Causal Inference Methods in Data Science Lecture 4: Advanced Graphical Models: MEC, CPDAG, ADMG, Causal Discovery and Tian's ID algorithm

Lin Liu

July 4, 2022

Preface

Most causal inference researchers in statistics do not understand graphical models to the level of doing creative research in this field but obviously people started to realize the importance of better combining causal graphs with statistical inference around about 2019

After this lecture, you could read the following people's recent papers if want to work on this field (without particular ordering):
Ilya Shpitser, Elias Barenboim, Thomas Richardson, Robin Evans, F. Richard Guo, Emilija Perković, Marloes Maathius, Yangbo He, Jiji Zhang, Jin Tian, Steffan Lauritzen, Samuel Wang, Caroline Uhler

Why causal graphs?

- As we have seen, causal inference relies heavily on background knowledge
- Causal graph is a very succinct way of representing background knowledge
- We have seen some examples: backdoor, frontdoor
- But ...

Motivating problems

- In certain applications, we do not have much background knowledge to begin with. But it might be easy to collect data (e.g. in biology). Can we learn background knowledge in the form of causal graph from data?

- If the causal graph is very complicated (like below), how can you identify any given causal query (in terms of counterfactuals)?

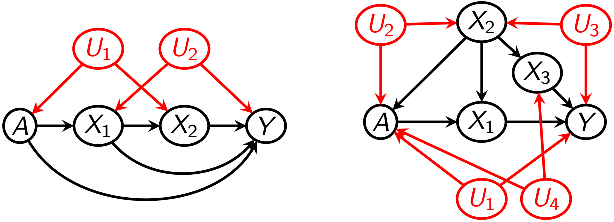


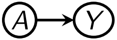
Figure: Is $p(Y(a))$ identifiable in the above two graphs?

- Finally, in complex causal graphs, the same causal query might have different identified formula. Which one should we use?

Causal discovery or structure learning: From data to causal graphs

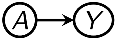
Markov equivalence class (MEC) of a causal DAG

- Suppose the underlying but unknown causal DAG $\mathcal{G} = (V, E)$ is



Markov equivalence class (MEC) of a causal DAG

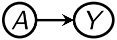
- Suppose the underlying but unknown causal DAG $\mathcal{G} = (V, E)$ is



- Qn: if we only observe $(A_i, Y_i)_{i=1}^n$, can we recover the above graph? Or equivalently, can we distinguish the following three structures? (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad\quad Y$

Markov equivalence class (MEC) of a causal DAG

- Suppose the underlying but unknown causal DAG $\mathcal{G} = (V, E)$ is

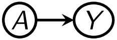


- Qn: if we only observe $(A_i, Y_i)_{i=1}^n$, can we recover the above graph? Or equivalently, can we distinguish the following three structures? (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad\quad Y$

- Assuming faithfulness: if $A \rightarrow Y$ or $A \leftarrow Y$, then A and Y are dependent (i.e. not d-separated by Z implies not independent conditional on Z)

Markov equivalence class (MEC) of a causal DAG

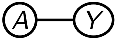
- Suppose the underlying but unknown causal DAG $\mathcal{G} = (V, E)$ is

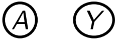


- Qn: if we only observe $(A_i, Y_i)_{i=1}^n$, can we recover the above graph? Or equivalently, can we distinguish the following three structures? (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad Y$

- Assuming faithfulness: if $A \rightarrow Y$ or $A \leftarrow Y$, then A and Y are dependent (i.e. not d-separated by Z implies not independent conditional on Z)

- Without modeling assumptions, one cannot infer the direction so the discovered graph is one of the following MECs (represented by PDAGs) of DAGs between A and Y:





Countering MEC?

The conclusion that we cannot distinguish between $A \longrightarrow Y$ and $A \longleftarrow Y$ is a “nonparametric” statement, in the following sense:

For any distribution P Markov factorized according to $A \longrightarrow Y$, one can always find a distribution Q Markov factorized according to $A \longleftarrow Y$ such that $P \sim Q$

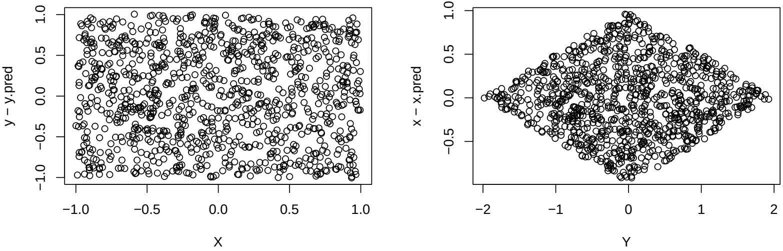
But it does not rule out the possibility of distinguishing DAGs in one MEC by imposing more modeling assumptions: e.g. linear non-Gaussian models

Distinguishable orientation based on Linear non-Gaussian models

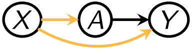
Let's assume the following SEM:

$$A \sim \text{Unif}([-1,1]), Y \sim A + \text{Unif}([-1,1])$$

We could fit two linear regressions $\text{lm}(Y \sim A - 1)$ and $\text{lm}(A \sim Y - 1)$ and look at their residual plots:

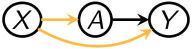


Example 1



Let's say we have known all the yellow edges, but we want to distinguish (1) $A \to Y$; (2) $A \leftarrow Y$; (3) $A \quad Y$

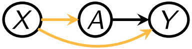
Example 1



Let's say we have known all the yellow edges, but we want to distinguish (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad Y$

Guess what to do?

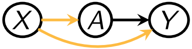
Example 1

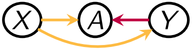


Let's say we have known all the yellow edges, but we want to distinguish (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad\quad Y$

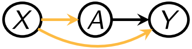
Guess what to do?

1) Testing conditional independence $Y \perp\!\!\!\perp A | X$:
   if accept, then $A \quad\quad Y$; else, then $A \rightarrow Y$ or $A \leftarrow Y$





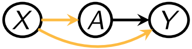
Example 1

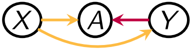


Let's say we have known all the yellow edges, but we want to distinguish (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad Y$

Guess what to do?

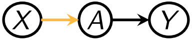
1) Testing conditional independence $Y \perp\!\!\!\perp A|X$:
   if accept, then $A \quad Y$; else, then $A \rightarrow Y$ or $A \leftarrow Y$





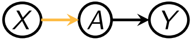
2) No other independence/conditional independence tests can further orient the uncertainty

Example 2



Let's say we have known all the yellow edges, but we want to distinguish (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad Y$

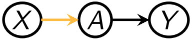
Example 2



Let's say we have known all the yellow edges, but we want to distinguish (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad Y$

Guess what to do?

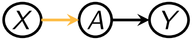
Example 2

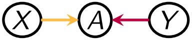


Let's say we have known all the yellow edges, but we want to distinguish (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad Y$

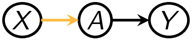
Guess what to do?

1) Testing independence $Y \perp\!\!\!\perp A$:
   if accept, then $A \quad Y$; else, then $A \rightarrow Y$ or $A \leftarrow Y$





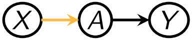
Example 2

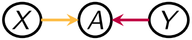


Let's say we have known all the yellow edges, but we want to distinguish (1) $A \rightarrow Y$; (2) $A \leftarrow Y$; (3) $A \quad Y$

Guess what to do?

1) Testing independence $Y \perp\!\!\!\perp A$:
   if accept, then $A \quad Y$; else, then $A \rightarrow Y$ or $A \leftarrow Y$





2) Testing conditional independence $Y \perp\!\!\!\perp X|A$:
   if accept, then $A \rightarrow Y$; else $A \leftarrow Y$

Conditional independence testing

A classical problem in statistics, and DIFFICULT!

Conditional independence testing

A classical problem in statistics, and DIFFICULT!

Active area of research in mathematical statistics; many different ideas

Conditional independence testing

A classical problem in statistics, and DIFFICULT!

Active area of research in mathematical statistics; many different ideas

But impossibility results by Jonas Peters and Rajen Shah AoS 2020 and Neykov, Balakrishnan, Wasserman AoS 2022 when the variable being conditioned on is continuous

High level definition of MEC

High level definition (what this definition is trying to accomplish?):

High level definition of MEC

High level definition (what this definition is trying to accomplish?):

Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that one cannot distinguish among members in $[\mathcal{G}]$ with only observational data

Operational definition of MEC

Operational definition (precise mathematical translation of high level definition):

Operational definition of MEC

Operational definition (precise mathematical translation of high level definition):

Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that every member of $[\mathcal{G}]$ shares the same independence and conditional independence constraints, or equivalently the same

Operational definition of MEC

Operational definition (precise mathematical translation of high level definition):

Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that every member of $[\mathcal{G}]$ shares the same independence and conditional independence constraints, or equivalently the same

1) skeletons (turning all directed edges into undirected edges)

Operational definition of MEC

Operational definition (precise mathematical translation of high level definition):

Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that every member of $[\mathcal{G}]$ shares the same independence and conditional independence constraints, or equivalently the same

1) skeletons (turning all directed edges into undirected edges)
2) d-separation constraints

Constructive definition of MEC

Constructive definition (need a proof that shows Operational $\Leftrightarrow$ Constructive):

Constructive definition of MEC

Constructive definition (need a proof that shows Operational $\Leftrightarrow$ Constructive):

Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that every member of $[\mathcal{G}]$ shares the same

Constructive definition of MEC

Constructive definition (need a proof that shows Operational ⇔ Constructive):

Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that every member of $[\mathcal{G}]$ shares the same

1) skeletons

Constructive definition of MEC

Constructive definition (need a proof that shows Operational $\Leftrightarrow$ Constructive):

Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that every member of $[\mathcal{G}]$ shares the same

1) skeletons
2) v-structures ($V_1 \rightarrow V_3 \leftarrow V_2$)

Constructive definition of MEC

Constructive definition (need a proof that shows Operational $\Leftrightarrow$ Constructive):

Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that every member of $[\mathcal{G}]$ shares the same

1) skeletons
2) v-structures ($V_1 \rightarrow V_3 \leftarrow V_2$)

This was proved in:
REF: Verma, Pearl. On the Equivalence of Causal Models. UAI 1990

Constructive definition of MEC

Constructive definition (need a proof that shows Operational $\Leftrightarrow$ Constructive):

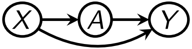
Markov equivalence class (MEC) of a causal DAG $\mathcal{G}_0$: a set of causal DAGs $[\mathcal{G}]$ containing $\mathcal{G}_0$ such that every member of $[\mathcal{G}]$ shares the same

1) skeletons
2) v-structures ($V_1 \rightarrow V_3 \leftarrow V_2$)

This was proved in:
REF: Verma, Pearl. On the Equivalence of Causal Models. UAI 1990
WARNING: v-structure means $V_1 \rightarrow V_3 \leftarrow V_2$ and there shall be no arrows between V_1 and V_2; v-structure is also called "unshielded collider"

Examples

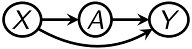
MEC of the following graph $\mathcal{G}_0$ (note: no v-structure)



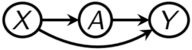
How many in total? Which should be excluded? How many are left?

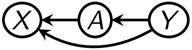
Examples

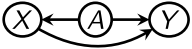
MEC of the following graph $\mathcal{G}_0$ (note: no v-structure)

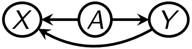


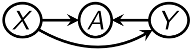
How many in total? Which should be excluded? How many are left?

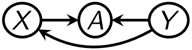












Completed Partially DAG (CPDAG)

- MEC is a set of DAGs: when there are many, inconvenient to write down

Completed Partially DAG (CPDAG)

- MEC is a set of DAGs: when there are many, inconvenient to write down

- A more succinct representation? CPDAG (or essential graphs)

Completed Partially DAG (CPDAG)

- MEC is a set of DAGs: when there are many, inconvenient to write down
- A more succinct representation? CPDAG (or essential graphs)
- Construction rule: denote CPDAG of $\mathcal{G}$ as $\mathcal{C}$:

Completed Partially DAG (CPDAG)

- MEC is a set of DAGs: when there are many, inconvenient to write down

- A more succinct representation? CPDAG (or essential graphs)

- Construction rule: denote CPDAG of $\mathcal{G}$ as $\mathcal{C}$:
  1) If there is at least one $X \rightarrow Y$ and at least one $X \leftarrow Y$ in $[\mathcal{G}]$, then $X - Y$ in $\mathcal{C}$

Completed Partially DAG (CPDAG)

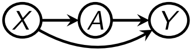
- MEC is a set of DAGs: when there are many, inconvenient to write down

- A more succinct representation? CPDAG (or essential graphs)

- Construction rule: denote CPDAG of $\mathcal{G}$ as $\mathcal{C}$:
  1) If there is at least one $X \rightarrow Y$ and at least one $X \leftarrow Y$ in $[\mathcal{G}]$, then $X - Y$ in $\mathcal{C}$
  2) If $X \rightarrow Y$ is in every DAG in $[\mathcal{G}]$, then $X \rightarrow Y$ in $\mathcal{C}$

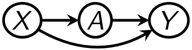
Example

CPDAG of

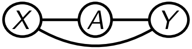


Example

CPDAG of



is



How to reduce uncertainty of a CPDAG?

- With purely observational data (no interventions), in the worst case, the best one can do is to recover the CPDAG assuming no hidden variables

How to reduce uncertainty of a CPDAG?

