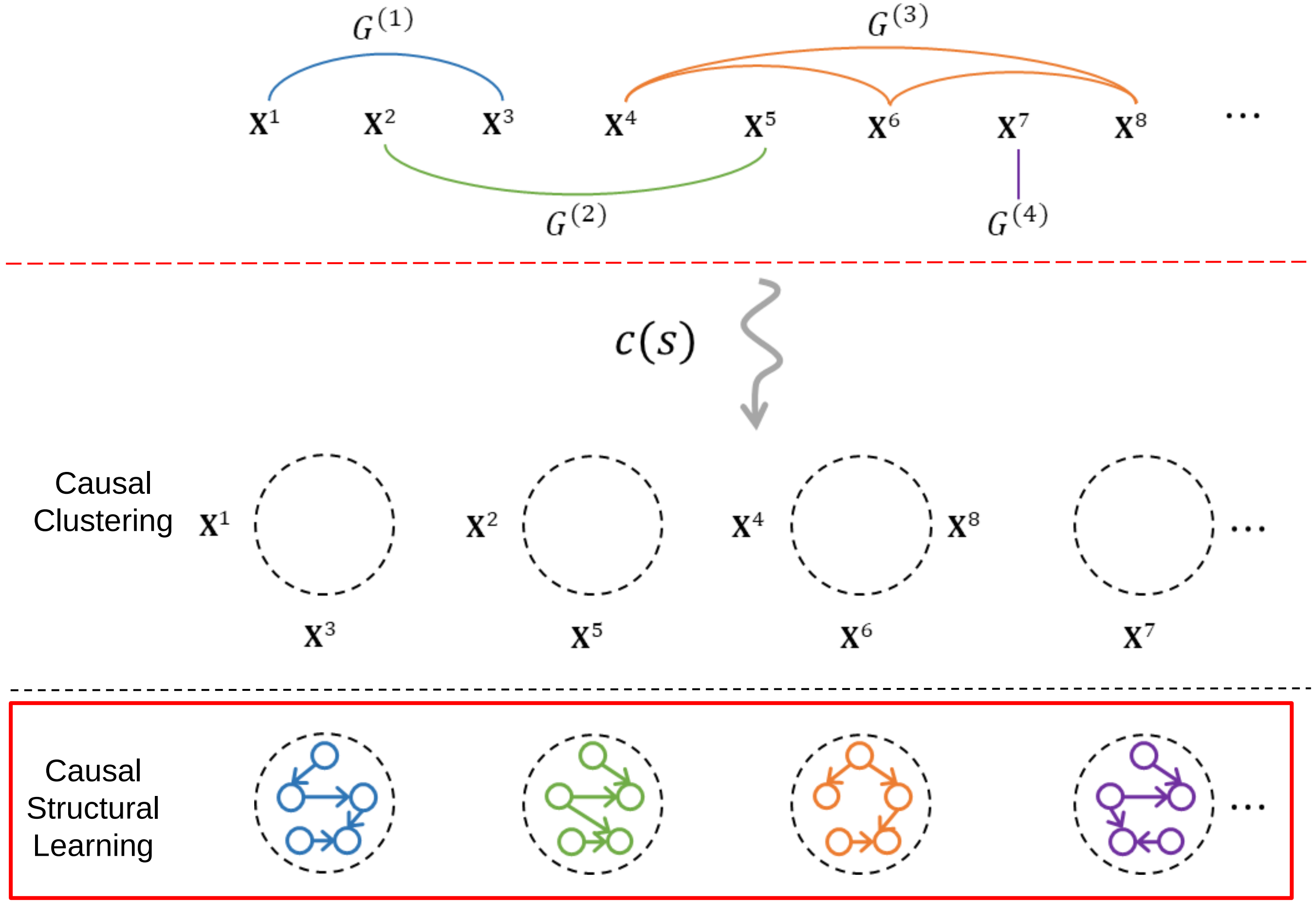
Chen, W., Wu, Y., **Cai, R.**, et al. *CCSL: A Causal Structure Learning Method from Multiple Unknown Environments*. arXiv, 2022.

CCSL: causal structural learning

☑ Challenge 2: how to recover the causal structure?

☑ Causal Structure Model for Each Cluster:

$$\mathbf{X}^{(k)}(t) = B^{(k)}X^{(k)}(t) + \sum_{p=1}^{p_l} A_p^{(k)}X^{(k)}(l-p) + E^{(k)}(t)$$

