Recursive Causal Discovery with Julia

11 July 2024

Sepehr Elahi[1]

[1] School of Computer and Communication Sciences, EPFL

Overview

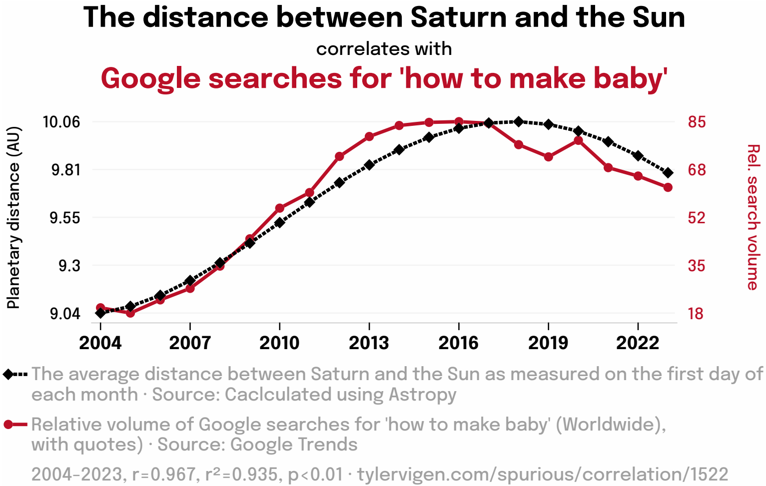
1. Causality
2. Causal discovery
3. Recursive causal discovery
4. Recursive causal discovery with Julia
5. Future direction

What is causality?

What is causality?

Causation is **NOT** correlation!

What is causality?

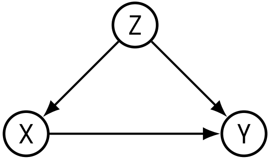


What is causality?

Causality: the relationship between cause and effect: cause $\rightarrow$ effect.

What is causality?

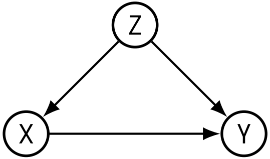
Causality: the relationship between cause and effect: cause $\rightarrow$ effect.



X: Education level
Y: Income
Z: Parental socioeconomic status

What is causality?

Causality: the relationship between cause and effect: cause $\rightarrow$ effect.

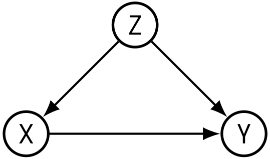


X: Education level
Y: Income
Z: Parental socioeconomic status

Research significance:

What is causality?

Causality: the relationship between cause and effect: cause $\rightarrow$ effect.



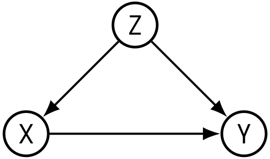
X: Education level
Y: Income
Z: Parental socioeconomic status

Research significance:

- Public health: Understanding if a new drug reduces the incidence of disease.

What is causality?

Causality: the relationship between cause and effect: cause $\to$ effect.



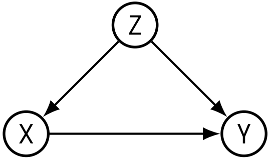
X: Education level
Y: Income
Z: Parental socioeconomic status

Research significance:

- Public health: Understanding if a new drug reduces the incidence of disease.
- Education: Assessing whether smaller class sizes improve student performance.

What is causality?

Causality: the relationship between cause and effect: cause $\to$ effect.



X: Education level
Y: Income
Z: Parental socioeconomic status

Research significance:

- Public health: Understanding if a new drug reduces the incidence of disease.
- Education: Assessing whether smaller class sizes improve student performance.
- Policy making: Evaluating if tax incentives stimulate business growth.