- With purely observational data (no interventions), in the worst case, the best one can do is to recover the CPDAG assuming no hidden variables

- How can we reduce uncertainty of such a CPDAG?

How to reduce uncertainty of a CPDAG?

- With purely observational data (no interventions), in the worst case, the best one can do is to recover the CPDAG assuming no hidden variables

- How can we reduce uncertainty of such a CPDAG?
  1) Imposing more modeling assumptions: e.g. non-Gaussian errors may orient the direction of an arrow with only pair of observations
     REF: Shimizu et al. A linear non-Gaussian acyclic model for causal discovery. JMLR 2006.
     REF: Wang, Drton. Causal discovery with unobserved confounding and non-Gaussian data. 2021.

How to reduce uncertainty of a CPDAG?

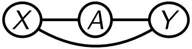
- With purely observational data (no interventions), in the worst case, the best one can do is to recover the CPDAG assuming no hidden variables

- How can we reduce uncertainty of such a CPDAG?
  1) Imposing more modeling assumptions: e.g. non-Gaussian errors may orient the direction of an arrow with only pair of observations
     REF: Shimizu et al. A linear non-Gaussian acyclic model for causal discovery. JMLR 2006.
     REF: Wang, Drton. Causal discovery with unobserved confounding and non-Gaussian data. 2021.
  2) Background knowledge or doing intervention

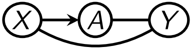
From CPDAG to MPDAG

The power of background knowledge or simply doing intervention (action in reinforcement learning):

Example:



if we additionally have background knowledge $X \to A$, then we have the following Maximal PDAG (MPDAG), denoted as $\mathcal{M}$



how about we intervene X? can we have more precise MPDAG?

Research frontier: fundamental open problems

With all the above definitions, we can ask the following fundamental questions

1) Given purely observational data, what algorithm can recover the CPDAG? (PC algorithm)

Research frontier: fundamental open problems

With all the above definitions, we can ask the following fundamental questions

1) Given purely observational data, what algorithm can recover the CPDAG? (PC algorithm)

2) Given a DAG/CPDAG, what causal query can be identified? (see Elias Bareinboim's recent papers)

Research frontier: fundamental open problems

With all the above definitions, we can ask the following fundamental questions

1) Given purely observational data, what algorithm can recover the CPDAG? (PC algorithm)

2) Given a DAG/CPDAG, what causal query can be identified? (see Elias Bareinboim's recent papers)

3) Given identifiable causal query, what is the optimal identification formula? (Rotnitzky and Smucler, JMLR 2020)

Research frontier: fundamental open problems

With all the above definitions, we can ask the following fundamental questions

1) Given purely observational data, what algorithm can recover the CPDAG? (PC algorithm)

2) Given a DAG/CPDAG, what causal query can be identified? (see Elias Bareinboim's recent papers)

3) Given identifiable causal query, what is the optimal identification formula? (Rotnitzky and Smucler, JMLR 2020)

4) 1), 2), 3) with interventional data (or background knowledge) and/or allowing for latent factors

Q1). Peter-Clark (PC) algorithm

- Invented by Peter Spirtes and Clark Glymour

Q1). Peter-Clark (PC) algorithm

- Invented by Peter Spirtes and Clark Glymour

- Ground rule 1: $X \quad Y$ (no edge) if and only if there exists $\boldsymbol{S}_{X,Y} \subseteq V \setminus \{X, Y\}$ such that

$$X \perp\!\!\!\perp Y | \boldsymbol{S}_{X,Y}$$

Q1). Peter-Clark (PC) algorithm

- Invented by Peter Spirtes and Clark Glymour

- Ground rule 1: $X \quad Y$ (no edge) if and only if there exists $\boldsymbol{S}_{X,Y} \subseteq V \setminus \{X, Y\}$ such that

$$X \perp\!\!\!\perp Y | \boldsymbol{S}_{X,Y}$$

- Ground rule 2: if $X \quad Y$ but $X \rightarrow Z \leftarrow Y$ (v-structure), then

$$Z \notin \boldsymbol{S}_{X,Y}$$

PC algorithm: vanilla version

Initialize a complete undirected graph

1) For every pair (X, Y), if there exists $\boldsymbol{S}_{X,Y} \subseteq V \setminus \{X, Y\}$ such that

$$X \perp\!\!\!\perp Y | \boldsymbol{S}_{X,Y},$$

remove the edge in $X - Y$

PC algorithm: vanilla version

Initialize a complete undirected graph

1) For every pair (X, Y), if there exists $\boldsymbol{S}_{X,Y} \subseteq V \setminus \{X, Y\}$ such that

$$X \perp\!\!\!\perp Y | \boldsymbol{S}_{X,Y},$$

   remove the edge in $X - Y$

2) If there exists $X - Z - Y$ but $X \quad Y$, and $Z \notin \boldsymbol{S}_{X,Y}$, then orient $X - Z - Y$ to $X \rightarrow Z \leftarrow Y$

PC algorithm: vanilla version

Initialize a complete undirected graph

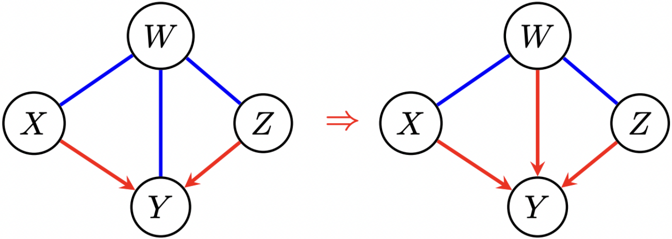
1) For every pair (X, Y), if there exists $\boldsymbol{S}_{X,Y} \subseteq V \setminus \{X, Y\}$ such that

$$X \perp\!\!\!\perp Y | \boldsymbol{S}_{X,Y},$$

remove the edge in $X - Y$

2) If there exists $X - Z - Y$ but $X \quad Y$, and $Z \notin \boldsymbol{S}_{X,Y}$, then orient $X - Z - Y$ to $X \rightarrow Z \leftarrow Y$

3) Apply the following sub-rules:
    i) $X \quad Z$ and $X \rightarrow Y - Z \Rightarrow X \rightarrow Y \rightarrow Z$ [due to 2) and logic]
    ii) $X \rightarrow Y \rightarrow Z$ and $X - Z \Rightarrow X \rightarrow Z$ [acyclicity]
    iii) $X \quad Z$, $X - W - Z$, $X \rightarrow Y \leftarrow Z$ and $W - Y \Rightarrow W \rightarrow Y$ [why?]



Why not $Y \rightarrow W$? If it were the case, then also need to orient $X \rightarrow W$ and $Z \rightarrow W$ by sub-rule 3):ii) so we have $X \rightarrow W \leftarrow Z$ and $X \quad Z$

But we should have had oriented such v-structure in step 2)

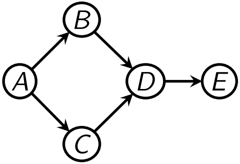
PC algorithm: improve computational efficiency

- Intractable: for each pair (X, Y), if there are p vertices in the graph, then to find $\boldsymbol{S}_{X,Y}$ one needs to enumerate over 2^{p-2} possible sets of vertices

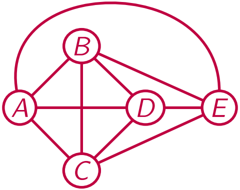
PC algorithm: improve computational efficiency

- Intractable: for each pair (X, Y), if there are p vertices in the graph, then to find $\boldsymbol{S}_{X,Y}$ one needs to enumerate over 2^{p-2} possible sets of vertices
- See the following paper on how to improve computational efficiency
  REF: Computation, causation and discovery
  Main idea: searching for $\boldsymbol{S}_{X,Y}$, starting from empty set, increasing the cardinality one by one

Example



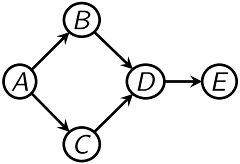
Unknown true graph



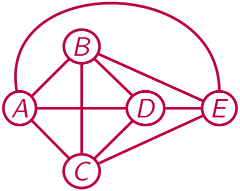
Initial complete undirected graph

Check marginal independencies

No pairs are independent marginally. Nothing changed



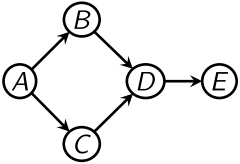
Unknown true graph



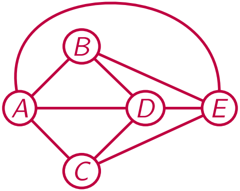
Initial complete undirected graph

Check conditional independencies

$B \perp\!\!\!\perp C | A$

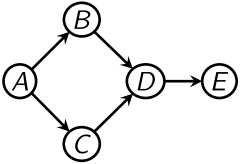


Unknown true graph

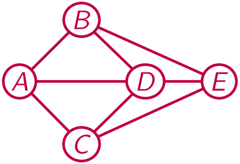


Check conditional independencies

$A \perp\!\!\!\perp E | D$

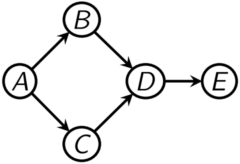


Unknown true graph

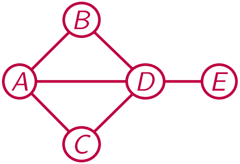


Check conditional independencies

$B \perp\!\!\!\perp E|D$, $C \perp\!\!\!\perp E|D$

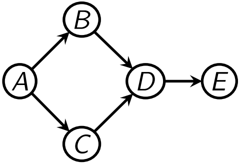


Unknown true graph

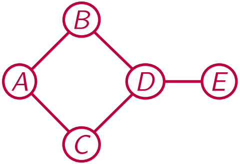


Check conditional independencies

$A \perp\!\!\!\perp D | \{B, C\}$



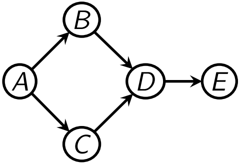
Unknown true graph



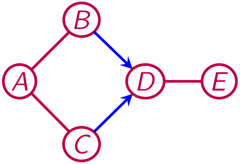
Skeleton of the unknown true graph

Orientation of edges in the skeleton

Rule 2) (about v-structure)



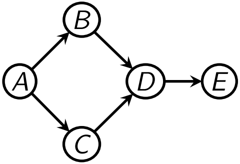
Unknown true graph



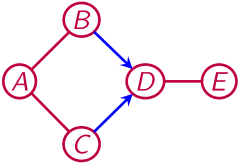
Orientation of the v-structure

Orientation of edges in the skeleton

Rule 2) (about v-structure)

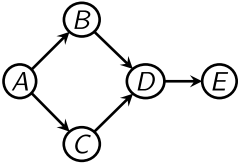


Unknown true graph

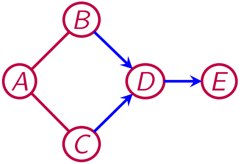


Orientation of the v-structure

Rule 3):i)

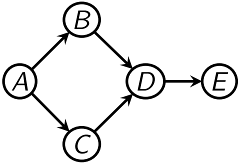


Unknown true graph

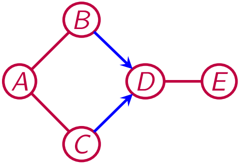


Orientation of edges in the skeleton

Rule 2) (about v-structure)

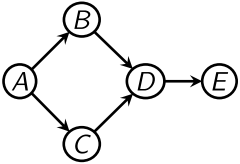


Unknown true graph

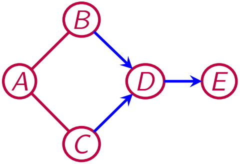


Orientation of the v-structure

Rule 3):i)



Unknown true graph



Do we need to consider Rule 3):iii)?

Theoretical guarantee

Spirtes and Glymour showed consistency of PC algorithm:

Theorem 1

Under the faithfulness assumption, with input data $V_i, i = 1, \cdots, n$, the output of the PC algorithm $\widehat{\mathcal{C}}$ converges to the true CPDAG $\mathcal{C}$ of the underlying unknown DAG $\mathcal{G}$, as $n \to \infty$.

Other algorithms

- As always, there are almost no problems that can only be solved by one algorithm

Other algorithms

- As always, there are almost no problems that can only be solved by one algorithm
- Greedy Equivalence Search (Chickering 2002; Chickering & Meek 2002)

Other algorithms

- As always, there are almost no problems that can only be solved by one algorithm

- Greedy Equivalence Search (Chickering 2002; Chickering & Meek 2002)

- BIC-score based search (Madigan & Raftery, 1994)

Other algorithms

- As always, there are almost no problems that can only be solved by one algorithm

- Greedy Equivalence Search (Chickering 2002; Chickering & Meek 2002)

- BIC-score based search (Madigan & Raftery, 1994)

- Maathuis et al. Nature Methods 2009: IDA algorithm (but assuming linear models so bummer)
  By far, the most influential work in causal discovery in applications (genomics).
  But in genomics, whether it provides significant gain compared to existing correlation/partial correlation based method still remains to be seen.