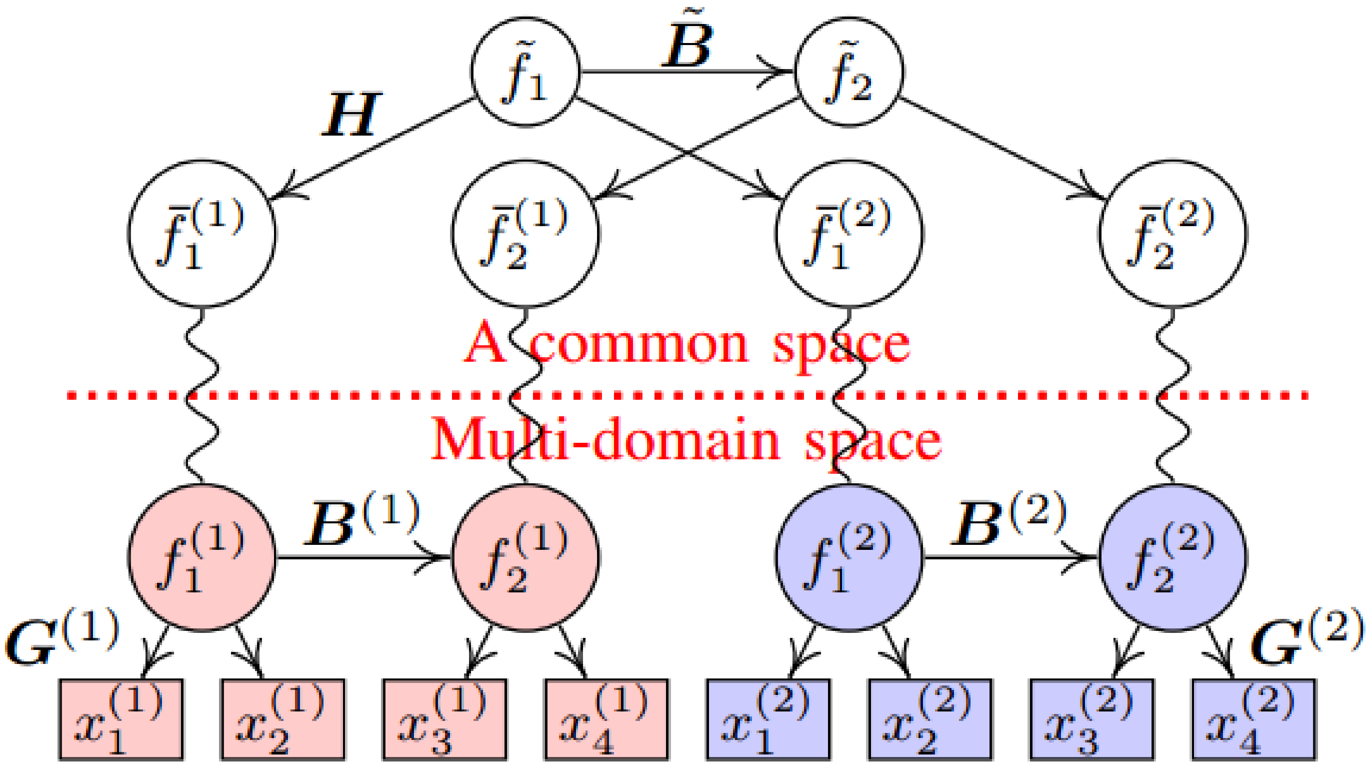


$$P(\mathbf{X}^s | c_s = k)$$

Chen, W., Wu, Y., **Cai, R.**, et al. *CCSL: A Causal Structure Learning Method from Multiple Unknown Environments*. arXiv, 2022.

MD-LiNA：Multi-Domain causal discovery

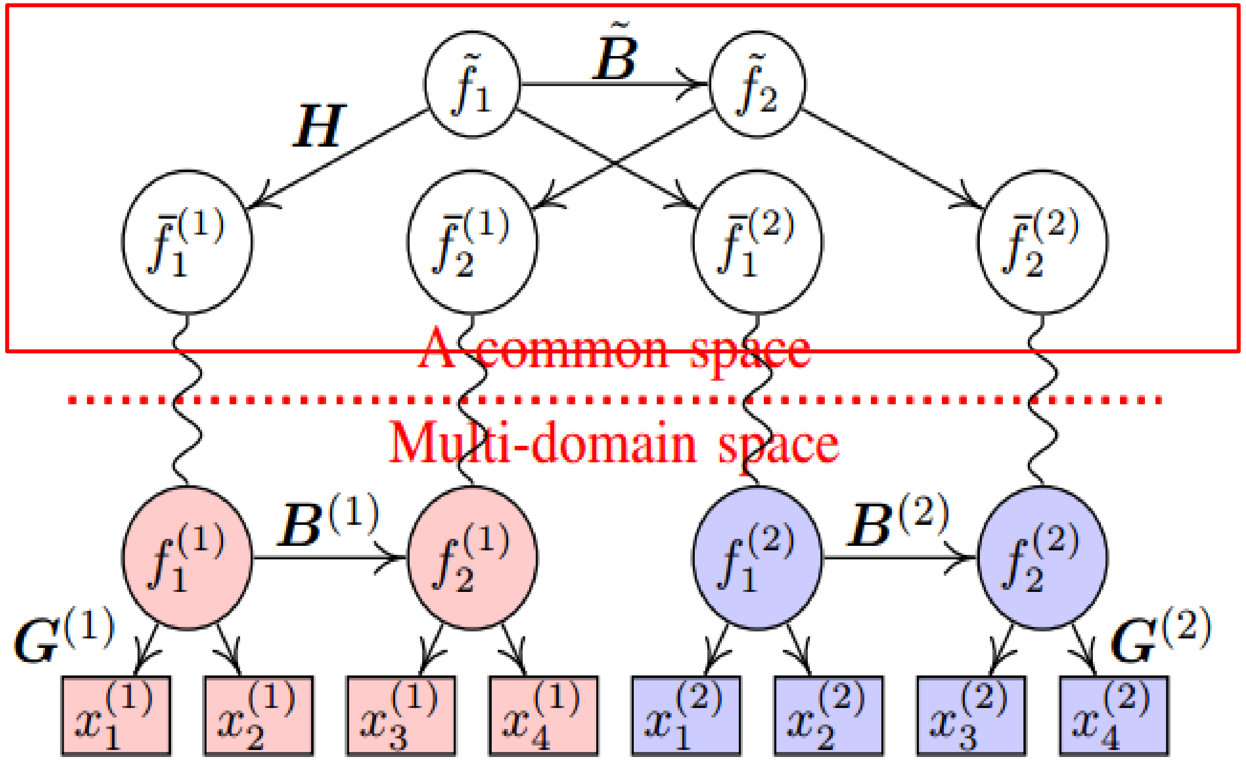
☑ Common Space V.S. Multi-domain space



Zeng, Y., Shimizu, S., **Cai, R.,** Xie, F., Yamamoto, M., & Hao, Z. (2020). *Causal discovery with multi-domain LiNGAM for latent factors*. IJCAI 2021.

MD-LiNA: Formalization

☑ The formalization of our model is shown in these three equations. To integrate the multi-domain data, we use a simple coding representation method.

