Advanced techniques in applied economics

Lecture 4: Causal discovery and non-Gaussian identification

Piotr Zwiernik

upf. Universitat Pompeu Fabra Barcelona

Statistics, Probability and Machine Learning Research Group

BSE Barcelona School of Economics

Spring 2026

What can we learn from observational data alone?

What changes today?

In Lecture 3 we gave causal meaning to DAGs using factorization, d-separation, and interventions. Today we ask a harder question:

How much of the causal direction can be recovered without intervening?

Constraint-based discovery
Use conditional independence tests to recover the skeleton and the v-structures.

Extra identifying structure
Use assumptions such as non-Gaussianity or nonlinear asymmetry to go beyond CPDAGs.

Main question: Which arrow directions are visible from conditional independence alone, and which require extra asymmetry assumptions?

Roadmap for today

1. constraint-based discovery and the PC algorithm
2. why Markov equivalence is the main obstacle
3. linear SEMs and why Gaussian models do not identify direction
4. LiNGAM and the role of non-Gaussianity
5. more realistic shocks: common volatility and mean independence
6. a brief link to additive noise models

Conditional independence narrows down the graph; asymmetry assumptions identify direction.

Part 1: Constraint-based discovery

Why conditional independence alone stops at a CPDAG

Constraint-based discovery uses the logic of Lecture 3:

- d-separation implies conditional independence,
- missing edges correspond to some separating set,
- colliders can be oriented from separating sets.

From observational data we may learn

- the **skeleton**,
- the **v-structures**,
- and therefore a **CPDAG**.

What this does not buy us

Without extra assumptions, we usually cannot orient every edge.

High-level idea of the PC algorithm

The PC algorithm is the canonical constraint-based discovery method.

Skeleton phase

1. Start from the complete undirected graph.
2. Test marginal and conditional independences.
3. Remove an edge whenever some separating set is found.

Orientation phase

1. Use separating sets to orient v-structures.
2. Apply graphical rules to orient additional edges.

Output

A **CPDAG**, not necessarily a unique DAG.

PC is a discovery procedure based on conditional independence, not on interventions.

For the general logic of constraint-based discovery, see Spirtes, Glymour, and Scheines, *Causation, Prediction, and Search*, Chapters 4–5.

The PC algorithm: skeleton phase

Input

Data on $X_1, \ldots, X_m$, a conditional independence test, and a significance level α.

Skeleton phase

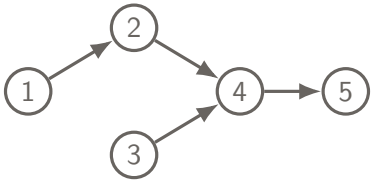
1. Initialize with the complete undirected graph on $\{1, \ldots, m\}$.
2. For conditioning sets of size $0, 1, 2, \ldots$:
   - for each adjacent pair (i, j),
   - test $X_i \perp\!\!\!\perp X_j \mid X_S$ for candidate sets S,
   - if independence is not rejected, remove edge $i - j$ and store S.

Interpretation

Search for a small set of variables that explains away the apparent association between X_i and X_j.

A toy discovery example

Suppose the true DAG is



Skeleton phase intuition

- marginal tests may already remove $1 - 3$ and $2 - 3$,
- then conditioning on 2 may remove $1 - 4$ and $1 - 5$,
- conditioning on 4 may remove $2 - 5$ and $3 - 5$.

The skeleton is found by asking which pairs can be separated by some conditioning set.

Orientation phase: where colliders come from

Suppose the skeleton contains

with no edge between i and k. If the separating set for (i, k) *does not* contain node j, then PC orients

$$i \to j \leftarrow k.$$

Why?

If j were a non-collider, then j would have to belong to every **minimal** separator between i and k.

After orienting all such colliders, PC applies **Meek's rules** to propagate additional arrow directions while avoiding new v-structures and directed cycles.

For the orientation rules, see Meek (1995), *UAI*.