More recent works

- First continuous optimization framework for linear model DAG learning, see Zheng et al. NeurIPS 2018 (DAGs with NoTear

More recent works

- First continuous optimization framework for linear model DAG learning, see Zheng et al. NeurIPS 2018 (DAGs with NoTear
- Graph neural networks with DAG learning, see Yu et al. ICML 2019

More recent works

- First continuous optimization framework for linear model DAG learning, see Zheng et al. NeurIPS 2018 (DAGs with NoTear
- Graph neural networks with DAG learning, see Yu et al. ICML 2019
- Continuous optimization framework for single-index model DAG learning, see Yu et al. ICML 2021 (DAGs with no curl)

Further orienting the uncertain edges

- Easy if we have further background knowledge: CPDAG $\Rightarrow$ MPDAG

Further orienting the uncertain edges

- Easy if we have further background knowledge: CPDAG $\Rightarrow$ MPDAG
- Nontrivial if we have interventional ($\mathcal{I}$) data: MEC $\Rightarrow$ $\mathcal{I}$-MEC

Further orienting the uncertain edges

- Easy if we have further background knowledge: CPDAG $\Rightarrow$ MPDAG
- Nontrivial if we have interventional ($\mathcal{I}$) data: MEC $\Rightarrow$ $\mathcal{I}$-MEC
- Once we consider interventional data, many new directions/problems suddenly appear!
  1) Hard or soft intervention?
  2) Are the intervened vertices known or unknown to us?
  3) Is the intervention error-prone? (e.g. CRISPR-based gene knockdown techniques are known to have off-target effects)
  4) Are latent variables allowed?

Further orienting the uncertain edges

- Easy if we have further background knowledge: CPDAG $\Rightarrow$ MPDAG

- Nontrivial if we have interventional ($\mathcal{I}$) data: MEC $\Rightarrow$ $\mathcal{I}$-MEC

- Once we consider interventional data, many new directions/problems suddenly appear!
  1) Hard or soft intervention?
  2) Are the intervened vertices known or unknown to us?
  3) Is the intervention error-prone? (e.g. CRISPR-based gene knockdown techniques are known to have off-target effects)
  4) Are latent variables allowed?

- Hauser and Bühlmann JMLR 2012 proved: Without hidden variables, given an intervention target sets $\mathcal{I}$, with correct and hard intervention, two DAGs $\mathcal{G}_1$ and $\mathcal{G}_2$ belong to the same $\mathcal{I}$-MEC if and only if for every $I \in \mathcal{I}$, $\mathcal{G}_1(I) \sim \mathcal{G}_2(I)$ in observational sense

R Exercise

- Implementing pcalg

Q2). ID algorithm

- For DAG without hidden variables, identification is easy by Robins' g-formula

$$p(Y(a)=y)=\int \prod_{i\in \mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)} p_i(v_i|\mathrm{pa}_{\mathcal{G}}(v_i)) \prod_{i\in \mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)\setminus Y} \mathrm{d}v_i$$

Q2). ID algorithm

- For DAG without hidden variables, identification is easy by Robins' g-formula

$$p(Y(a)=y)=\int\prod_{i\in\mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)}p_i(v_i|\mathrm{pa}_{\mathcal{G}}(v_i))\prod_{i\in\mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)\setminus Y}\mathrm{d}v_i$$

- For CPDAG/MPDAG, Perković 2020 UAI gave the identification formula

Q2). ID algorithm

- For DAG without hidden variables, identification is easy by Robins' g-formula

$$p(Y(a) = y) = \int \prod_{i \in \mathrm{an}_{\mathcal{G}_{V \setminus A}}(Y)} p_i(v_i | \mathrm{pa}_{\mathcal{G}}(v_i)) \prod_{i \in \mathrm{an}_{\mathcal{G}_{V \setminus A}}(Y) \setminus Y} \mathrm{d}v_i$$

- For CPDAG/MPDAG, Perković 2020 UAI gave the identification formula

- The intuition is clear: since CPDAG/MPDAG involves undirected edges, decompose the vertices $V = \cup_{j=1}^{k} \boldsymbol{B}_j$ into smaller units, which is called "bucket" by Perković: in graph-theoretic terms, bucket is the maximally connected component by undirected edges

  For DAGs, buckets are all the singletons of vertices

Q2). ID algorithm

- For DAG without hidden variables, identification is easy by Robins' g-formula

$$p(Y(a)=y)=\int \prod_{i\in \mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)} p_i(v_i|\mathrm{pa}_{\mathcal{G}}(v_i)) \prod_{i\in \mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)\setminus Y} \mathrm{d}v_i$$

- For CPDAG/MPDAG, Perković 2020 UAI gave the identification formula

Theorem 2 (Theorem 3.6 of Perković 2020 UAI)

If there is no path $\langle A, V_1, \cdots, V_k, \cdots, Y\rangle$ from A to Y without edge $V_j \to V_i$ for any $j > i$ in $\mathcal{G}$ starting with $A - \cdots$, then

$$p(Y(a)=y)=\int \prod_{j=1}^{k} p(\boldsymbol{b}_j|\mathrm{pa}_{\mathcal{G}}(\boldsymbol{b}_j))\mathrm{d}\bar{\boldsymbol{b}}$$

where buckets $\boldsymbol{B}_j$'s are buckets of $\mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)$ and $\bar{\boldsymbol{B}}=\mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)\setminus Y$

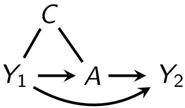
Q2). ID algorithm

Paths like $\langle A, V_1, \cdots, V_k, \cdots, Y \rangle$ without edge $V_j \rightarrow V_i$ for any $j > i$ in $\mathcal{G}$ are called "possibly causal paths". Why excluding "possibly causal paths" that start with $A - \cdots$?

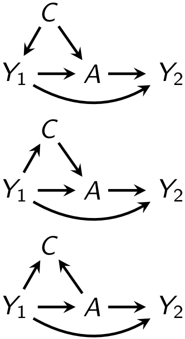
$$A — Y$$

This is a possibly causal path from A to Y, with the first edge being undirected. Why we cannot identify the causal effect of A on Y?

Q2). ID algorithm: Example



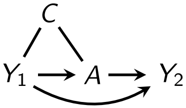
CPDAG/MPDAG



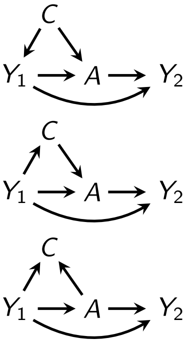
MEC of DAGs

$p((Y_1, Y_2)(a))$ identifiable?

Q2). ID algorithm: Example



CPDAG/MPDAG

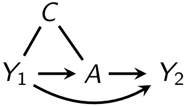


MEC of DAGs

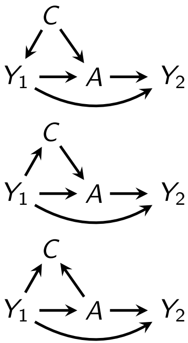
$p((Y_1, Y_2)(a))$ identifiable?

Yes! $A - C - Y_1$, $A \leftarrow Y_1 \rightarrow Y_2$, $A - C - Y_1 \rightarrow Y_2$ not "possibly causal paths" due to $Y_1 \rightarrow A$; $A \rightarrow Y_2$ "possibly causal path" yet not starting with $A - \cdots$

Q2). ID algorithm: Example



CPDAG/MPDAG



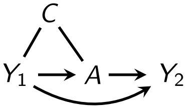
MEC of DAGs

$p((Y_1, Y_2)(a))$ identifiable?

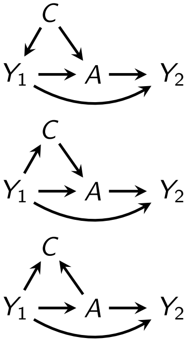
Identification formula: Buckets in $\text{an}_{\mathcal{G}_{V \setminus A}}(Y_1, Y_2) \equiv \{Y_1, Y_2\}$ are $\{Y_1\}$ and $\{Y_2\}$ so

$$p((Y_1, Y_2)(a) = (y_1, y_2)) = p(y_2 | a, y_1) p(y_1)$$

Q2). ID algorithm: Example



CPDAG/MPDAG

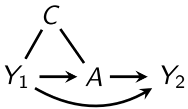


MEC of DAGs

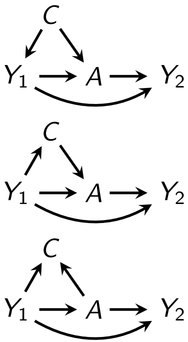
$p((Y_1, Y_2)(a))$ identifiable?

Try g-formula with all the DAGs/SWIGs in the MEC

Q2). ID algorithm: Example



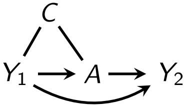
CPDAG/MPDAG



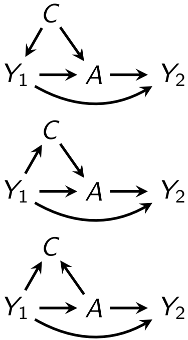
MEC of DAGs

$p((C, Y_1, Y_2)(a))$ identifiable?

Q2). ID algorithm: Example



CPDAG/MPDAG

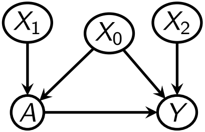


MEC of DAGs

$p((C, Y_1, Y_2)(a))$ identifiable?
No! $A - C$ “possibly causal path” that starts with $A - \cdots$

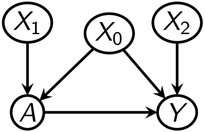
Q3) Efficient adjustment set (research frontier)

- After ID algorithm outputs yes, a causal query is identifiable but could be *over-identified*, e.g.



Q3) Efficient adjustment set (research frontier)

- After ID algorithm outputs yes, a causal query is identifiable but could be *over-identified*, e.g.



If we want to query $\mathbb{E}[Y(1)]$, there are at least four g-formula:

$$\begin{aligned}
\theta_0 &= \mathbb{E}_{X_0}[\mathbb{E}_Y[Y|X_0, A=1]] \\
\theta_{01} &= \mathbb{E}_{X_0,X_1}[\mathbb{E}_Y[Y|X_0, X_1, A=1]] \\
\theta_{02} &= \mathbb{E}_{X_0,X_2}[\mathbb{E}_Y[Y|X_0, X_2, A=1]] \\
\theta_{012} &= \mathbb{E}_{X_0,X_2}[\mathbb{E}_Y[Y|X_0, X_1, X_2, A=1]]
\end{aligned}$$

So $\{X_0\}$, $\{X_0, X_1\}$, $\{X_0, X_2\}$ and $\{X_0, X_1, X_2\}$ are all valid adjustment sets

- Which one would you pick to use?

Q3) Efficient adjustment set

- Henckel et al. 2019 first formalized the above problem for average causal effects $\mathbb{E}[Y(a)]$ under linear causal models

Q3) Efficient adjustment set

- Henckel et al. 2019 first formalized the above problem for average causal effects $\mathbb{E}[Y(a)]$ under linear causal models

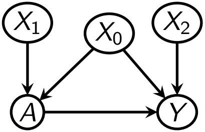
- Rotnitzky and Smucler 2020 demonstrated Henckel et al.'s algorithm also applies to nonlinear models but using very different proof techniques

Q3) Efficient adjustment set

- Henckel et al. 2019 first formalized the above problem for average causal effects $\mathbb{E}[Y(a)]$ under linear causal models

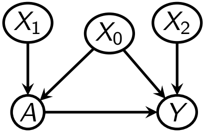
- Rotnitzky and Smucler 2020 demonstrated Henckel et al.'s algorithm also applies to nonlinear models but using very different proof techniques

- Their results have been folklore (with proof) to causal inference researchers for a long time



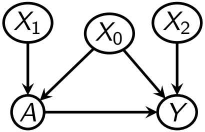
In the above example, sufficient to only adjust for X_0; further adjusting for X_2 can reduce variance but further adjusting for X_1 can inflate variance

Efficient adjustment set



In the above example, sufficient to only adjust for X_0; further adjusting for X_2 can reduce variance but further adjusting for X_1 can inflate variance

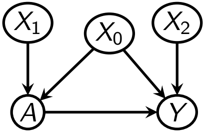
Efficient adjustment set



In the above example, sufficient to only adjust for X_0; further adjusting for X_2 can reduce variance but further adjusting for X_1 can inflate variance

X_0 is obviously a confounder; for the above reason, X_2 is called “precision variable”

Efficient adjustment set



In the above example, sufficient to only adjust for X_0; further adjusting for X_2 can reduce variance but further adjusting for X_1 can inflate variance

X_0 is obviously a confounder; for the above reason, X_2 is called “precision variable”

For future reference, $\{X_0\}$ is called minimal adjustment set because no proper subset of $\{X_0\}$ is still a valid adjustment set

Illustration via linear models

Data generated via linear model

$$\begin{aligned}\mathbb{E}[Y|A,X] &= \beta_{AY}A + \beta_{X_0Y}X_0 + \beta_{X_2Y}X_2 \\ \mathbb{E}[A|X] &= \beta_{X_0A}X_0 + \beta_{X_1A}X_1\end{aligned}$$

Illustration via linear models

Data generated via linear model

$$\begin{aligned}\mathbb{E}[Y|A,X] &= \beta_{AY}A + \beta_{X_0Y}X_0 + \beta_{X_2Y}X_2 \\ \mathbb{E}[A|X] &= \beta_{X_0A}X_0 + \beta_{X_1A}X_1\end{aligned}$$

Then the coefficient β_{AY} of A is the causal effect of A on Y

Illustration via linear models

Data generated via linear model

$$\mathbb{E}[Y|A,X] = \beta_{AY}A + \beta_{X_0Y}X_0 + \beta_{X_2Y}X_2$$
$$\mathbb{E}[A|X] = \beta_{X_0A}X_0 + \beta_{X_1A}X_1$$