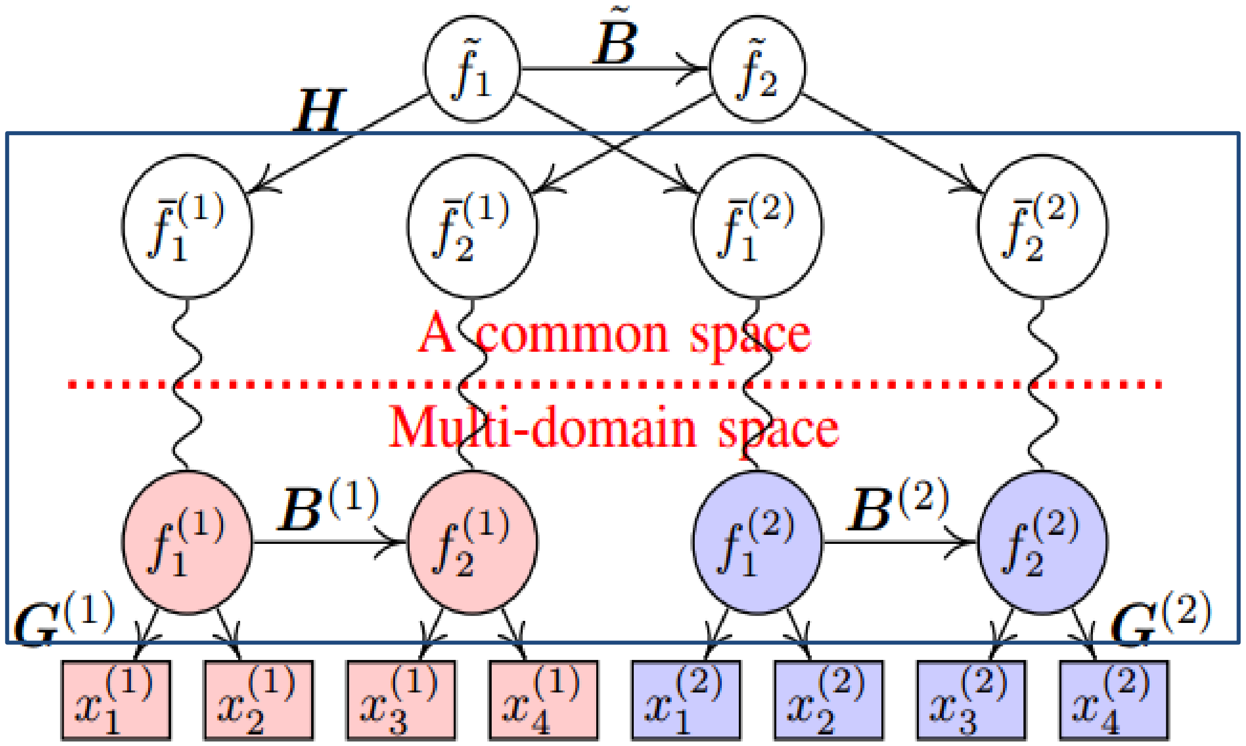


⇨ $\boldsymbol{f}^{(m)} = \boldsymbol{B}^{(m)} \boldsymbol{f}^{(m)} + \boldsymbol{\varepsilon}^{(m)}$

Zeng, Y., Shimizu, S., **Cai, R.,** Xie, F., Yamamoto, M., & Hao, Z. (2020). *Causal discovery with multi-domain LiNGAM for latent factors*. IJCAI 2021.

MD-LiNA：Formalization

☑ The formalization of our model is shown in these three equations. To integrate the multi-domain data, we use a simple coding representation method.



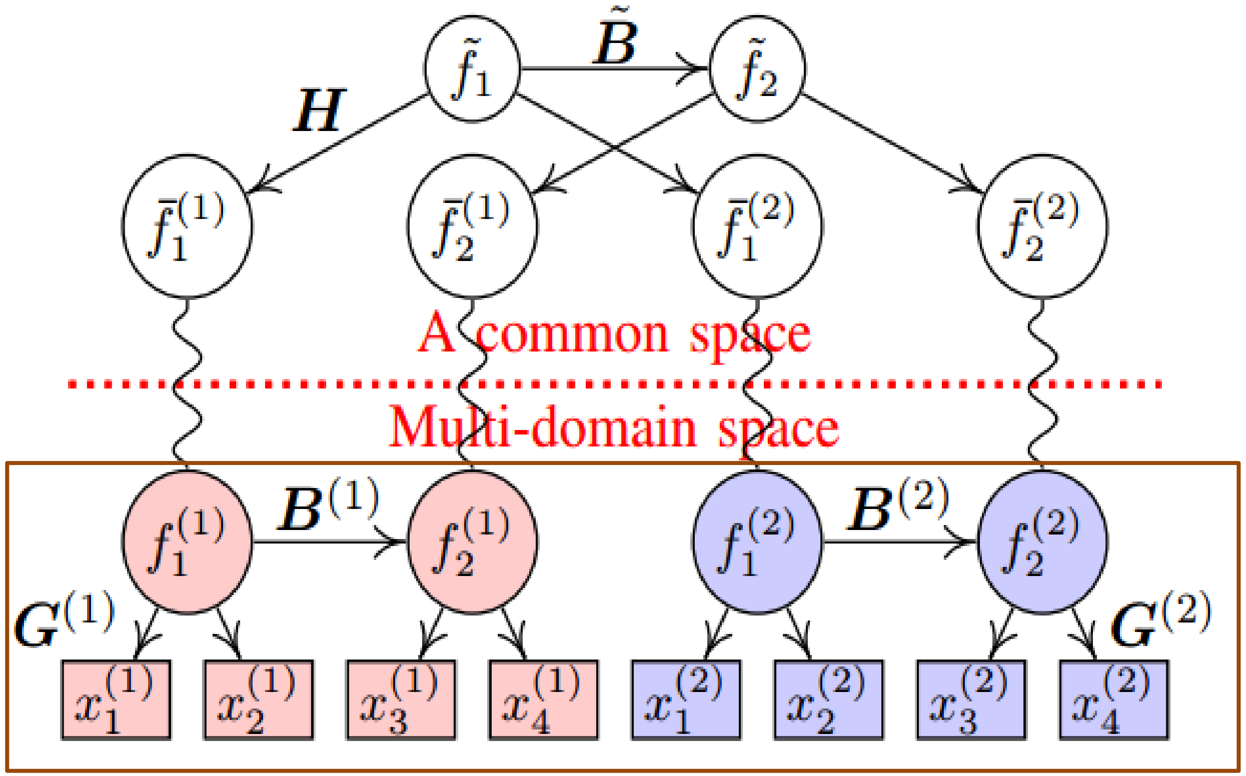
$$\boldsymbol{f}^{(m)} = \boldsymbol{B}^{(m)} \boldsymbol{f}^{(m)} + \boldsymbol{\varepsilon}^{(m)}$$