A first PC example in R

```
library(pcalg)
library(MASS)
set.seed(1)

Sigma <- matrix(c(1,.7,0,.49,0,
                  .7,1,0,.7,0,
                  0,0,1,.6,0,
                  .49,.7,.6,1,.7,
                  0,0,0,.7,1), 5, 5)
X <- mvrnorm(500, mu = rep(0, 5),
             Sigma = Sigma)
```

```
suffStat <- list(C = cor(X), n = nrow(X))
pc.fit <- pc(suffStat,
             indepTest = gaussCItest,
             alpha = 0.01,
             labels = paste0("X", 1:5),
             method = "stable")

pc.fit
```

Interpretation

For Gaussian data, the PC algorithm uses partial-correlation tests to recover a CPDAG.

For the pcalg implementation, see Kalisch et al. (2012), *Journal of Statistical Software*.

Why we should be cautious with PC

Constraint-based discovery is elegant, but it rests on demanding assumptions.

Statistical issues

- many CI tests,
- finite-sample error propagation,
- high-dimensional conditioning sets,
- sensitivity to near-unfaithfulness.

Economic issues

- latent confounding is common,
- Gaussianity may be implausible,
- observational data may identify only part of the structure.

PC is best viewed as a disciplined way to learn a CPDAG, not as a button that magically discovers the true causal graph.

Whenever possible, incorporate expert knowledge into the learning algorithm.

Part 2: Linear SEMs and Gaussian symmetry

Linear structural equation models

A linear SEM associated with a DAG has the form

$$X_i = \sum_{j \in \mathrm{pa}(i)} \beta_{ij} X_j + \varepsilon_i, \qquad i = 1, \ldots, m,$$

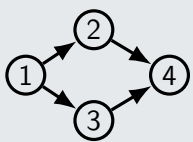
where $\varepsilon_1, \ldots, \varepsilon_m$ are mutually independent (or at least satisfy $\mathbb{E}(\varepsilon_i \mid X_{\mathrm{pa}(i)}) = 0$).

Matrix form: $X = BX + \varepsilon$,
where $B_{ij} = 0$ when $j \notin \mathrm{pa}(i)$.

Note: The zero pattern of B is exactly the DAG.

In the example, X_1 is a primitive shock, X_2 and X_3 are intermediate channels, and X_4 is the outcome of interest.

$$X_1 = \varepsilon_1,$$
$$X_2 = \beta_{21} X_1 + \varepsilon_2,$$
$$X_3 = \beta_{31} X_1 + \varepsilon_3,$$
$$X_4 = \beta_{42} X_2 + \beta_{43} X_3 + \varepsilon_4.$$



For linear SEMs, see Bollen, *Structural Equations with Latent Variables*, Chapter 3, or Drton, Eichler, and Richardson in the *Handbook of Graphical Models*.

Gaussian LSEMs are exactly Gaussian DAG models

If X is jointly Gaussian, then each conditional distribution $X_i \mid X_{\mathrm{pa}(i)}$ is Gaussian. So a Gaussian DAG model is specified by choosing these Gaussian conditionals.

Gaussian LSEM over a DAG writes

$$X_i = \sum_{j \in \mathrm{pa}(i)} \beta_{ij} X_j + \varepsilon_i, \qquad \varepsilon_i \sim N(0, \omega_i),$$

with independent Gaussian errors.

Conditional law:

$$X_i \mid X_{\mathrm{pa}(i)} \sim N\Big(\sum_{j \in \mathrm{pa}(i)} \beta_{ij} X_j, \omega_i\Big).$$

Gaussian LSEMs are exactly Gaussian DAG models

Comparing the two, we conclude that linear Gaussian SEM specifies exactly the Gaussian conditionals appearing in the DAG factorization.

Structural coefficients and total effects

Structural coefficients
In the linear SEM $X = BX + \varepsilon$, B records the **direct** linear effects encoded by the DAG.

Reduced-form map Solving the system gives

$$X = (I - B)^{-1}\varepsilon.$$

So the matrix $(I - B)^{-1}$ combines **direct and indirect** effects.

Structural coefficients describe local mechanisms.
The reduced-form map describes the total propagation of shocks.

Why linear Gaussian SEMs do not identify direction

Take two centered Gaussian variables (X, Y).

Forward representation

$$Y = aX + \varepsilon_Y, \qquad \varepsilon_Y \perp\!\!\!\perp X.$$

Backward representation

$$X = bY + \varepsilon_X, \qquad \varepsilon_X \perp\!\!\!\perp Y.$$

Key point

The same bivariate Gaussian distribution admits both directions as valid linear SEMs with independent noise.