Then the coefficient β_{AY} of A is the causal effect of A on Y

From basic linear regression $Y \sim A + X_0 + X_1 + X_2$, least square estimator $\widehat{\tau}_{AY}$ satisfies

$$\sqrt{n}\,(\widehat{\tau}_{AY} - \beta_{AY}) \to N\left(0, \frac{\text{var}(Y - \beta_{AY}A - (X_0\ X_1\ X_2)\tau_{XY})}{\text{var}\,(A - (X_0\ X_1\ X_2)\tau_{XA})}\right)$$

Illustration via linear models

Data generated via linear model

$$
\begin{aligned}
\mathbb{E}[Y|A,X] &= \beta_{AY}A + \beta_{X_0Y}X_0 + \beta_{X_2Y}X_2 \\
\mathbb{E}[A|X] &= \beta_{X_0A}X_0 + \beta_{X_1A}X_1
\end{aligned}
$$

Then the coefficient β_{AY} of A is the causal effect of A on Y

From basic linear regression $Y \sim A + X_0 + X_1 + X_2$, least square estimator $\widehat{\tau}_{AY}$ satisfies

$$
\sqrt{n}\,(\widehat{\tau}_{AY} - \beta_{AY}) \to N\left(0, \frac{\text{var}(Y - \beta_{AY}A - (X_0\ X_1\ X_2)\tau_{XY})}{\text{var}\,(A - (X_0\ X_1\ X_2)\tau_{XA})}\right)
$$

To minimize $\text{var}\left[\sqrt{n}\left(\widehat{\beta}_{AY} - \beta_{AY}\right)\right]$:

Illustration via linear models

Data generated via linear model

$$\mathbb{E}[Y|A,X] = \beta_{AY}A + \beta_{X_0Y}X_0 + \beta_{X_2Y}X_2$$
$$\mathbb{E}[A|X] = \beta_{X_0A}X_0 + \beta_{X_1A}X_1$$

Then the coefficient β_{AY} of A is the causal effect of A on Y

From basic linear regression $Y \sim A + X_0 + X_1 + X_2$, least square estimator $\widehat{\tau}_{AY}$ satisfies

$$\sqrt{n}\left(\widehat{\tau}_{AY} - \beta_{AY}\right) \to N\left(0, \frac{\mathrm{var}(Y - \beta_{AY}A - (X_0 \ X_1 \ X_2)\tau_{XY})}{\mathrm{var}\left(A - (X_0 \ X_1 \ X_2)\tau_{XA}\right)}\right)$$

To minimize var $\left[\sqrt{n}\left(\widehat{\beta}_{AY} - \beta_{AY}\right)\right]$:

1. minimize the numerator: only keep X_0, X_2 because X_2 reduces prediction error

Illustration via linear models

Data generated via linear model

$$\mathbb{E}[Y|A,X] = \beta_{AY}A + \beta_{X_0 Y}X_0 + \beta_{X_2 Y}X_2$$
$$\mathbb{E}[A|X] = \beta_{X_0 A}X_0 + \beta_{X_1 A}X_1$$

Then the coefficient β_{AY} of A is the causal effect of A on Y

From basic linear regression $Y \sim A + X_0 + X_1 + X_2$, least square estimator $\widehat{\tau}_{AY}$ satisfies

$$\sqrt{n}\,(\widehat{\tau}_{AY} - \beta_{AY}) \to N\left(0, \frac{\text{var}(Y - \beta_{AY}A - (X_0\ X_1\ X_2)\tau_{XY})}{\text{var}\,(A - (X_0\ X_1\ X_2)\tau_{XA})}\right)$$

To minimize var $\left[\sqrt{n}\left(\widehat{\beta}_{AY} - \beta_{AY}\right)\right]$:

1. minimize the numerator: only keep X_0, X_2 because X_2 reduces prediction error
2. maximize the denominator: get rid of X_1 to inflate prediction error

Improving efficiency (reducing variance) from any valid adjustment set

Lemma 1 (Rotnitzky and Smucler, JMLR 2020)

Denote θ_E as the g formula adjusting for the set E. Let B be any valid adjustment set. If there exists a set C s.t. $A \perp\!\!\!\perp C | B$, then $B \cup C$ is also a valid adjustment set, and $\theta_{B \cup C}$ improves over θ_B in the following sense:

- the variance $\sigma^2_{B \cup C}$ of any "optimal estimator" of $\theta_{B \cup C}$ is no greater than that σ^2_B of θ_B

$$\sigma^2_{B \cup C} \leq \sigma^2_B$$

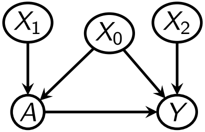
Improving efficiency (reducing variance) from any valid adjustment set

Lemma 1 (Rotnitzky and Smucler, JMLR 2020)

Denote θ_E as the g formula adjusting for the set E. Let B be any valid adjustment set. If there exists a set C s.t. $A \perp\!\!\!\perp C | B$, then $B \cup C$ is also a valid adjustment set, and $\theta_{B \cup C}$ improves over θ_B in the following sense:

- the variance $\sigma^2_{B \cup C}$ of any "optimal estimator" of $\theta_{B \cup C}$ is no greater than that σ^2_B of θ_B

$$\sigma^2_{B \cup C} \leq \sigma^2_B$$



In the above example, $\{X_0\}$ is a valid adjustment set, and $A \perp\!\!\!\perp X_2 | X_0$, so adjusting for $\{X_0, X_2\}$ is better

Comments

- "Optimal estimator" is in fact "semiparametric efficient estimator", which will not be covered in this course due to time constraint

Comments

- "Optimal estimator" is in fact "semiparametric efficient estimator", which will not be covered in this course due to time constraint

- How to prove? I will use the same example as an illustration: for any valid adjustment set B

$$\begin{aligned}
\sigma_B^2 &= \mathbb{E}\left[\left\{\frac{A}{\Pr(A=1|B)}(Y-\mathbb{E}[Y|A=1,B])+\mathbb{E}[Y|A=1,B]-\theta_B\right\}^2\right] \\
&= \mathbb{E}\left[\frac{A}{\Pr(A=1|B)^2}(Y-\mathbb{E}[Y|A=1,B])^2\right] \\
&\quad + \mathbb{E}_B\left[\{\mathbb{E}[Y|A=1,B]-\theta_B\}^2\right] \\
&= \mathbb{E}\left[\frac{1}{\Pr(A=1|B)}\mathbb{E}\left[(Y-\mathbb{E}[Y|A=1,B])^2|A=1,B\right]\right] \\
&\quad + \mathbb{E}_B\left[\{\mathbb{E}[Y|A=1,B]-\theta_B\}^2\right]
\end{aligned}$$

Comments (continue)

$$
\begin{aligned}
&\sigma_B^2 \\
&= \mathbb{E}\left[\left\{\frac{1}{\Pr(A=1|B)}-1\right\}\mathbb{E}\left[(Y-\mathbb{E}[Y|A=1,B])^2|A=1,B\right]\right] \\
&\quad + \mathbb{E}_B\left[\mathbb{E}\left[(Y-\mathbb{E}[Y|A=1,B])^2|A=1,B\right]\right] + \mathbb{E}_B\left[\left\{\mathbb{E}[Y|A=1,B]-\theta_B\right\}^2\right] \\
&= \mathbb{E}\left[\left\{\frac{1}{\Pr(A=1|B)}-1\right\}\mathrm{var}\left[Y|A=1,B\right]\right] + \mathbb{E}_B[\mathbb{E}[(Y-\theta_B)^2|A=1,B]]
\end{aligned}
$$

Comments (continue)

$$
\begin{aligned}
&\sigma_B^2 \\
&= \mathbb{E}\left[\left\{\frac{1}{\Pr(A=1|B)} - 1\right\}\mathbb{E}\left[(Y - \mathbb{E}[Y|A=1,B])^2|A=1,B\right]\right] \\
&\quad + \mathbb{E}_B\left[\mathbb{E}\left[(Y - \mathbb{E}[Y|A=1,B])^2|A=1,B\right]\right] + \mathbb{E}_B\left[\{\mathbb{E}[Y|A=1,B] - \theta_B\}^2\right] \\
&= \mathbb{E}\left[\left\{\frac{1}{\Pr(A=1|B)} - 1\right\}\mathrm{var}\left[Y|A=1,B\right]\right] + \mathbb{E}_B[\mathbb{E}[(Y-\theta_B)^2|A=1,B]]
\end{aligned}
$$

Similarly

$$
\begin{aligned}
\sigma_{B\cup C}^2 &= \mathbb{E}\left[\left\{\frac{1}{\Pr(A=1|B,C)} - 1\right\}\mathrm{var}\left[Y|A=1,B,C\right]\right] \\
&\quad + \mathbb{E}_{B,C}[\mathbb{E}[(Y-\theta_{B\cup C})^2|A=1,B,C]] \\
\text{(why?)} &= \mathbb{E}\left[\left\{\frac{1}{\Pr(A=1|B)} - 1\right\}\mathbb{E}\left[\mathrm{var}\left[Y|A=1,B,C\right]|A=1,B\right]\right] \\
&\quad + \mathbb{E}_B[\mathbb{E}[(Y-\theta_B)^2|A=1,B]]
\end{aligned}
$$

Comments (continue)

Take the difference:

$$
\begin{aligned}
&\sigma_B^2 - \sigma_{B\cup C}^2 \\
&= \mathbb{E}\Bigg[\left\{\frac{1}{\Pr(A=1|B)} - 1\right\}\left\{\mathsf{var}\left[Y|A=1,B\right] - \mathbb{E}\left[\mathsf{var}\left[Y|A=1,B,C\right]|A=1,B\right] \right. \\
&= \mathbb{E}\left[\left\{\frac{1}{\Pr(A=1|B)} - 1\right\}\mathsf{var}\left\{\mathbb{E}\left[Y|A=1,B,C\right]|A=1,B\right\}\right] \geq 0
\end{aligned}
$$

Q.E.D.

Improving efficiency (reducing variance) from any valid adjustment set

Lemma 2 (Rotnitzky and Smucler, JMLR 2020)

Denote θ_E as the g formula adjusting for the set E. Let $B \cup C$ be any valid adjustment set. If $Y \perp\!\!\!\perp C|B, A$, then B is also a valid adjustment set, and θ_B improves over $\theta_{B \cup C}$ in the following sense:

- the variance σ^2_B of any "optimal estimator" of θ_B is no greater than that $\sigma^2_{B \cup C}$ of $\theta_{B \cup C}$

$$\sigma^2_B \leq \sigma^2_{B \cup C}$$

So combining Lemma 1 and Lemma 2, one can get the benefits of both

Finale of efficient adjustment set

Theorem 3 (Henckel et al. 2019, Rotnitzky and Smucler JMLR 2020)

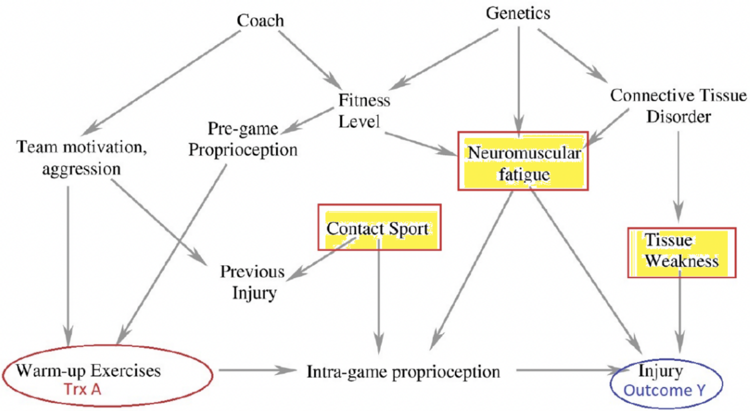
Define

$$O = \left\{ \begin{array}{c} \text{non-descendants of } A \text{ that are also parents of } Y \\ \text{or parents of vertices on the causal path between } A \text{ and } Y \end{array} \right\}$$

Then O is a globally optimal valid adjustment set.

This theorem formalized statisticians' long-standing intuition in the most general way under causal sufficiency

Example



An example stolen from Andrea Rotnitzky's ocis talk

Open problem

- Optimal g-formula for general case in DAG (longitudinal and dynamic regime have been shown to be impossible by Rotnitzky and Smucler, change definition?)

Open problem

- Optimal g-formula for general case in DAG (longitudinal and dynamic regime have been shown to be impossible by Rotnitzky and Smucler, change definition?)

- Optimal adjustment formula for CPDAG/MPDAG in nonlinear case (linear case done in Guo and Perković, JMLR 2022)

Open problem

- Optimal g-formula for general case in DAG (longitudinal and dynamic regime have been shown to be impossible by Rotnitzky and Smucler, change definition?)