$$\Rightarrow \quad \bar{\boldsymbol{f}} = \boldsymbol{H} \tilde{\boldsymbol{f}}$$

Zeng, Y., Shimizu, S., **Cai, R.,** Xie, F., Yamamoto, M., & Hao, Z. (2020). *Causal discovery with multi-domain LiNGAM for latent factors*. IJCAI 2021.

MD-LiNA： Formalization

☑ The formalization of our model is shown in these three equations. To integrate the multi-domain data, we use a simple coding representation method.



$$\boldsymbol{f}^{(m)} = \boldsymbol{B}^{(m)} \boldsymbol{f}^{(m)} + \boldsymbol{\varepsilon}^{(m)}$$

$$\bar{\boldsymbol{f}} = \boldsymbol{H} \tilde{\boldsymbol{f}}$$

$$\Rightarrow \quad \boldsymbol{x}^{(m)} = \boldsymbol{G}^{(m)} \boldsymbol{f}^{(m)} + \boldsymbol{e}^{(m)}$$

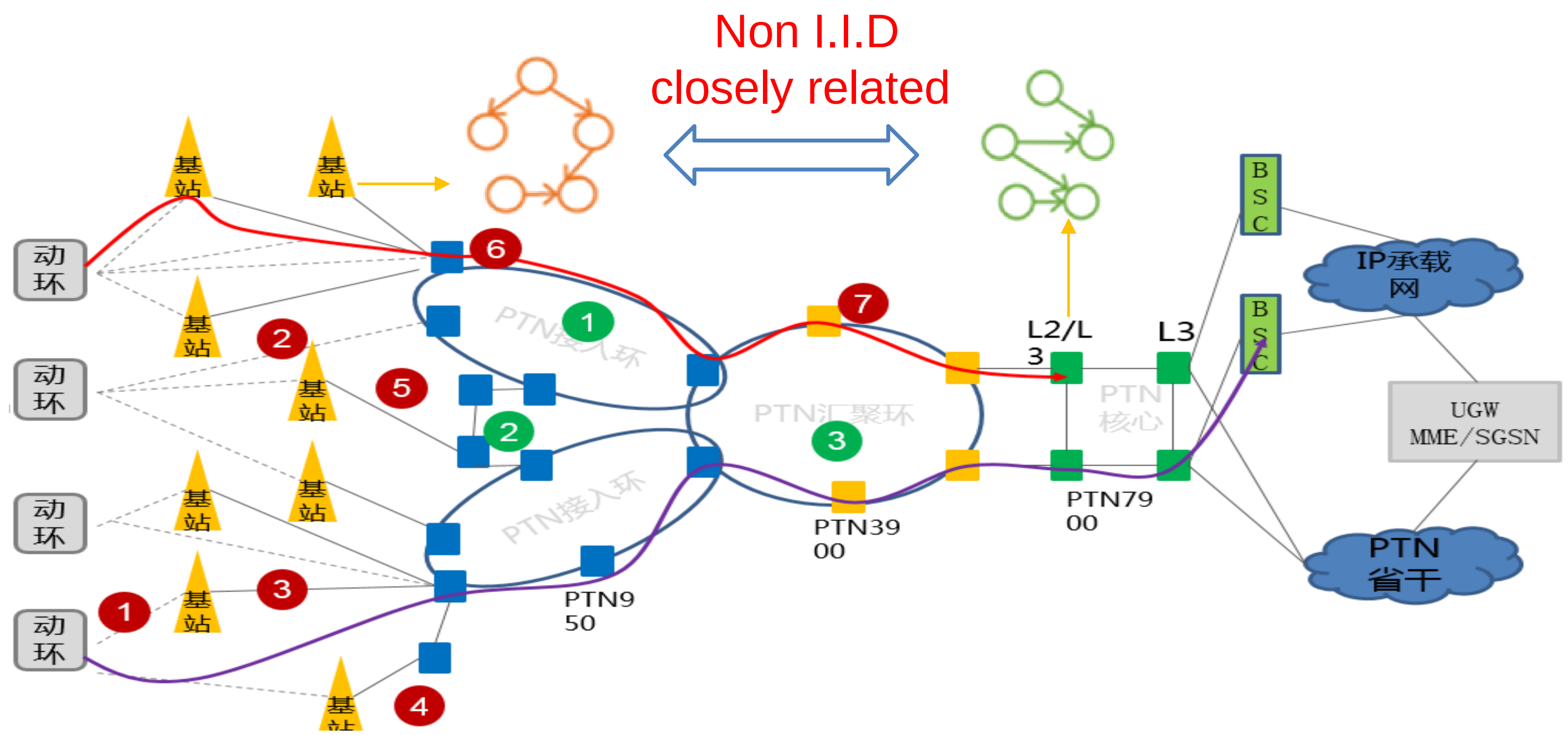
Zeng, Y., Shimizu, S., **Cai, R.,** Xie, F., Yamamoto, M., & Hao, Z. (2020). *Causal discovery with multi-domain LiNGAM for latent factors*. IJCAI 2021.