Causal graphs

Causal graphs: causal relationships are often represented using DAGs:

Causal graphs

Causal graphs: causal relationships are often represented using DAGs:
Directed: only directed edges

Causal graphs

Causal graphs: causal relationships are often represented using DAGs:
Directed: only directed edges   *Acyclic*: no cycles

Causal graphs

Causal graphs: causal relationships are often represented using DAGs:
Directed: only directed edges *Acyclic*: no cycles *Graph*: set of vertices and edges

Causal graphs

Causal graphs: causal relationships are often represented using DAGs:

Directed: only directed edges *Acyclic*: no cycles *Graph*: set of vertices and edges

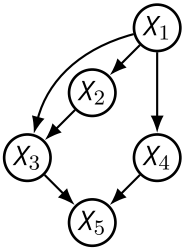


Figure: Causal graph $\mathcal{G}$

Causal graphs

Causal graphs: causal relationships are often represented using DAGs:
Directed: only directed edges   *Acyclic*: no cycles   *Graph*: set of vertices and edges

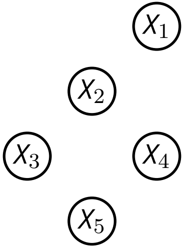


Figure: Causal graph $\mathcal{G}$

What if we don't know the causal graph?

Causal discovery

Causal discovery: the process of inferring causal relationships from data.

Causal discovery

Causal discovery: the process of inferring causal relationships from data.

Table: Observational data $\mathcal{D}$

X_1	X_2	X_3	X_4	X_5
1	0	1	0	1
0	1	0	1	0
1	1	1	0	0
0	0	1	1	1
1	0	0	1	0
		$\vdots$		

Causal discovery

Causal discovery: the process of inferring causal relationships from data.

Table: Observational data $\mathcal{D}$

X_1	X_2	X_3	X_4	X_5
1	0	1	0	1
0	1	0	1	0
1	1	1	0	0
0	0	1	1	1
1	0	0	1	0
		$\vdots$		

Common approach: PC algorithm

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.
Steps:

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.

Steps:

1. **Learn the skeleton:**

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.

Steps:

1. **Learn the skeleton:**
    - Test each pair of variables for conditional independence (CI) given a set of other variables.

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.

Steps:

1. **Learn the skeleton:**
    - Test each pair of variables for conditional independence (CI) given a set of other variables.
    - If $X \perp Y \mid Z \Rightarrow$, there is no edge between X and Y in $\mathcal{G}$.

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.

Steps:

1. **Learn the skeleton:**
    - Test each pair of variables for conditional independence (CI) given a set of other variables.
    - If $X \perp Y \mid Z \Rightarrow$, there is no edge between X and Y in $\mathcal{G}$.
    - Z starts empty and is iteratively grows.

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.

Steps:

1. **Learn the skeleton:**
    - Test each pair of variables for conditional independence (CI) given a set of other variables.
    - If $X \perp Y \mid Z \Rightarrow$, there is no edge between X and Y in $\mathcal{G}$.
    - Z starts empty and is iteratively grows.
2. **Orient edges:**
    - Apply Meek rules to orient the edges and form a partially directed acyclic graph (PDAG).

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.
Steps:

1. **Learn the skeleton:**
   - ☞ Test each pair of variables for conditional independence (CI) given a set of other variables. Very slow!
   - If $X \perp Y \mid Z \Rightarrow$, there is no edge between X and Y in $\mathcal{G}$.
   - Z starts empty and is iteratively grows.
2. **Orient edges:**
   - Apply Meek rules to orient the edges and form a partially directed acyclic graph (PDAG).

PC can be very slow!

PC Algorithm [Spirtes et al., 2000]

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$.

Steps:

1. **Learn the skeleton:**
   ☞ Test each pair of variables for conditional independence (CI) given a set of other variables. Very slow!
   - If $X \perp Y \mid Z \Rightarrow$, there is no edge between X and Y in $\mathcal{G}$.
   - Z starts empty and is iteratively grows.
2. **Orient edges:**
   - Apply Meek rules to orient the edges and form a partially directed acyclic graph (PDAG).

PC can be very slow! Exponential complexity: $O(n^2 2^n)$!