- Optimal adjustment formula for CPDAG/MPDAG in nonlinear case (linear case done in Guo and Perković, JMLR 2022)

- Optimal g-formula for CPDAG/MPDAG

Causal graph allowing latent variables

- So far we have considered the whole pipeline: observational data + experiments/background knowledge → MPDAG → identification

Causal graph allowing latent variables

- So far we have considered the whole pipeline: observational data $+$ experiments/background knowledge $\rightarrow$ MPDAG $\rightarrow$ identification

- What is left is to allow for latent variables, which is a much more challenging problem

Causal graph allowing latent variables

- So far we have considered the whole pipeline: observational data + experiments/background knowledge → MPDAG → identification

- What is left is to allow for latent variables, which is a much more challenging problem

- Directly adding all latent variables into DAG is inconvenient for obvious reasons

Causal graph allowing latent variables

- So far we have considered the whole pipeline: observational data + experiments/background knowledge → MPDAG → identification

- What is left is to allow for latent variables, which is a much more challenging problem

- Directly adding all latent variables into DAG is inconvenient for obvious reasons

- We need new graphical models – Acyclic Directed Mixed Graphs (ADMGs) and mDAGs

Causal graph allowing latent variables

- So far we have considered the whole pipeline: observational data + experiments/background knowledge → MPDAG → identification

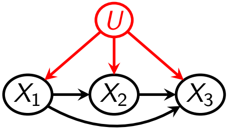
- What is left is to allow for latent variables, which is a much more challenging problem

- Directly adding all latent variables into DAG is inconvenient for obvious reasons

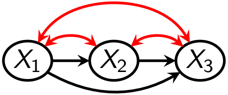
- We need new graphical models – Acyclic Directed Mixed Graphs (ADMGs) and mDAGs

- We will see several examples of ADMGs and mDAGs, but our focus will be on ADMGs

Examples of ADMGs



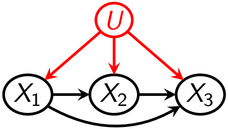
DAG with latent variable



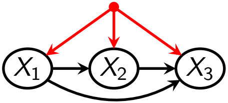
Corresponding ADMG

Introducing bidirected edges, but losing information that all three observables share the same latent variable(s) U

Examples of mDAGs



DAG with latent variable



Corresponding mDAG

Introducing hyperedges (the red trident structure in the right graph), increasing the representation complexity (may eventually need a hyperedge with many many endpoints), but keeping more information of the original DAG

Latent projection: Map DAG with latent to ADMG

- Before talking about ADMG, we first define a graphical operation called latent projection

Latent projection: Map DAG with latent to ADMG

- Before talking about ADMG, we first define a graphical operation called latent projection
- A DAG $\mathcal{G}$ with vertices $O \cup L$ where L are latent

Latent projection: Map DAG with latent to ADMG

- Before talking about ADMG, we first define a graphical operation called latent projection

- A DAG $\mathcal{G}$ with vertices $O \cup L$ where L are latent

- Denote latent projection of $\mathcal{G}$ as $\mathcal{G}[O]$, constructed as follows

Latent projection: Map DAG with latent to ADMG

- Before talking about ADMG, we first define a graphical operation called latent projection
- A DAG $\mathcal{G}$ with vertices $O \cup L$ where L are latent
- Denote latent projection of $\mathcal{G}$ as $\mathcal{G}[O]$, constructed as follows
    - Keep vertices in O and edges between every pair of vertices in O
    - If $X, Y \in O$, $\circ \in L$, and there is a causal path $X \to \circ \!\to \cdots \to\! \circ \to Y$, then add $X \to Y$ if it is not already there

Latent projection: Map DAG with latent to ADMG

- Before talking about ADMG, we first define a graphical operation called latent projection
- A DAG $\mathcal{G}$ with vertices $O \cup L$ where L are latent
- Denote latent projection of $\mathcal{G}$ as $\mathcal{G}[O]$, constructed as follows
  - Keep vertices in O and edges between every pair of vertices in O
  - If $X, Y \in O$, $\circ \in L$, and there is a causal path $X \rightarrow \circ \rightarrow \cdots \rightarrow \circ \rightarrow Y$, then add $X \rightarrow Y$ if it is not already there
  - If there exists a path between X and Y such that the non-endpoints are non-colliders in L, and such that the edge adjacent to the end points are both pointing to the end points, then add $X \leftrightarrow Y$

ADMG

- Definition: A graph $\mathcal{G}^\dagger$ is an ADMG if $\mathcal{G}^\dagger = \mathcal{G}[V]$ for some DAG $\mathcal{G}$

ADMG

- Definition: A graph $\mathcal{G}^\dagger$ is an ADMG if $\mathcal{G}^\dagger = \mathcal{G}[V]$ for some DAG $\mathcal{G}$

- Equivalent definition: A graph $\mathcal{G}^\dagger$ is an ADMG if
  (i) the edges are either $\rightarrow$ or $\leftrightarrow$
  (ii) there are no directed cycles

ADMG

- Definition: A graph $\mathcal{G}^\dagger$ is an ADMG if $\mathcal{G}^\dagger = \mathcal{G}[V]$ for some DAG $\mathcal{G}$

- Equivalent definition: A graph $\mathcal{G}^\dagger$ is an ADMG if
  1. (i) the edges are either $\to$ or $\leftrightarrow$
  2. (ii) there are no directed cycles

- Exercise: try to prove the above two definitions are equivalent

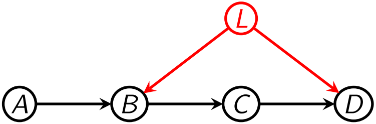
ADMG

- Definition: A graph $\mathcal{G}^\dagger$ is an ADMG if $\mathcal{G}^\dagger = \mathcal{G}[V]$ for some DAG $\mathcal{G}$

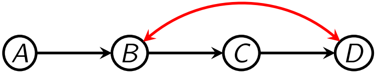
- Equivalent definition: A graph $\mathcal{G}^\dagger$ is an ADMG if
  (i) the edges are either $\to$ or $\leftrightarrow$
  (ii) there are no directed cycles

- Exercise: try to prove the above two definitions are equivalent

- Example: Verma

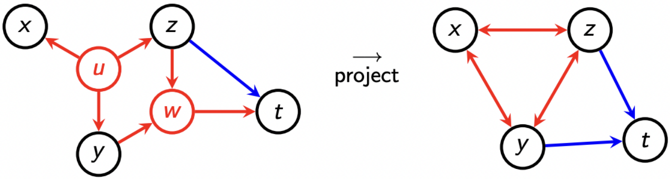


DAG $\mathcal{G}$



ADMG $\mathcal{G}[\{A, B, C, D\}]$

Example



Latent projection leads to an **acyclic directed mixed graph** (ADMG)

Can read off independences with d/m-separation.

The projection preserves the causal structure; Verma and Pearl (1992).

MEC of ADMGs?

- Obviously, to even consider causal discovery, one needs concepts like MEC of DAGs

MEC of ADMGs?

- Obviously, to even consider causal discovery, one needs concepts like MEC of DAGs
- Unfortunately, MEC of ADMGs has been an open problem for 30 years [Shpitser et al. Introduction to nested Markov models conjectured MEC for ADMGs with four vertices via computer-assisted proof]
- Robin Evans (Oxford statistics) is getting ϵ-close to prove MEC of ADMGs (personal communication)

MEC of ADMGs?

- Obviously, to even consider causal discovery, one needs concepts like MEC of DAGs

- Unfortunately, MEC of ADMGs has been an open problem for 30 years [Shpitser et al. Introduction to nested Markov models conjectured MEC for ADMGs with four vertices via computer-assisted proof]

- Robin Evans (Oxford statistics) is getting ϵ-close to prove MEC of ADMGs (personal communication)

- Richardson and Spirtes 2002 have long figured out MEC of a super-model of ADMGs, maximal ancestral graphs (MAGs)

MEC of ADMGs?

- Obviously, to even consider causal discovery, one needs concepts like MEC of DAGs

- Unfortunately, MEC of ADMGs has been an open problem for 30 years [Shpitser et al. Introduction to nested Markov models conjectured MEC for ADMGs with four vertices via computer-assisted proof]

- Robin Evans (Oxford statistics) is getting ϵ-close to prove MEC of ADMGs (personal communication)

- Richardson and Spirtes 2002 have long figured out MEC of a super-model of ADMGs, maximal ancestral graphs (MAGs)

- Spirtes et al. ["Computation, causation, and discovery" Chapter 6] developed FCI (fast causal inference) algorithm which is sound and complete (after modified by Jiji Zhang) for recovering MAGs

Computer-assisted conjecture of Shpitser for four-vertex ADMG

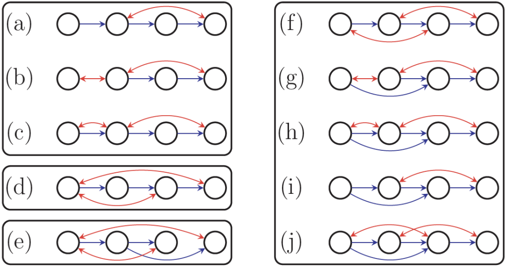
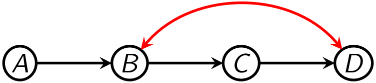


Figure 10: The conjectured equivalence classes among graph patterns (with vertex labeling suppressed) of 4 node ADMGs corresponding to nested Markov models that are strict submodels of the ordinary Markov models given by the same ADMG.

MAG definition

- Ancestral graphs (AGs) do not allow for *almost directed cycles*: the graph below is not an AG because $B \to C \to D$ and $B \leftrightarrow D$ form an *almost directed cycle*

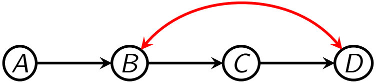


ADMG $\mathcal{G}[\{A, B, C, D\}]$ but not a MAG

[1] Every non-endpoint on the path is a collider and every collider is an ancestor of an endpoint of the path

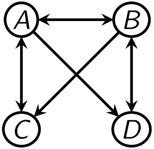
MAG definition

- Ancestral graphs (AGs) do not allow for *almost directed cycles*: the graph below is not an AG because $B \to C \to D$ and $B \leftrightarrow D$ form an *almost directed cycle*

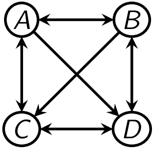


ADMG $\mathcal{G}[\{A, B, C, D\}]$ but not a MAG

- Maximal AGs do not allow for *inducing paths*[1] between any two non-adjacent vertices



Ancestral but not maximal because $\{C - A - B - D\}$ is an inducing path



MAG

[1] Every non-endpoint on the path is a collider and every collider is an ancestor of an endpoint of the path

Intuition on almost directed cycles and inducing paths

- Why not almost directed cycles?

Intuition on almost directed cycles and inducing paths

- Why not almost directed cycles?

To preserve ancestral relationships among the observables

Intuition on almost directed cycles and inducing paths

- Why not almost directed cycles?

To preserve ancestral relationships among the observables

- Why not inducing paths between non-adjacent vertices?

Intuition on almost directed cycles and inducing paths

- Why not almost directed cycles?

To preserve ancestral relationships among the observables

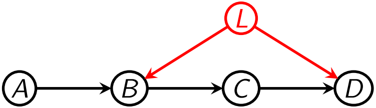
- Why not inducing paths between non-adjacent vertices?

On a DAG, every two non-adjacent vertices have to be d-separated by some other vertices; MAG needs to preserve such a property because no observed data can distinguish whether or not the edge between C and D exists

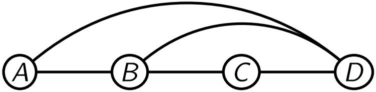
From DAG to MAG

Two step procedure: input a DAG $\mathcal{G}$ output its MAG $\mathcal{M}$

(1) Two observables A and B in $\mathcal{G}$ are adjacent in $\mathcal{M}$ if and only if there is an inducing path between A and B relative to the hidden vertices (i.e. hidden vertices on the path can be non-colliders)



DAG with hidden L



MAG from $\mathcal{G}$

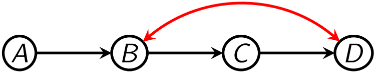
- every direct edge between two observables is an inducing path relative to L
- $A \rightarrow B \leftarrow L \rightarrow D$ inducing path rel. to L: B is a collider and $B \in \text{an}(D)$
- $B \rightarrow L \leftarrow D$ inducing path rel. to L

From DAG to MAG

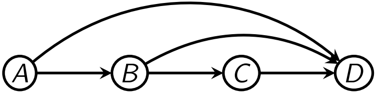
Two step procedure: input a DAG $\mathcal{G}$ output its MAG $\mathcal{M}$

(1) Two observables A and B in $\mathcal{G}$ are adjacent in $\mathcal{M}$ if and only if there is an inducing path between A and B relative to the hidden vertices (i.e. hidden vertices on the path can be non-colliders)

(2) Orient $A \rightarrow B$ if $A = \text{an}(B)$, $A \leftarrow B$ if $B = \text{an}(A)$, and $A \leftrightarrow B$ if otherwise



ADMG $\mathcal{G}[\{A, B, C, D\}]$ but not a MAG



MAG from $\mathcal{G}$

MAGs and m-separation

- All the d-separation criterion for DAGs can be carried over to MAGs and

MAGs and m-separation

- All the d-separation criterion for DAGs can be carried over to MAGs and
- they are called m-separation

MEC of MAGs

- High level: the set of MAGs that observational data cannot tell apart from the true MAG based on independence and conditional independence constraints

MEC of MAGs

- High level: the set of MAGs that observational data cannot tell apart from the true MAG based on independence and conditional independence constraints

- Operational: MAGs that share the same m-separations

MEC of MAGs

- High level: the set of MAGs that observational data cannot tell apart from the true MAG based on independence and conditional independence constraints

- Operational: MAGs that share the same m-separations

- Constructive: Spirtes, Richardson. AAAI 1996 and Zhang, 2012 proved two MAGs are Markov equivalent if and only if (1) they share the same skeleton, (2) they share the same v-structures, and (3) they share the same "discriminating path" for the same vertex that is a (non)-collider on both MAGs

MEC of MAGs

- High level: the set of MAGs that observational data cannot tell apart from the true MAG based on independence and conditional independence constraints

- Operational: MAGs that share the same m-separations

- Constructive: Spirtes, Richardson. AAAI 1996 and Zhang, 2012 proved two MAGs are Markov equivalent if and only if (1) they share the same skeleton, (2) they share the same v-structures, and (3) they share the same "discriminating path" for the same vertex that is a (non)-collider on both MAGs

- MEC of MAGs can be represented by PAGs (partial ancestral graphs), analogous to CPDAGs

Supplement: Discriminating path

In a MAG, a path p between A and B, e.g. $p = (A, \cdots, W, V, B)$ is a discriminating path for V if

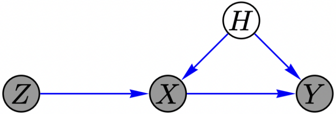
1. p includes at least three edges (and of course, at least four vertices)
2. V is adjacent to one endpoint on p and in the above case, B
3. A and B non-adjacent, and every vertex between A and V is a collider on p and is a parent of B

The reason we have to consider discriminating path for MAG is that when p is a discriminating path, it behaves as a v-structure between A and B in terms of the triple (W, V, B) in the following sense:

1. (W, V, B) is a non-collider if and only if every set m-separating A and B contains V
2. (W, V, B) is a collider with $W \to B$ or $W \leftarrow B$ if and only if every set m-separating A and B does not contain V

Difference (I)

There can be observed variables which are not adjacent, but for which no subset of the other observed variables d-separates them.