应用探索

☑ 根因故障定位

通信网络故障根因故障定位

☑ The data are generated from multiple topologically related environments, how to discover the causal structures from data?



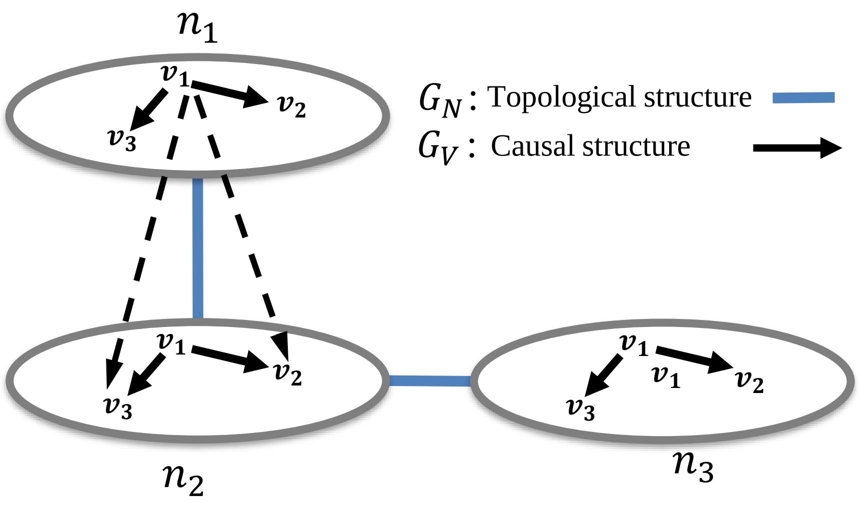
☑ It is crucial to consider the topological structure behind the data for learning Granger causality.

Cai R, Wu S et al. *THP: Topological Hawkes Processes for Learning Granger Causality on Event Sequences*. Submitted to TNNLS, arXiv:2105.10884, 2021.

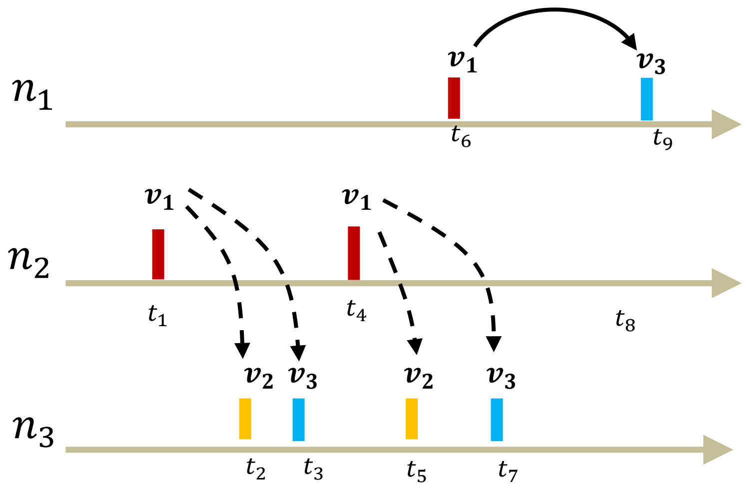
THP：The topological-temporal Hawkes model

☑ How to learn **the Granger causality** among **event types** using the **event sequences** that generated by nodes in a **topological network**?

topological-temporal Hawkes model



topological-causal view



temporal-causal view (hawkes process)

Cai R, Wu S et al. *THP: Topological Hawkes Processes for Learning Granger Causality on Event Sequences*. Submitted to TNNLS, arXiv:2105.10884, 2021.

A unified view from convolution

☑ Look at the Hawkes process from a temporal convolution perspective:

$$\lambda_v(t) = \mu_v + \sum_{v' \in \mathbf{PA}_v} \int_{t' \in \mathbf{T}_{t-}} \phi_{v',v}(t - t') d\mathrm{C}_{v'}(t') \quad \Longleftrightarrow \quad \lambda_v(t) = \mu_v + \sum_{v' \in \mathbf{PA}_v} \left(\phi_{v',v} * d\mathrm{C}_{v'}\right)_{\mathbf{T}}(t)$$

☑ Look at the topological structure from a graph convolution perspective :

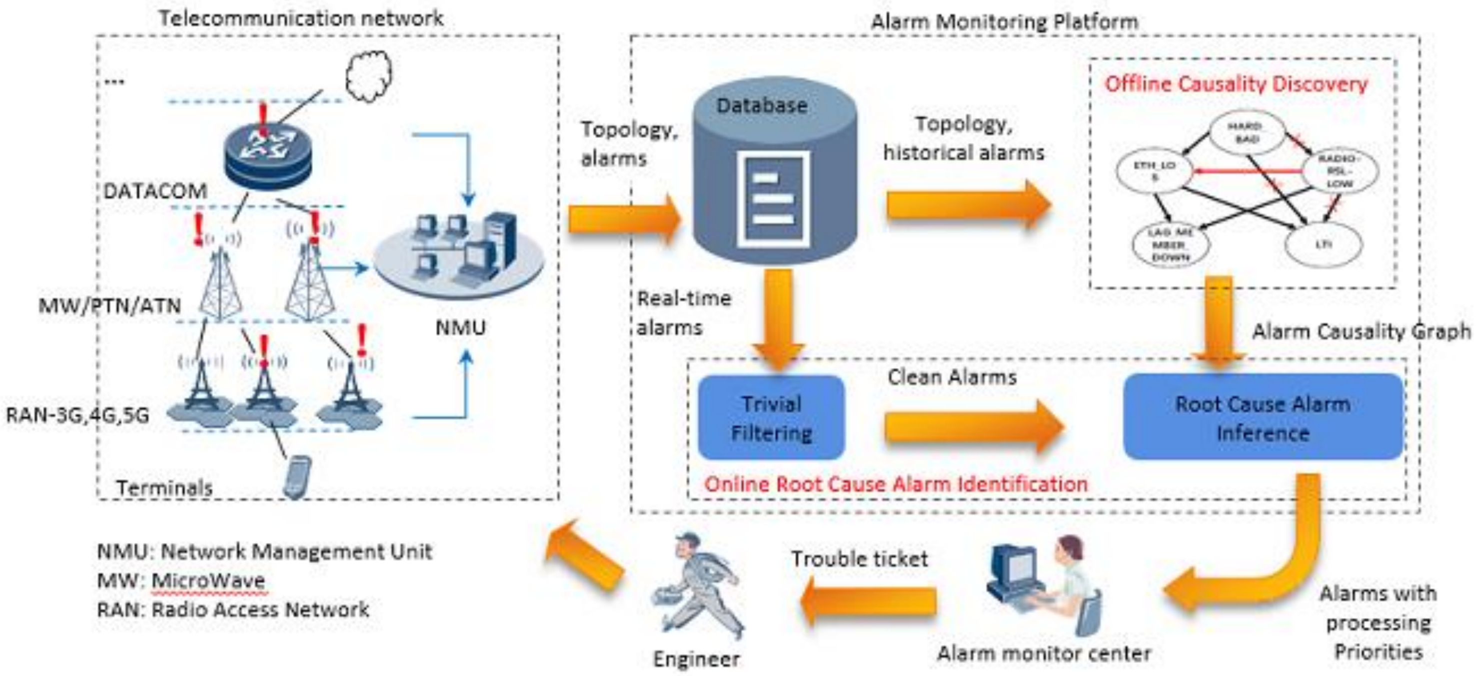
$$y = g_\theta(L)s = g_\theta(U\Gamma U^T)s = Ug_\theta(\Gamma)U^T s$$