So linearity alone is not enough. Within Gaussian LSEMs, causal direction is hidden by distributional symmetry.

Part 3: LiNGAM and non-Gaussianity

What kind of extra assumptions can break the symmetry?

To identify direction from observational data, we need an asymmetry that is invisible in the Gaussian model.

Four common strategies

- **Constraint-based methods:** rely on conditional independence and stop at a CPDAG.
- **Gaussian linear models with symmetries:** e.g. $\sigma_i^2 = \sigma^2$ for all i.
- **Nonlinear Gaussian noise models:** additive noise models use functional asymmetry.
- **Non-Gaussian linear models:** LiNGAM uses non-Gaussianity of the shocks.

Today's focus

We stay in the linear SEM world and ask what non-Gaussian shocks can reveal.

LiNGAM assumptions

LiNGAM stands for **Linear Non-Gaussian Acyclic Model**.

Model

$$X = BX + \varepsilon,$$

where

- B is compatible with a DAG,
- the errors $\varepsilon_1, \ldots, \varepsilon_m$ are mutually independent,
- the errors are non-Gaussian (except possibly one),

LiNGAM keeps linearity and acyclicity, but replaces Gaussian noise by non-Gaussian noise.

For the original LiNGAM result, see Shimizu et al. (2006), *Journal of Machine Learning Research*.

Why non-Gaussianity helps: $X \to Y$

In the Gaussian case, both directions may admit a linear model with independent noise.

With non-Gaussian noise, this symmetry breaks.

Bivariate intuition

Suppose

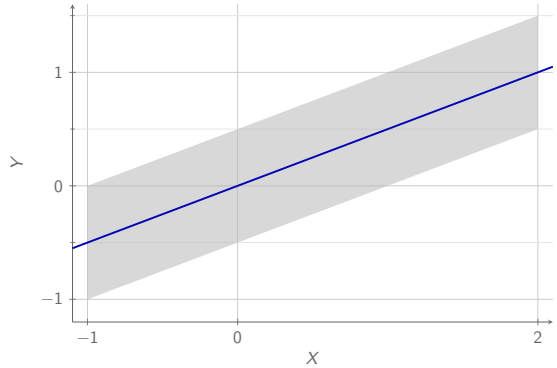
$$Y = aX + \varepsilon, \qquad \varepsilon \perp\!\!\!\perp X,$$

with non-Gaussian X and ε. Then, in general, there is **no** backward linear model

$$X = bY + \widetilde{\varepsilon}, \qquad \widetilde{\varepsilon} \perp\!\!\!\perp Y.$$

For LiNGAM identifiability, see Shimizu et al. (2006), *A Linear Non-Gaussian Acyclic Model for Causal Discovery*.

Why non-Gaussianity helps: forward model

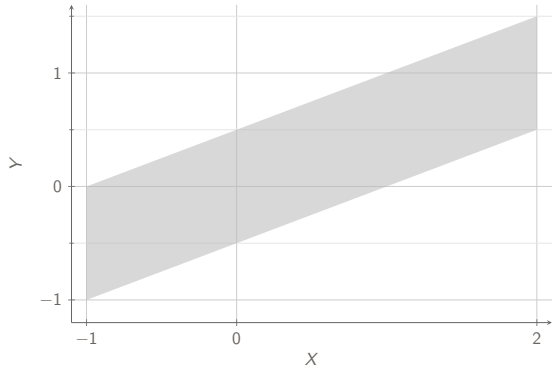


Consider the forward model

$$Y = 0.5\,X + \varepsilon_Y, \qquad \varepsilon_Y \perp\!\!\!\perp X.$$

The gray strip is exactly the support implied by $\varepsilon_Y \sim U[-\frac{1}{2}, \frac{1}{2}]$ and $X \sim U[-1, 2]$.