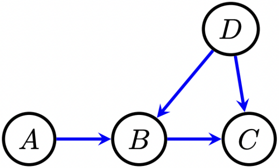
Here H is unobserved.

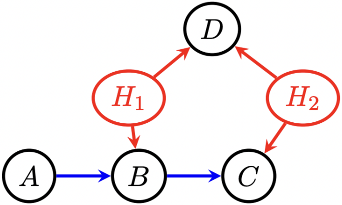
Z and Y are d-connected given $\emptyset$.

Z and Y are d-connected given $\{X\}$.

Difference (II)

For DAGs without hidden variables, *same adjacencies* and *same unshielded colliders* were necessary and sufficient for equivalence ⇒ only need to look at structures involving at most 3 vertices.





These graphs are not Markov equivalent over the observed margin.

A is d-separated from C given $\{B, D\}$ in the left graph.
A is d-separated from C given $\{B\}$ in the right graph.

⇒ Need to look at more complex structures.

Q1) Causal discovery with latent variables: FCI algorithm

- For more recent development, check out Bhattacharya, Nagarajan, Malinsky, Shpitser AISTATS 2021.

Q1) Causal discovery with latent variables: FCI algorithm

- For more recent development, check out Bhattacharya, Nagarajan, Malinsky, Shpitser AISTATS 2021.

- Will not describe FCI in detail but note the following

Q1) Causal discovery with latent variables: FCI algorithm

- For more recent development, check out Bhattacharya, Nagarajan, Malinsky, Shpitser AISTATS 2021.

- Will not describe FCI in detail but note the following

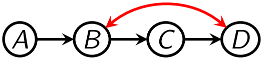
- Spirtes et al. also proved as sample size $n \to \infty$, the output of FCI converges to the MEC of the true underlying MAG, under faithfulness

What do we lose by only considering MAGs instead of ADMGs

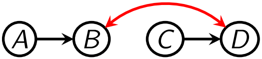
- At a high level, MAG preserves and only preserves (1) all (conditional) independence constraints and (2) ancestral relationships on observables implied by the original DAG

What do we lose by only considering MAGs instead of ADMGs

- At a high level, MAG preserves and only preserves (1) all (conditional) independence constraints and (2) ancestral relationships on observables implied by the original DAG
- But in reality, DAGs with hidden variables imply other constraints on the observables other than (conditional) independences



$\overset{\text{reweight by } \frac{\widetilde{f}(c)}{f(c|b)}}{\Rightarrow}$

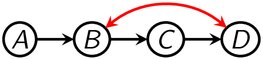


Dormant independence: $A \perp\!\!\!\perp D$

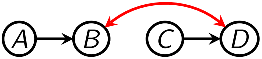
What do we lose by only considering MAGs instead of ADMGs

- At a high level, MAG preserves and only preserves (1) all (conditional) independence constraints and (2) ancestral relationships on observables implied by the original DAG

- But in reality, DAGs with hidden variables imply other constraints on the observables other than (conditional) independences



reweight by $\frac{\widetilde{f}(c)}{f(c|b)}$ $\Rightarrow$



Dormant independence: $A \perp\!\!\!\perp D$

- MAGs fail to preserve such dormant independences, or nested Markov properties

Controversies over causal discovery

- As promising as all the causal discovery algorithms may sound, there are two fundamental issues

Controversies over causal discovery

- As promising as all the causal discovery algorithms may sound, there are two fundamental issues
  i. As many computer science problems, almost all papers on causal discovery assume the oracle query model, hence assuming away all statistical errors. But conditional independence is impossible to test without restrictive modeling assumptions

Controversies over causal discovery

- As promising as all the causal discovery algorithms may sound, there are two fundamental issues
  1. As many computer science problems, almost all papers on causal discovery assume the oracle query model, hence assuming away all statistical errors. But conditional independence is impossible to test without restrictive modeling assumptions
  2. Faithfulness assumption is problematic IF we only have finite sample

Controversies over causal discovery

- As promising as all the causal discovery algorithms may sound, there are two fundamental issues
  i. As many computer science problems, almost all papers on causal discovery assume the oracle query model, hence assuming away all statistical errors. But conditional independence is impossible to test without restrictive modeling assumptions
  ii. Faithfulness assumption is problematic IF we only have finite sample

- As a result of highly fruitful debate between statisticians (James Robins and Larry Wasserman) and CMU causal philosophers (Spirtes et al.), science of causal discovery progresses in light speed in late 1990's and early 2000's (see Chapters 8-11 of "Computation, causation and discovery")

Controversies over causal discovery

- As promising as all the causal discovery algorithms may sound, there are two fundamental issues
    1. As many computer science problems, almost all papers on causal discovery assume the oracle query model, hence assuming away all statistical errors. But conditional independence is impossible to test without restrictive modeling assumptions
    2. Faithfulness assumption is problematic IF we only have finite sample

- As a result of highly fruitful debate between statisticians (James Robins and Larry Wasserman) and CMU causal philosophers (Spirtes et al.), science of causal discovery progresses in light speed in late 1990's and early 2000's (see Chapters 8-11 of "Computation, causation and discovery")

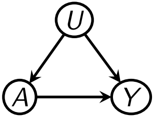
- It culminated at the paper by Caroline Uhler et al. in 2014, who ingeniously applies classical results from algebraic geometry to this problem

What is faithfulness assumption

- We talked about faithfulness assumption but have not really discussed it enough

What is faithfulness assumption

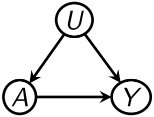
- We talked about faithfulness assumption but have not really discussed it enough
- Recall definition of faithfulness:



Faithfulness: $A \perp\!\!\!\perp Y \Rightarrow A$ and Y are d-separated

What is faithfulness assumption

- We talked about faithfulness assumption but have not really discussed it enough
- Recall definition of faithfulness:

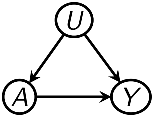


Faithfulness: $A \perp\!\!\!\perp Y \Rightarrow A$ and Y are d-separated

$\Leftrightarrow A$ and Y are not d-separated $\Rightarrow A \not\!\perp\!\!\!\perp Y$

What is faithfulness assumption

- We talked about faithfulness assumption but have not really discussed it enough
- Recall definition of faithfulness:



Faithfulness: $A \perp\!\!\!\perp Y \Rightarrow A$ and Y are d-separated

$\Leftrightarrow A$ and Y are not d-separated $\Rightarrow A \not\perp\!\!\!\perp Y$

- Intuitively speaking, in the above DAG, faithfulness rules out the possibility that $A \leftarrow U \rightarrow Y$ effect somehow cancel out the effect $A \rightarrow Y$ when looking at the marginal dependence between A and Y!

Is it reasonable to assume faithfulness?

The answer is quite mixed.

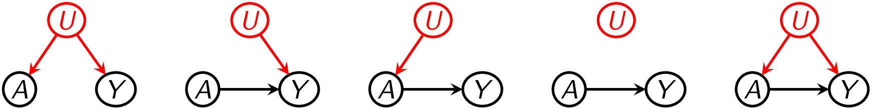
On one hand, for probability distributions parameterized by finite-dimensional parameters, the set of distributions Markov to a DAG but unfaithful to that DAG has Lebesgue measure 0 (Meek. UAI 1995)!

On the other hand... let's see towards the end of this section

Is consistency enough?

We will consider the following example: observe (A, Y) and U could be unmeasured but we assume the background knowledge U precedes A and Y and A precedes Y. So we have the following 8 potential DAGs:

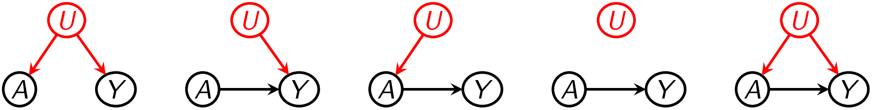
Subset 1: $A \not\perp\!\!\!\perp Y$, FCI reports “don’t know”



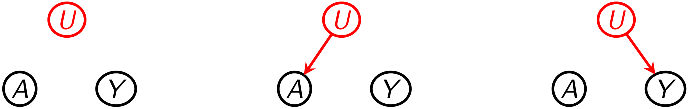
Is consistency enough?

We will consider the following example: observe (A, Y) and U could be unmeasured but we assume the background knowledge U precedes A and Y and A precedes Y. So we have the following 8 potential DAGs:

Subset 1: $A \not\perp\!\!\!\perp Y$, FCI reports "don't know"



Subset 2: $A \perp\!\!\!\perp Y$, FCI reports "A does not cause Y"



If U is observed but U is continuous

- When U is observed, one can do one independence test between A and Y and one conditional independence test between A and Y given U

If U is observed but U is continuous

- When U is observed, one can do one independence test between A and Y and one conditional independence test between A and Y given U

- If U is continuous, then there may exist consistent tests but no uniformly consistent test of $H_0 : Y \perp\!\!\!\perp A | U$ (proved in Shah, Peters. Annals of Statistics 2020.)

If U is observed but U is continuous

- When U is observed, one can do one independence test between A and Y and one conditional independence test between A and Y given U

- If U is continuous, then there may exist consistent tests but no uniformly consistent test of $H_0 : Y \perp\!\!\!\perp A|U$ (proved in Shah, Peters. Annals of Statistics 2020.)

- ϵ-δ translation of consistency:
  For the true but unknown distribution $\mathcal{P}$, given an error tolerance $\epsilon > 0$, we can find a large integer $N(\epsilon, \mathcal{P}) \equiv N > 0$, such that for every $n > N$, the sum of type-I and type-II errors of testing H_0 is below ϵ

If U is observed but U is continuous

- When U is observed, one can do one independence test between A and Y and one conditional independence test between A and Y given U

- If U is continuous, then there may exist consistent tests but no uniformly consistent test of $H_0 : Y \perp\!\!\!\perp A | U$ (proved in Shah, Peters. Annals of Statistics 2020.)

- ϵ-δ translation of consistency:
  For the true but unknown distribution $\mathcal{P}$, given an error tolerance $\epsilon > 0$, we can find a large integer $N(\epsilon, \mathcal{P}) \equiv N > 0$, such that for every $n > N$, the sum of type-I and type-II errors of testing H_0 is below ϵ

- ϵ-δ translation of uniform consistency:
  For every distribution, given an error tolerance $\epsilon > 0$, we can find a large integer $N(\epsilon) \equiv N > 0$, such that for every $n > N$, the sum of type-I and (non-local) type-II errors of testing H_0 is below ϵ

Why uniform consistency is what we need?

Consistency tells us, for the given datasets, there exists a threshold N, that depends on the unknown distribution of the given datasets and error tolerance ϵ, such that whenever the actual sample size $n > N$, we can guarantee the statistical error is below ϵ

Why uniform consistency is what we need?

Consistency tells us, for the given datasets, there exists a threshold N, that depends on the unknown distribution of the given datasets and error tolerance ϵ, such that whenever the actual sample size $n > N$, we can guarantee the statistical error is below ϵ

Consistency cannot guide us in terms of study design because the required minimum sample size depends on the unknown distribution, which is the target of our statistical analysis

Why uniform consistency is what we need?

Consistency tells us, for the given datasets, there exists a threshold N, that depends on the unknown distribution of the given datasets and error tolerance ϵ, such that whenever the actual sample size $n > N$, we can guarantee the statistical error is below ϵ

Consistency cannot guide us in terms of study design because the required minimum sample size depends on the unknown distribution, which is the target of our statistical analysis

Uniform consistency, however, can tell us for all possible distributions that we are considering, there exists a universal N such that $n > N$, the worst-case statistical error is guaranteed to be below ϵ

Why uniform consistency is what we need?

Consistency tells us, for the given datasets, there exists a threshold N, that depends on the unknown distribution of the given datasets and error tolerance ϵ, such that whenever the actual sample size $n > N$, we can guarantee the statistical error is below ϵ

Consistency cannot guide us in terms of study design because the required minimum sample size depends on the unknown distribution, which is the target of our statistical analysis

Uniform consistency, however, can tell us for all possible distributions that we are considering, there exists a universal N such that $n > N$, the worst-case statistical error is guaranteed to be below ϵ

Therefore, a meaningful statistical theory should be uniform rather than point-wise (one can also connects such disparity to Hodges' estimator in classical statistical theory)

If U is observed and U is categorical

- When U is observed, one can do one independence test between A and Y and one conditional independence test between A and Y given U

- If U is categorical, then testing $H_0 : Y \perp\!\!\!\perp A | U$ is equivalent to testing finitely many marginal independences, for which there may exist uniformly consistent tests (e.g. dcorr, BETs, ...)