☑ Extend the causal intensity function to the topological-temporal domain $\mathcal{G}_N \times \mathrm{T}$.

$$\lambda_v(n,t) = \mu_v + \sum_{v' \in \mathbf{PA}} (\psi_{v',v} * dC_{v'})_{\mathcal{G}_N \times \mathbf{T}}(n,t),$$

Cai R, Wu S et al. *THP: Topological Hawkes Processes for Learning Granger Causality on Event Sequences*. Submitted to TNNLS, arXiv:2105.10884, 2021.

The 1St in PCIC 2021 Causal Discovery challenge

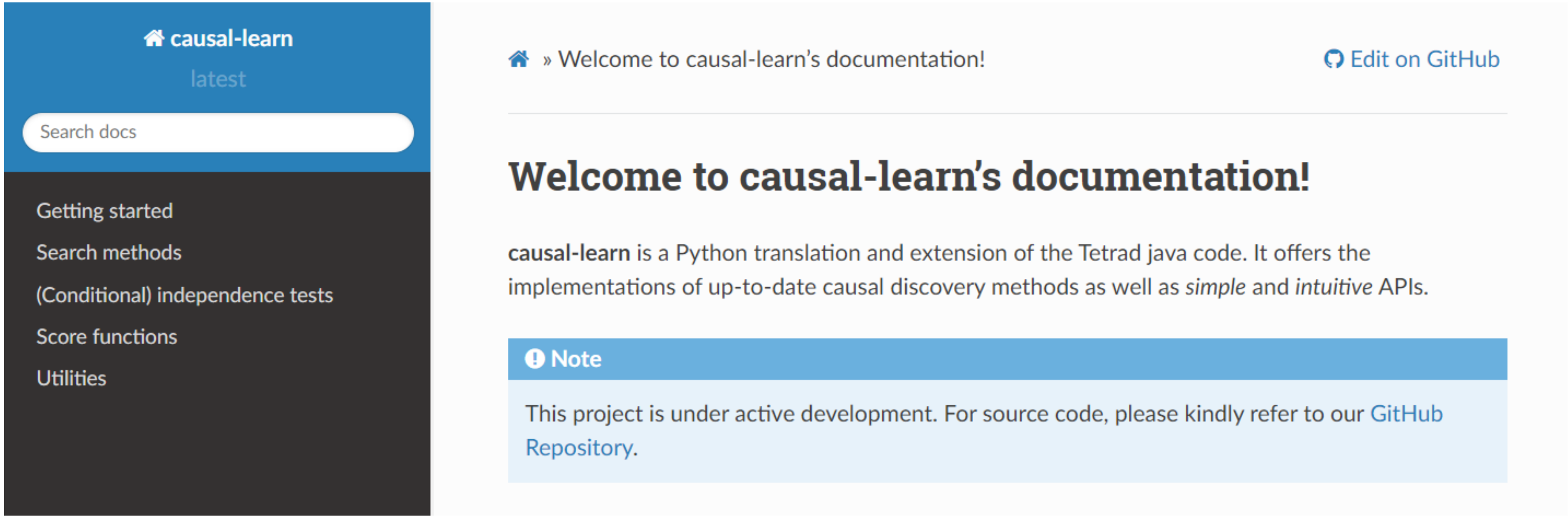


Datasets: https://competition.huaweicloud.com/information/1000041487/introduction
Codes: https://github.com/DMIRLAB-Group/CDMIR

CAUSAL-LEARN：因果学习开源算法平台

causal-learn: 因果学习算法平台

- ☑ 基于Python实现了经典和部分最新的因果学习算法。其中包含了因果发现的经典算法与API，并且提供了模块化的代码



- ☑ GitHub: https://github.com/cmu-phil/causal-learn
- ☑ 文档: https://causal-learn.readthedocs.io/en/latest/
- ☑ 简单使用案例：https://github.com/cmu-phil/causal-learn/tree/main/tests

causal-learn: 因果学习算法平台

☑ causal-learn支持:

- **基于约束的因果发现方法**（Constrained-based causal discovery methods）: PC、FCI、CD-NOD算法等;
- **基于评分的因果发现方法**（Score-based causal discovery methods）： 包含BIC、BDeu、generalized score等评分的GES算法;
- **基于函数因果模型的因果发现方法**（Functional causal models-based causal discovery methods）：LiNGAM及其拓展方法、ANM、PNL等;
- **隐因果表征学习方法**（Hidden causal representation learning）：GIN方法;
- **格兰杰因果分析**（Granger causal analysis）；
- **多个独立的基础模块**，比如独立性测试，评分函数，图操作，评测指标;
- 更多最新的因果发现算法，如gradient-based methods等。

causal-learn: 使用方法

☑ **安装**：可以通过pip来实现

```
pip install causal-learn
```

```
1  ! pip install causal-learn
```

```
Collecting causal-learn
  Downloading causal_learn-0.1.2.3-py3-none-any.whl (163 kB)
     |████████████████████████████████| 163 kB 26.6 MB/s
Requirement already satisfied: statsmodels in /usr/local/lib/python3.7/dist-packages (from causal-learn) (0.10.2)
Requirement already satisfied: graphviz in /usr/local/lib/python3.7/dist-packages (from causal-learn) (0.10.1)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.7/dist-packages (from causal-learn) (1.0.2)
Requirement already satisfied: tqdm in /usr/local/lib/python3.7/dist-packages (from causal-learn) (4.64.0)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (from causal-learn) (1.21.6)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.7/dist-packages (from causal-learn) (3.2.2)
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (from causal-learn) (1.3.5)
Requirement already satisfied: pydot in /usr/local/lib/python3.7/dist-packages (from causal-learn) (1.3.0)
Requirement already satisfied: networkx in /usr/local/lib/python3.7/dist-packages (from causal-learn) (2.6.3)
Requirement already satisfied: scipy in /usr/local/lib/python3.7/dist-packages (from causal-learn) (1.4.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-packages (from matplotlib->causal-learn) (0.11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib->causal-learn) (1.4.2)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib->causal-learn) (3.0.8)
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib->causal-learn) (2.8.2)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.7/dist-packages (from kiwisolver>=1.0.1->matplotlib->causal-learn) (4.1.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.1->matplotlib->causal-learn) (1.15.0)
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pandas->causal-learn) (2022.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.7/dist-packages (from scikit-learn->causal-learn) (3.1.0)
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/dist-packages (from scikit-learn->causal-learn) (1.1.0)
Requirement already satisfied: patsy>=0.4.0 in /usr/local/lib/python3.7/dist-packages (from statsmodels->causal-learn) (0.5.2)
Installing collected packages: causal-learn
Successfully installed causal-learn-0.1.2.3
```

causal-learn: PC算法使用方法

☑ **使用**：causal-learn为所有模型都提供了简单易用的接口，用户可以通过一行代码在自己的数据上进行因果发现

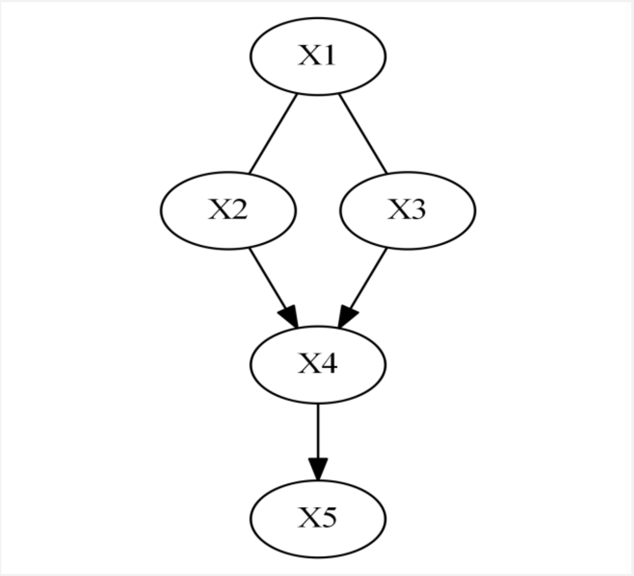
```python
from causallearn.search.ConstraintBased.PC import pc
cg = pc(data, alpha, indep_test, stable, uc_rule, uc_priority, mvpc,
correction_name, background_knowledge, verbose, show_progress)

# visualization using pydot
cg.draw_pydot_graph()
```

```
3 dep 4 | (0, 2) with p-value 0.000000
3 dep 4 | (1, 2) with p-value 0.000000
Graph Nodes:
X1;X2;X3;X4;X5

Graph Edges:
1. X1 --- X2
2. X1 --- X3
3. X2 --> X4
4. X3 --> X4
5. X4 --> X5

Depth=2, working on node 4: 100%|██████████| 5/5 [00:00<00:00, 418.08it/s]
```