Problem with the PC Algorithm

PC has exponential complexity: PC requires potentially conditioning on every subset of variables: $O(n^2 2^n)$.

Recursive causal discovery

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$ *faster* by recursively removing variables and learning their neighbors.

Recursive causal discovery

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$ *faster* by recursively removing variables and learning their neighbors.

Steps:

Recursive causal discovery

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$ *faster* by recursively removing variables and learning their neighbors.

Steps:

- **Learn the skeleton:**

Recursive causal discovery

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$ *faster* by recursively removing variables and learning their neighbors.

Steps:

- **Learn the skeleton:**
  1. Identify a removable variable X using CI tests.

Recursive causal discovery

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$ *faster* by recursively removing variables and learning their neighbors.

Steps:

- **Learn the skeleton:**
    1. Identify a removable variable X using CI tests.
    2. Learn the neighbors of X using CI tests.

Recursive causal discovery

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$ *faster* by recursively removing variables and learning their neighbors.

Steps:

- **Learn the skeleton:**
    1. Identify a removable variable X using CI tests.
    2. Learn the neighbors of X using CI tests.
    3. Remove X from the graph. Go to step 1.

Recursive causal discovery

Goal: learn causal graph $\mathcal{G}$ from observational data $\mathcal{D}$ *faster* by recursively removing variables and learning their neighbors.

Steps:

- **Learn the skeleton:**
  1. Identify a removable variable X using CI tests.
  2. Learn the neighbors of X using CI tests.
  3. Remove X from the graph. Go to step 1.
- **Orient edges:**
  - Apply Meek rules to orient the edges and form a partially directed acyclic graph (PDAG).