If U is observed and U is categorical

- When U is observed, one can do one independence test between A and Y and one conditional independence test between A and Y given U

- If U is categorical, then testing $H_0 : Y \perp\!\!\!\perp A | U$ is equivalent to testing finitely many marginal independences, for which there may exist uniformly consistent tests (e.g. dcorr, BETs, ...)

- Since categorical U seems easy, let's assume that throughout

If U is observed and U is categorical

- When U is observed, one can do one independence test between A and Y and one conditional independence test between A and Y given U

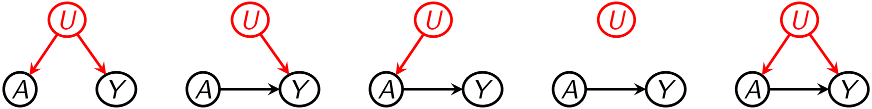
- If U is categorical, then testing $H_0 : Y \perp\!\!\!\perp A|U$ is equivalent to testing finitely many marginal independences, for which there may exist uniformly consistent tests (e.g. dcorr, BETs, ...)

- Since categorical U seems easy, let's assume that throughout

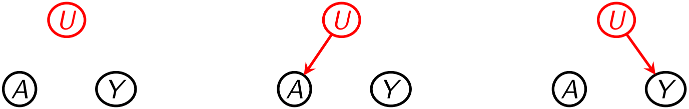
- If the level of U is large compared to sample size (e.g. high-dimensional multinomial), then it again becomes hard

If U is latent but no faithfulness for U is assumed

Subset 1: $A \not\perp\!\!\!\perp Y$, FCI reports "don't know"

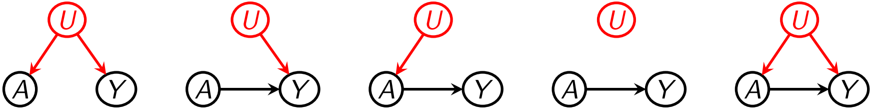


Subset 2: $A \perp\!\!\!\perp Y$, FCI reports "A does not cause Y"

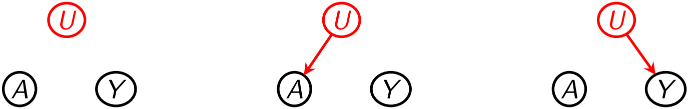


If U is latent but no faithfulness for U is assumed

Subset 1: $A \not\!\perp\!\!\!\perp Y$, FCI reports “don’t know”



Subset 2: $A \perp\!\!\!\perp Y$, FCI reports “A does not cause Y”



Without faithfulness, even no consistent method because $A \perp\!\!\!\perp Y$ can be compatible with the last DAG in Subset 1, so
H_0 : no causal effect from A to Y will be erroneously accepted

If U is latent and faithfulness is assumed

With faithfulness, we rule out the last DAG in Subset 1, so H_0 : no causal effect from A to Y can be safely rejected or accepted...

If U is latent and faithfulness is assumed

With faithfulness, we rule out the last DAG in Subset 1, so
H_0 : no causal effect from A to Y can be safely rejected or accepted...

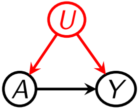
except that we can't!

If U is latent and faithfulness is assumed

With faithfulness, we rule out the last DAG in Subset 1, so H_0 : no causal effect from A to Y can be safely rejected or accepted...

except that we can't!

Robins et al.'s counterexample does require some deep statistical thinking, but intuitively speaking they simply constructed the following:



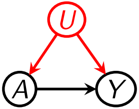
A distribution $\mathcal{P}_n$

If U is latent and faithfulness is assumed

With faithfulness, we rule out the last DAG in Subset 1, so
H_0 : no causal effect from A to Y can be safely rejected or accepted...

except that we can't!

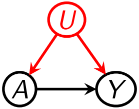
Robins et al.'s counterexample does require some deep statistical thinking, but intuitively speaking they simply constructed the following:



A distribution $\mathcal{P}_n$

1) Markov to the above DAG with a very strong $A \rightarrow Y$ effect
2) The existence of U induces a non-zero marginal dependence between A and Y
3) Yet such dependence depends on n and converges to 0 at rate $n^{-1/2}$

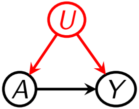
Robins et al.'s counterexample



A distribution $\mathcal{P}_n$

1) Markov to the above DAG with a very strong $A \to Y$ effect
2) The existence of U induces a non-zero marginal dependence between A and Y
3) Yet such dependence depends on n and converges to 0 at rate $O(n^{-1/2})$

Robins et al.'s counterexample

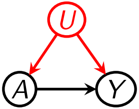


A distribution $\mathcal{P}_n$

1) Markov to the above DAG with a very strong $A \to Y$ effect
2) The existence of U induces a non-zero marginal dependence between A and Y
3) Yet such dependence depends on n and converges to 0 at rate $O(n^{-1/2})$

Then a valid independence test between A and Y will fail to reject $H_0 : A \perp\!\!\!\perp Y$ even as $n \to \infty$

Robins et al.'s counterexample



A distribution $\mathcal{P}_n$

1) Markov to the above DAG with a very strong $A \rightarrow Y$ effect
2) The existence of U induces a non-zero marginal dependence between A and Y
3) Yet such dependence depends on n and converges to 0 at rate $O(n^{-1/2})$

Then a valid independence test between A and Y will fail to reject $H_0 : A \perp\!\!\!\perp Y$ even as $n \rightarrow \infty$

Because a valid independence test must not reject H_0 with high probability when H_0 is indeed correct and $O(n^{-1/2})$ is the finite sampling error of such a test

Zhang and Spirtes's resolution

- Since Robins et al. (including Spirtes himself) emphatically dis-proved the statistical content of FCI algorithm when latent variable are allowed

Zhang and Spirtes's resolution

- Since Robins et al. (including Spirtes himself) emphatically dis-proved the statistical content of FCI algorithm when latent variable are allowed

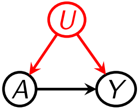
- Peter Spirtes and his then PhD student Jiji Zhang (now Lingnan University in HK) proposed the following resolution:

Zhang and Spirtes's resolution

- Since Robins et al. (including Spirtes himself) emphatically dis-proved the statistical content of FCI algorithm when latent variable are allowed

- Peter Spirtes and his then PhD student Jiji Zhang (now Lingnan University in HK) proposed the following resolution: Let's assume a stronger faithfulness (Zhang, Spirtes. UAI 2003)!

- λ-strong faithfulness: also remove distributions for which the marginal dependence between A and Y is $O\{\lambda\}$

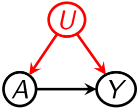


Zhang and Spirtes's resolution

- Since Robins et al. (including Spirtes himself) emphatically dis-proved the statistical content of FCI algorithm when latent variable are allowed

- Peter Spirtes and his then PhD student Jiji Zhang (now Lingnan University in HK) proposed the following resolution:
  Let's assume a stronger faithfulness (Zhang, Spirtes. UAI 2003)!

- λ-strong faithfulness: also remove distributions for which the marginal dependence between A and Y is $O\{\lambda\}$



- Under $(n/\log n)^{1/2}$-strong faithfulness, one can get uniform consistency

Caroline Uhler's counter-argument

- Zhang and Spirtes also showed that by toying with the constants, the Lebesgue measure of distributions violating λ-strong faithfulness can be made arbitrarily small

Caroline Uhler's counter-argument

- Zhang and Spirtes also showed that by toying with the constants, the Lebesgue measure of distributions violating λ-strong faithfulness can be made arbitrarily small

- In 2012, Caroline Uhler proved that Zhang and Spirtes are overly optimistic using tools from Real Algebraic Geometry (an important subject matter for theoretical optimization and theoretical computer science)

Caroline Uhler's counter-argument

- Zhang and Spirtes also showed that by toying with the constants, the Lebesgue measure of distributions violating λ-strong faithfulness can be made arbitrarily small

- In 2012, Caroline Uhler proved that Zhang and Spirtes are overly optimistic using tools from Real Algebraic Geometry (an important subject matter for theoretical optimization and theoretical computer science)

- Out of many interesting results, she showed that for DAGs as simple as a tree with p vertices, the Lebesgue measure of distributions violating λ-strong faithfulness is

$$\geq 1 - (1 - \lambda)^{p-1}$$

Caroline Uhler's counter-argument

- Zhang and Spirtes also showed that by toying with the constants, the Lebesgue measure of distributions violating λ-strong faithfulness can be made arbitrarily small
- In 2012, Caroline Uhler proved that Zhang and Spirtes are overly optimistic using tools from Real Algebraic Geometry (an important subject matter for theoretical optimization and theoretical computer science)
- Out of many interesting results, she showed that for DAGs as simple as a tree with p vertices, the Lebesgue measure of distributions violating λ-strong faithfulness is

$$\geq 1 - (1 - \lambda)^{p-1}$$

- When p is moderately large, this lower bound gets close to 1

Summary of causal discovery controversy

- If some random variables are continuous, whether or not (1) assuming faithfulness and (2) allowing for latent factors, no uniformly consistent causal discovery method exists

Summary of causal discovery controversy

- If some random variables are continuous, whether or not (1) assuming faithfulness and (2) allowing for latent factors, no uniformly consistent causal discovery method exists

- If all random variables are discrete, if no latent factors, without faithfulness, there exist uniformly consistent methods

Summary of causal discovery controversy

- If some random variables are continuous, whether or not (1) assuming faithfulness and (2) allowing for latent factors, no uniformly consistent causal discovery method exists

- If all random variables are discrete, if no latent factors, without faithfulness, there exist uniformly consistent methods

- If all random variables are discrete, with latent factors, without faithfulness, no consistent method exists

Summary of causal discovery controversy

- If some random variables are continuous, whether or not (1) assuming faithfulness and (2) allowing for latent factors, no uniformly consistent causal discovery method exists

- If all random variables are discrete, if no latent factors, without faithfulness, there exist uniformly consistent methods

- If all random variables are discrete, with latent factors, without faithfulness, no consistent method exists

- If all random variables are discrete, with latent factors, with faithfulness, no uniformly consistent method exists

Summary of causal discovery controversy

- If some random variables are continuous, whether or not (1) assuming faithfulness and (2) allowing for latent factors, no uniformly consistent causal discovery method exists

- If all random variables are discrete, if no latent factors, without faithfulness, there exist uniformly consistent methods

- If all random variables are discrete, with latent factors, without faithfulness, no consistent method exists

- If all random variables are discrete, with latent factors, with faithfulness, no uniformly consistent method exists

- But the above does not rule out the possibility of high-quality causal discovery given (1) sufficient background knowledge, (2) correct modeling assumption, and (3) possibility of high-quality randomization or perturbation

Final words about causal discovery

- Despite all these negative results, most of the research in causal inference is about causal discovery because there are demands from applications

Final words about causal discovery

- Despite all these negative results, most of the research in causal inference is about causal discovery because there are demands from applications

- These problems are easier to be mathematized and sound fancier than classical causal inference in statistics and econometrics: reinforcement learning/high-dimensional linear models/neural causal models/...

Final words about causal discovery

- Despite all these negative results, most of the research in causal inference is about causal discovery because there are demands from applications

- These problems are easier to be mathematized and sound fancier than classical causal inference in statistics and econometrics: reinforcement learning/high-dimensional linear models/neural causal models/...

- But do take it with caution – I rarely hear good feedback from applied researchers about IDA/PC/FCI algorithms etc. on structure learning problems, which may mean two different things – the problem is as pessimistic as Robins and Wasserman had warned us; or real scientists just do not give a bleep

Final words about causal discovery

- Despite all these negative results, most of the research in causal inference is about causal discovery because there are demands from applications

- These problems are easier to be mathematized and sound fancier than classical causal inference in statistics and econometrics: reinforcement learning/high-dimensional linear models/neural causal models/...

- But do take it with caution – I rarely hear good feedback from applied researchers about IDA/PC/FCI algorithms etc. on structure learning problems, which may mean two different things – the problem is as pessimistic as Robins and Wasserman had warned us; or real scientists just do not give a bleep

- Combining more and more interventional data might be promising!

Q2). Tian's ID algorithm

- Identification theory with latent variables can be answered by Tian's ID algorithm

Q2). Tian's ID algorithm

- Identification theory with latent variables can be answered by Tian's ID algorithm
  - In fact, Tian and Pearl (2002) only proved ID algorithm is sound (like sufficiency): if ID outputs an identification formula for the causal query, then such formula is correct
    REF: Tian, Pearl. A General Identification Condition for Causal Effects. AAAI 2002.

Q2). Tian's ID algorithm

- Identification theory with latent variables can be answered by Tian's ID algorithm
  - In fact, Tian and Pearl (2002) only proved ID algorithm is sound (like sufficiency): if ID outputs an identification formula for the causal query, then such formula is correct
    REF: Tian, Pearl. A General Identification Condition for Causal Effects. AAAI 2002.
  - Later, Shpitser and Pearl proved ID algorithm is complete (like necessary): if ID outputs "not identifiable", then neither do any other algorithms
    Shpitser, Pearl. Identification of Joint Interventional Distributions in Recursive Semi-Markovian Causal Models. AAAI 2006.
    Shpitser, Pearl. Complete identification methods for the causal hierarchy. JMLR 2008.