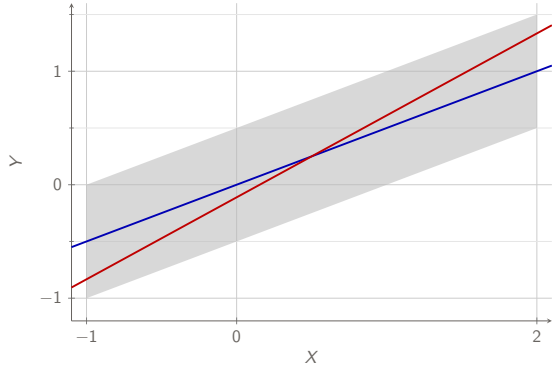
Why non-Gaussianity helps: reverse question



Question: can the same joint distribution arise from a backward linear model

$$X = bY + c + \varepsilon_X, \qquad \varepsilon_X \perp\!\!\!\perp Y \ ?$$

Why non-Gaussianity helps: failed reverse fit



Let the red line be the least-squares fit for predicting X from Y:

$$(b, c) = \arg\min_{b,c} \mathbb{E}(X - bY - c)^2.$$

Even with the best linear backward fit, the residuals cannot be independent of Y.

DirectLiNGAM: the main idea

DirectLiNGAM looks for an exogenous variable first.

Key insight. If X_j has no parents, then $X_j = \varepsilon_j$. Now take another variable X_i. In a linear SEM it can be decomposed as

$$X_i = \beta_{ij} X_j + r_i^{(j)},$$

where the remainder $r_i^{(j)}$ depends only on the other structural shocks.
Thus, $X_j = \varepsilon_j$ is independent of $r_k^{(j)}$.

1. Find a variable whose regression residuals are as independent from it as possible.
2. Treat it as exogenous and remove its effect from the remaining variables.
3. Repeat on the residual system.

For DirectLiNGAM, see Shimizu et al. (2011), *Journal of Machine Learning Research*.

A simple LiNGAM simulation in R

```r
set.seed(1)
n <- 2000

x1 <- rt(n, df = 5)
x2 <- 0.8 * x1 + rt(n, df = 5)
x3 <- -0.6 * x1 + 0.7 * x2 +
      rt(n, df = 5)

X <- cbind(x1, x2, x3)
colnames(X) <- c("x1", "x2", "x3")
```

```r
library(pcalg)
fit <- lingam(X)
fit$Bpruned
```

Interpretation

The estimated matrix `Bpruned` gives the directed linear effects. Because the shocks are non-Gaussian, the causal ordering can become identifiable even though the model is linear.

Examples

- asset-return shocks are heavy-tailed,
- firm-level outcomes are skewed,
- macro shocks often have outliers and asymmetries.

Non-Gaussianity is not only a robustness issue; it can also be an **identification resource**.

Part 4: Beyond strict independence of shocks

Why independent shocks may be too strong in economics

The basic LiNGAM assumption is that the structural shocks are mutually independent. This can be too strong in economic applications.

A simple example

Consider the linear SEM

$$X_1 = \varepsilon_1, \qquad X_2 = aX_1 + \varepsilon_2,$$

but suppose the shocks have the form

$$\varepsilon_1 = \sigma U_1, \qquad \varepsilon_2 = \sigma U_2,$$

where U_1, U_2 are independent, non-Gaussian; and $\sigma \geq 0$ is a random common component.

What happens?

Conditional on σ, the shocks are independent. Marginally, however, they are dependent.

Economic interpretation This is a crude model of common market volatility:

- shocks need not share a mean factor,
- but their size may co-move.

Other reasons independence may fail

- sectoral exposure to the same aggregate news,
- omitted low-frequency macro conditions,
- equilibrium or strategic dependence across units.

A weaker asymmetry than full independence

In economics, full mutual independence of shocks is often too restrictive. A more realistic goal is to look for weaker asymmetry conditions that still distinguish cause from effect.

One candidate: mean independence

Instead of requiring full independence, we assume residual satisfy $\mathbb{E}(\varepsilon_i | X_{\mathsf{pa}(i)}) = 0$.

This assumption allows to still recover the true causal direction.

This is a research direction rather than a standard off-the-shelf method. The lesson for us is that identifiability may survive under weaker assumptions than classical LiNGAM uses.

A toy simulation with common volatility

```r
set.seed(1)
n <- 2000

sigma <- abs(rnorm(n)) + 0.3
u1 <- rt(n, df = 5)
u2 <- rt(n, df = 5)

e1 <- sigma * u1
e2 <- sigma * u2
x1 <- e1
x2 <- 0.8 * x1 + e2
```

```r
par(mfrow = c(1,2))
plot(x1, x2, pch = 16, cex = .4)
plot(abs(e1), abs(e2), pch = 16,
     cex = .4)
```

Interpretation

The second plot reveals common volatility in the shocks. This violates strict independence, even though the directional SEM remains meaningful.