causal-learn: GIN算法使用方法

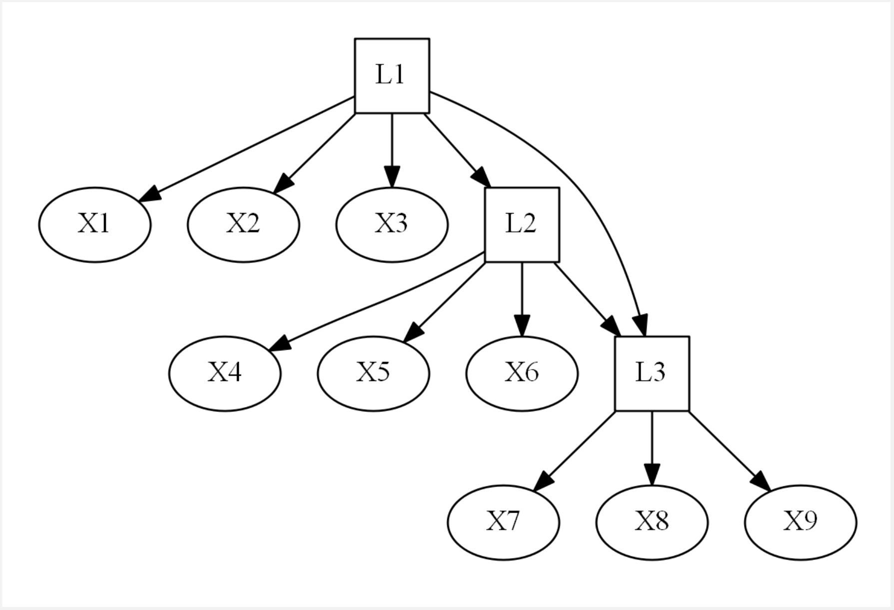
☑ **使用**：causal-learn为所有模型都提供了简单易用的接口，用户可以通过一行代码在自己的数据上进行因果发现

```
from causallearn.search.FCMBased.GIN.GIN import GIN
G, K = GIN(data, indep_test, alpha)

# visualization using pydot
pyd = GraphUtils.to_pydot(G)
pyd.write_png('test.png')
```

```
Runing the Step 1: Finding the Causal Clusters
Runing the Step 2: Learning the Causal Order of Latent Variables
Graph Nodes:
L1;X2;X3;X1;L2;X9;X7;X8;L3;X6;X4;X5

Graph Edges:
1. L1 ⟶ X2
2. L1 ⟶ X3
3. L1 ⟶ X1
4. L1 ⟶ L2
5. L1 ⟶ L3
6. L2 ⟶ X9
7. L2 ⟶ X7
8. L2 ⟶ X8
9. L2 ⟶ L3
10. L3 ⟶ X6
11. L3 ⟶ X4
12. L3 ⟶ X5
 [[1, 2, 0], [8, 6, 7], [5, 3, 4]]
```

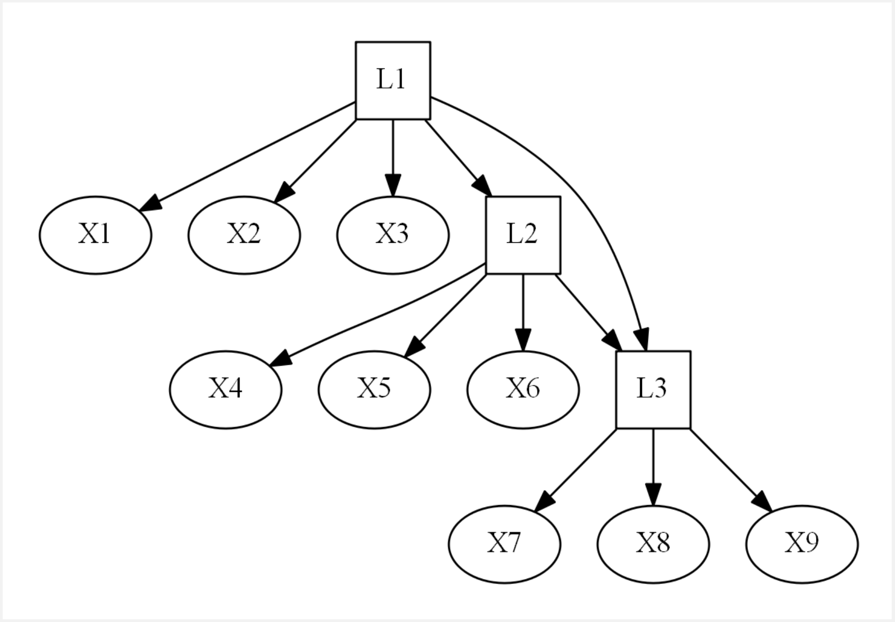


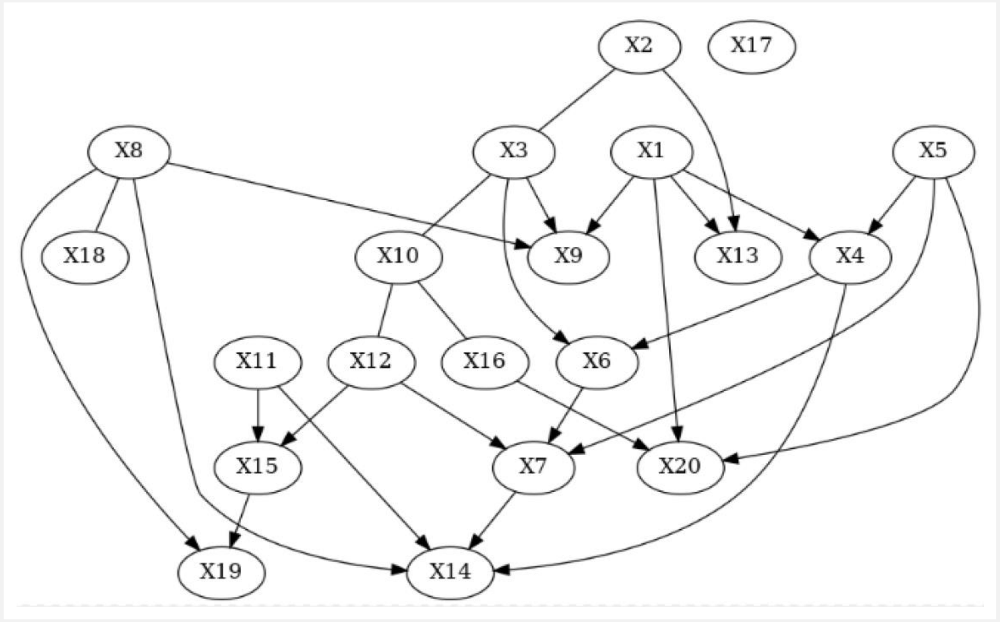
causal-learn: 使用方法

☑ **可视化结果与评测:** 在算法运行结束后，用户可以查看生成的因果图，并通过多种评测指标来与基准图进行对比

```python
from causallearn.utils.GraphUtils import GraphUtils

# visualization using pydot
pyd = GraphUtils.to_pydot(G)
pyd.write_png('test.png')
```





Conclusion

Causal discovery = Causal thinking/assumptions + ML/Statistical tools