RCD: example

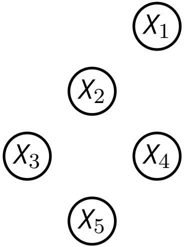


Figure: Remaining variables

Figure: Learned skeleton so far by RCD

RCD: example

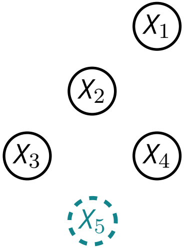


Figure: Remaining variables

Figure: Learned skeleton so far by RCD

RCD: example

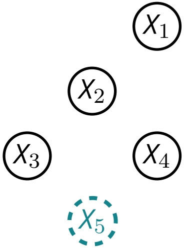


Figure: Remaining variables

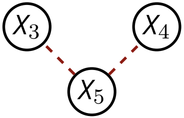


Figure: Learned skeleton so far by RCD

RCD: example

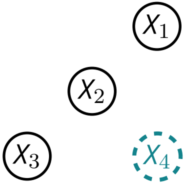


Figure: Remaining variables

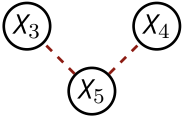


Figure: Learned skeleton so far by RCD

RCD: example

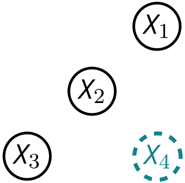


Figure: Remaining variables

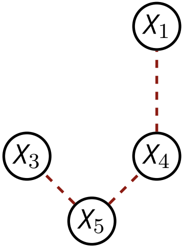


Figure: Learned skeleton so far by RCD

RCD: example

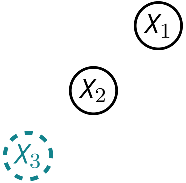


Figure: Remaining variables

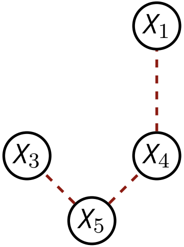


Figure: Learned skeleton so far by RCD

RCD: example

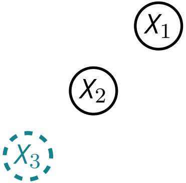


Figure: Remaining variables

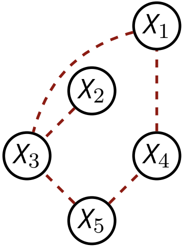


Figure: Learned skeleton so far by RCD

RCD: example

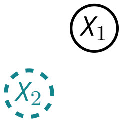


Figure: Remaining variables

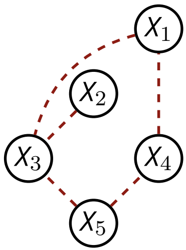


Figure: Learned skeleton so far by RCD

RCD: example

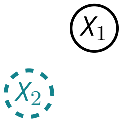


Figure: Remaining variables

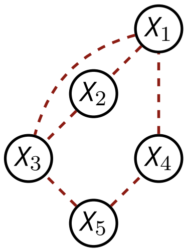


Figure: Learned skeleton so far by RCD

RCD: example

Figure: Remaining variables

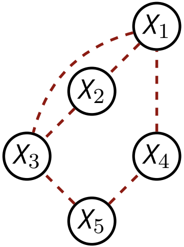


Figure: Learned skeleton so far by RCD

RCD: algorithms

Algorithm	Completeness	#CI tests
MARVEL	YES	$\mathcal{O}(n^2 + n\Delta_{in}^2 2^{\Delta_{in}})$
L-MARVEL	YES	$\mathcal{O}(n^2 + n(\Delta_{in}^+)^2 2^{\Delta_{in}^+})$
RSL	YES	$\mathcal{O}(n^2 + n\Delta_{in}^{m+1})$
ROL	NO	$\mathcal{O}(\textsc{MaxIter} \times n^3)$
PC	YES	$\mathcal{O}(n^2 2^n)$

RCD: algorithms

Algorithm	Completeness	#CI tests
MARVEL	YES	$\mathcal{O}(n^2 + n\Delta_{in}^2 2^{\Delta_{in}})$
L-MARVEL	YES	$\mathcal{O}(n^2 + n(\Delta_{in}^+)^2 2^{\Delta_{in}^+})$
RSL	YES	$\mathcal{O}(n^2 + n\Delta_{in}^{m+1})$
ROL	NO	$\mathcal{O}(\text{MAXITER} \times n^3)$
PC	YES	$\mathcal{O}(n^2 2^n)$

For more details, see our paper at go.epfl.ch/rcd.

RCD in Python

rcd package: Recursive causal discovery in Python.

RCD in Python

rcd package: Recursive causal discovery in Python.
Implemented algorithms:

- MARVEL
- L-MARVEL
- RSL
- ROL

RCD in Python

rcd package: Recursive causal discovery in Python.

Implemented algorithms:

- MARVEL
- L-MARVEL
- RSL
- ROL

Getting started:

`rcdpackage.com`

Installation:

```
pip install rcd
```

RCD in Python

rcd package: Recursive causal discovery in Python.

Implemented algorithms:

- MARVEL
- L-MARVEL
- RSL
- ROL

Getting started:

~~`rcdpackage.com`~~

Installation:

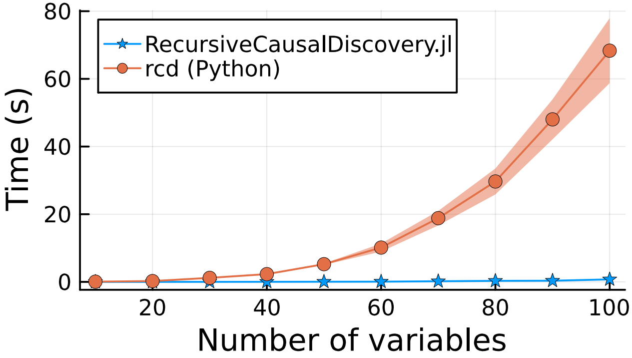
~~`pip install rcd`~~

RCD implementation in Julia versus Python

Comparison: RecursiveCausalDiscovery.jl vs. rcd on learning graphs from synthetic data.