Part 5: A brief look beyond linearity

Alternative route: additive noise models

A different route to identifiability is nonlinear asymmetry.

Additive noise model

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

Under suitable regularity conditions, the reverse representation

$$X = g(Y) + \widetilde{\varepsilon}, \qquad \widetilde{\varepsilon} \perp\!\!\!\perp Y$$

usually does not exist.

Comparison with LiNGAM

LiNGAM assumes linearity but relies on non-Gaussianity and independence of the noice.

Additive noise models assume Gaussianity but rely on nonlinearity and independence of noise.

See Hoyer et al. (2009), *NIPS*, and Peters, Janzing, and Schölkopf, *Elements of Causal Inference*, Chapter 5.

Extension: graphical ideas for time series

From static SEMs to dynamic graphical models

For a multivariate time series, a natural model is

$$X_t = A_1 X_{t-1} + \cdots + A_p X_{t-p} + \varepsilon_t.$$

Directed lagged structure: Zeros in the matrices A_ℓ mean that some lagged variables do not directly predict others. This is the dynamic analogue of missing directed edges.

Contemporaneous structure: If $\varepsilon_t \sim N(0, \Sigma_\varepsilon)$, then zeros in $K_\varepsilon = \Sigma_\varepsilon^{-1}$ encode conditional independence among shocks at time t.

Interpretation

A graphical time-series model separates

- **lagged directional effects** from
- **instantaneous conditional dependence of shocks.**

Why economists may care about graphical time series

These ideas are natural in macroeconomics and finance.

Lagged graph
A sparse lagged graph can summarize who predicts whom:

- sectoral spillovers,
- macro transmission channels,
- Granger-causal relations in large VARs.

Shock graph A sparse graph for ε_t can summarize contemporaneous residual links:

- common financial stress,
- instantaneous market co-movement,
- residual dependence after lagged dynamics are removed.

Dynamic graphical models ask: who predicts whom over time, and which shocks still interact contemporaneously?

For sparse VARs and graphical time-series ideas, see Basu and Michailidis (2015), *Annals of Statistics*, and the `BigVAR` package paper by Nicholson et al. (2020), *Journal of Statistical Software*.

A simple workflow in R

1. estimate a sparse VAR for lagged structure,
2. compute fitted values and residuals,
3. fit a Gaussian graph to the residuals.

```
library(BigVAR)
# X is a T x m matrix of series
mod <- constructModel(X, p = 1,
                      struct = "Basic",
                      gran = c(10, 10),
                      verbose = FALSE)
fit <- cv.BigVAR(mod)
B <- as.matrix(coef(fit))
Z <- as.matrix(fit@lagmatrix)
Yhat <- t(B %*% Z)
```

```
Yobs <- tail(fit@Data, nrow(Yhat))
Ehat <- Yobs - Yhat
S <- cov(Ehat)

library(glasso)
out <- glasso(S, rho = 0.05)

library(qgraph)
qgraph(-cov2cor(out$wi),
       layout = "spring")
```

Interpretation

The sparse VAR captures lagged directed effects. The residual precision matrix then estimates contemporaneous conditional dependence among the shocks.

Take-away

- Constraint-based discovery uses conditional independence to recover a skeleton, v-structures, and a CPDAG.
- Markov equivalence: with no extra assumptions some arrows may not be identifiable.
- Linear Gaussian SEMs are symmetric enough that both directions may fit equally well.
- LiNGAM breaks that symmetry by using non-Gaussian independent shocks.
- Even when full independence is too strong, weaker asymmetry conditions may still carry identifying information.

Main lesson of today: conditional independence narrows down the graph, but full causal direction usually requires some additional asymmetry.

Looking ahead

Next lecture

We move from observed-variable causal graphs to **hidden variables and latent confounding**.

What comes next?

- why hidden confounders break the clean DAG story,
- MAGs and PAGs,
- instrumental variables as a benchmark workaround,
- proxy controls and latent-factor adjustments.

Today the challenge was to orient edges from observational data. Next time the challenge will be that some important variables are not observed at all.