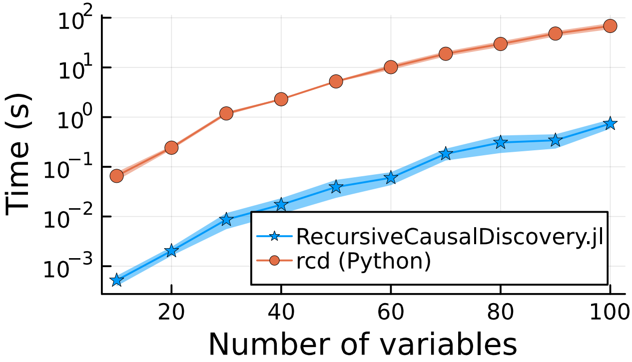
RCD implementation in Julia versus Python

Comparison: RecursiveCausalDiscovery.jl vs. rcd on learning graphs from synthetic data.



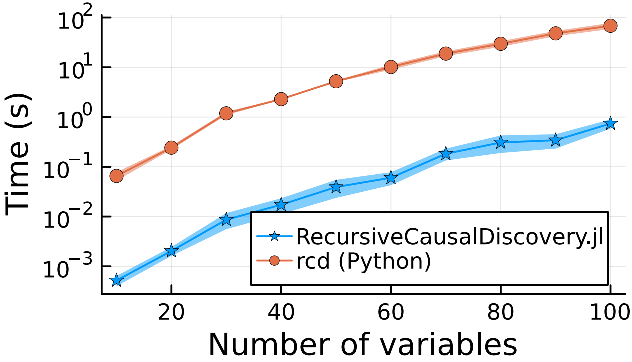
RCD implementation in Julia versus Python

Comparison: RecursiveCausalDiscovery.jl vs. rcd on learning graphs from synthetic data.



RCD implementation in Julia versus Python

Comparison: RecursiveCausalDiscovery.jl vs. rcd on learning graphs from synthetic data.



Julia is faster by a factor of 150!

RCD in Julia

RecursiveCausalDiscovery.jl: Recursive causal discovery in Julia.

RCD in Julia

RecursiveCausalDiscovery.jl: Recursive causal discovery in Julia.
Installation:

```
] add RecursiveCausalDiscovery
```

RCD in Julia

Simple demo:

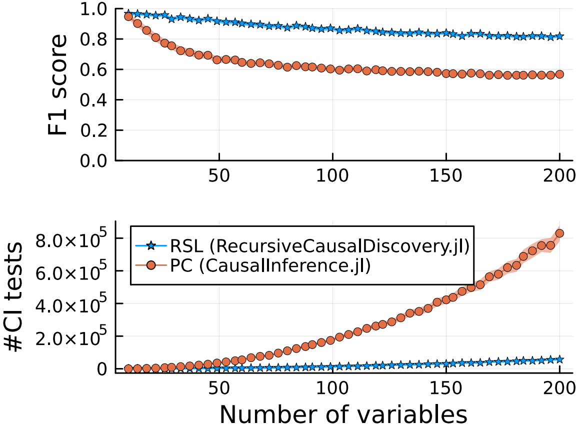
```julia
using RecursiveCausalDiscovery
using CSV
using Tables

# load data (columns are variables and rows are samples)
data = CSV.read("data.csv", Tables.matrix)

# use a Gaussian conditional independence test
sig_level =  0.01
ci_test = (x, y, cond_vec, data) ->  fisher_z(x, y, cond_vec, data, sig_level)

# learn the skeleton of causal graph using RSL
learned_skeleton = learn_and_get_skeleton(data, ci_test)
```

RCD versus PC

Comparison: RCD vs. PC on learning graphs from synthetic data.

RCD versus PC

Comparison: RCD vs. PC on learning graphs from synthetic data.



Future direction

- Implement the other RCD algorithms (MARVEL, L-MARVEL, ROL).

Future direction

- Implement the other RCD algorithms (MARVEL, L-MARVEL, ROL).
- Add more CI tests.

Future direction

- Implement the other RCD algorithms (MARVEL, L-MARVEL, ROL).
- Add more CI tests.
- Integrate with CausalInference.jl.