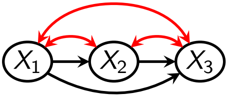
A very brief introduction to theory of ADMGs

CADMGs

ADMGs $\mathcal{G}(V, E)$ with E containing bi-directed edges

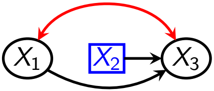
CADMGs (Conditional ADMGs) $\mathcal{G}(V', W, E')$, with V' and W denoting the "random" and "fixed" vertices respectively:

1. $V = V' \cup W$
2. No edges among vertices in W
3. Removing all the arrows into W
4. Turning circles around vertices in W into squares



ADMG:
$\mathcal{G}(\{X_1, X_2, X_3\}, E)$

$\overset{\text{Fixing } X_2}{\Longrightarrow}$



CADMG:
$\mathcal{G}(\{X_1, X_2\}, \{X_3\}, E)$

ID algorithm for ADMG

- Question: Is $p(Y(a) = y)$ or $\mathbb{E}[Y(a)]$ identifiable given an ADMG $\mathcal{G}$ and (subsets of) vertices A and Y?

- Jin Tian (Iowa State U.) solved the problem in his PhD thesis and proposed a complete and sound algorithm, later called Tian's ID algorithm (see next page)

ID algorithm for ADMG

- Question: Is $p(Y(a) = y)$ or $\mathbb{E}[Y(a)]$ identifiable given an ADMG $\mathcal{G}$ and (subsets of) vertices A and Y?

- Jin Tian (Iowa State U.) solved the problem in his PhD thesis and proposed a complete and sound algorithm, later called Tian's ID algorithm (see next page)

- But in this course, we will give you a one-line ID algorithm reformulated using theory of ADMG and "nested Markovian properties"

ID algorithm for ADMG

- Question: Is $p(Y(a) = y)$ or $\mathbb{E}[Y(a)]$ identifiable given an ADMG $\mathcal{G}$ and (subsets of) vertices A and Y?

- Jin Tian (Iowa State U.) solved the problem in his PhD thesis and proposed a complete and sound algorithm, later called Tian's ID algorithm (see next page)

- But in this course, we will give you a one-line ID algorithm reformulated using theory of ADMG and "nested Markovian properties"

- Some papers to read:
  Richardson, Evans, Robins, Shpitser. 2017
  Bhattacharya, Nabi, Shpitser. 2020

```
function $\mathbf{ID}(\mathbf{y}, \mathbf{x}, \mathrm{P}, \mathrm{G})$
INPUT: $\mathbf{x}$,$\mathbf{y}$ value assignments, P a probability distribution, G a causal diagram.
OUTPUT: Expression for $P_{\mathbf{x}}(\mathbf{y})$ in terms of P or FAIL(F,F').

  1 if $\mathbf{x} = \emptyset$ return $\sum_{\mathbf{v} \setminus \mathbf{y}} P(\mathbf{v})$.

  2 if $\mathbf{V} \setminus An(\mathbf{Y})_G \neq \emptyset$
    return $\mathbf{ID}(\mathbf{y}, \mathbf{x} \cap An(\mathbf{Y})_G, \sum_{\mathbf{v} \setminus An(\mathbf{Y})_G} P, G_{An(\mathbf{Y})})$.

  3 let $\mathbf{W} = (\mathbf{V} \setminus \mathbf{X}) \setminus An(\mathbf{Y})_{G_{\overline{\mathbf{X}}}}$.
    if $\mathbf{W} \neq \emptyset$, return $\mathbf{ID}(\mathbf{y}, \mathbf{x} \cup \mathbf{w}, P, G)$.

  4 if $C(G \setminus \mathbf{X}) = \{S_1, ..., S_k\}$
    return $\sum_{\mathbf{v} \setminus (\mathbf{y} \cup \mathbf{x})} \prod_i \mathbf{ID}(s_i, \mathbf{v} \setminus s_i, P, G)$.

    if $C(G \setminus \mathbf{X}) = \{S\}$

      5 if $C(G) = \{G\}$, throw $\mathbf{FAIL}(G, G \cap S)$.
      6 if $S \in C(G)$ return $\sum_{s \setminus \mathbf{y}} \prod_{\{i|V_i \in S\}} P(v_i | v_\pi^{(i-1)})$.
      7 if $(\exists S')S \subset S' \in C(G)$ return $\mathbf{ID}(\mathbf{y}, \mathbf{x} \cap S',$
        $\prod_{\{i|V_i \in S'\}} P(V_i | V_\pi^{(i-1)} \cap S', v_\pi^{(i-1)} \setminus S'), G_{S'})$.
```

Figure 4: A complete identification algorithm. **FAIL** propagates through recursive calls like an exception, and returns the hedge which witnesses non-identifiability. $V_\pi^{(i-1)}$ is the set of nodes preceding V_i in some topological ordering π in G.

Preparatory definitions

- Given any vertex v, district $\text{dis}_{\mathcal{G}}(v)$: maximal bidirected components containing v

Preparatory definitions

- Given any vertex v, district $\text{dis}_{\mathcal{G}}(v)$: maximal bidirected components containing v

- Set of districts $\mathcal{D}(\mathcal{G})$

Preparatory definitions

- Given any vertex v, district $\text{dis}_{\mathcal{G}}(v)$: maximal bidirected components containing v

- Set of districts $\mathcal{D}(\mathcal{G})$

- Markov Blanket: $\text{mb}_{\mathcal{G}}(v) = \text{pa}_{\mathcal{G}}(\text{dis}_{\mathcal{G}}(v)) \cup (\text{dis}_{\mathcal{G}}(v) \setminus \{v\})$

Preparatory definitions

- Given any vertex v, district $\mathrm{dis}_{\mathcal{G}}(v)$: maximal bidirected components containing v
- Set of districts $\mathcal{D}(\mathcal{G})$
- Markov Blanket: $\mathrm{mb}_{\mathcal{G}}(v) = \mathrm{pa}_{\mathcal{G}}(\mathrm{dis}_{\mathcal{G}}(v)) \cup (\mathrm{dis}_{\mathcal{G}}(v) \setminus \{v\})$
- Question: when ADMG $\mathcal{G}$ is a DAG, $\mathrm{mb}_{\mathcal{G}}(v) =?$

Preparatory definitions

- Given any vertex v, district $\text{dis}_{\mathcal{G}}(v)$: maximal bidirected components containing v
- Set of districts $\mathcal{D}(\mathcal{G})$
- Markov Blanket: $\text{mb}_{\mathcal{G}}(v) = \text{pa}_{\mathcal{G}}(\text{dis}_{\mathcal{G}}(v)) \cup (\text{dis}_{\mathcal{G}}(v) \setminus \{v\})$
- Question: when ADMG $\mathcal{G}$ is a DAG, $\text{mb}_{\mathcal{G}}(v) =?$

Proof.

For DAG, $\text{dis}_{\mathcal{G}}(v) = \{v\}$. Then

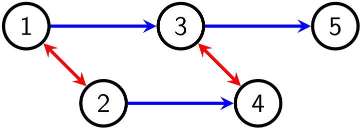
$$\text{mb}_{\mathcal{G}}(v) \equiv \text{pa}_{\mathcal{G}}(v) \cup (\{v\} \setminus \{v\}) \equiv \text{pa}_{\mathcal{G}}(v).$$

□

So Markov Blanket generalizes Parent Set in DAG to ADMG

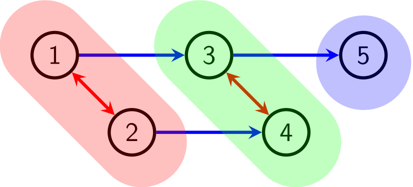
Districts

Define a **district** in a C/ADMG to be maximal sets connected by bi-directed edges:



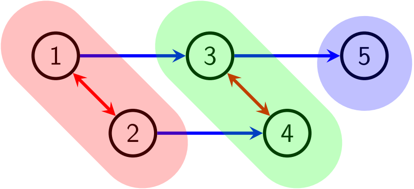
Districts

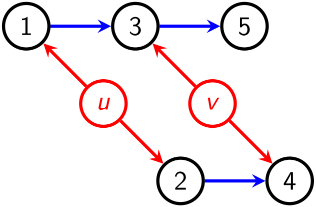
Define a **district** in a C/ADMG to be maximal sets connected by bi-directed edges:



Districts

Define a **district** in a C/ADMG to be maximal sets connected by bi-directed edges:

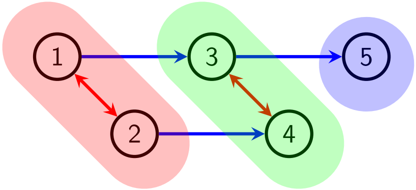


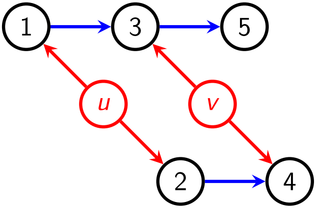


$$\sum_{u,v} p(u)\,p(x_1 \mid u)\,p(x_2 \mid u) \quad p(v)\,p(x_3 \mid x_1, v)\,p(x_4 \mid x_2, v) \quad p(x_5 \mid x_3)$$

Districts

Define a **district** in a C/ADMG to be maximal sets connected by bi-directed edges:

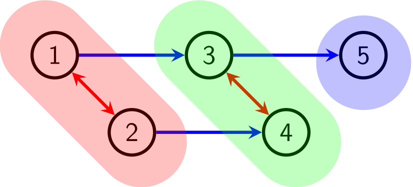


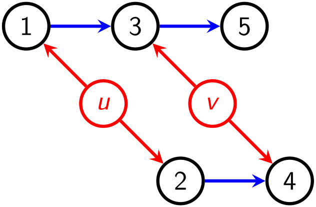


$$\sum_{u,v} p(u)\,p(x_1 \mid u)\,p(x_2 \mid u)\;\; p(v)\,p(x_3 \mid x_1, v)\,p(x_4 \mid x_2, v)\;\; p(x_5 \mid x_3)$$

Districts

Define a **district** in a C/ADMG to be maximal sets connected by bi-directed edges:



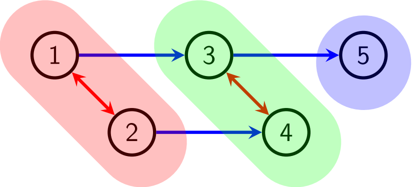


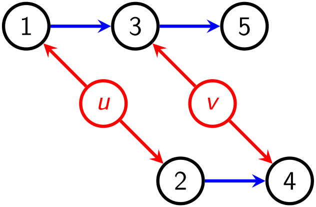
$$\sum_{u,v} p(u)\, p(x_1 \mid u)\, p(x_2 \mid u) \; p(v)\, p(x_3 \mid x_1, v)\, p(x_4 \mid x_2, v) \; p(x_5 \mid x_3)$$

$$= \sum_{u} p(u)\, p(x_1 \mid u)\, p(x_2 \mid u) \sum_{v} p(v)\, p(x_3 \mid x_1, v)\, p(x_4 \mid x_2, v) \; p(x_5 \mid x_3)$$

Districts

Define a **district** in a C/ADMG to be maximal sets connected by bi-directed edges:





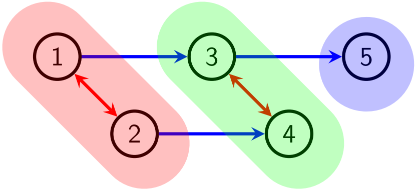
$$\sum_{u,v} p(u)\, p(x_1 \mid u)\, p(x_2 \mid u)\; p(v)\, p(x_3 \mid x_1, v)\, p(x_4 \mid x_2, v)\; p(x_5 \mid x_3)$$

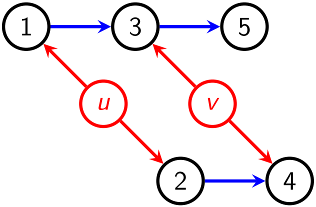
$$= \sum_{u} p(u)\, p(x_1 \mid u)\, p(x_2 \mid u) \sum_{v} p(v)\, p(x_3 \mid x_1, v)\, p(x_4 \mid x_2, v)\, p(x_5 \mid x_3)$$

$$= q(x_1, x_2) \cdot q(x_3, x_4 \mid x_1, x_2) \cdot q(x_5 \mid x_3).$$

Districts

Define a **district** in a C/ADMG to be maximal sets connected by bi-directed edges:





$$\sum_{u,v} p(u)\,p(x_1 \mid u)\,p(x_2 \mid u)\;\; p(v)\,p(x_3 \mid x_1, v)\,p(x_4 \mid x_2, v)\;\; p(x_5 \mid x_3)$$

$$= \sum_{u} p(u)\,p(x_1 \mid u)\,p(x_2 \mid u) \sum_{v} p(v)\,p(x_3 \mid x_1, v)\,p(x_4 \mid x_2, v)\; p(x_5 \mid x_3)$$

$$= q(x_1, x_2) \cdot q(x_3, x_4 \mid x_1, x_2) \cdot q(x_5 \mid x_3).$$

$$= \prod_i q_{D_i}(x_{D_i} \mid x_{\mathrm{pa}(D_i)\setminus D_i})$$

Fixing operation

- Tian's ID algorithm requires one line by introducing the "fixing" operation, which generalizes conditioning and marginalization

Fixing operation

- Tian's ID algorithm requires one line by introducing the “fixing” operation, which generalizes conditioning and marginalization

Fixing operation

- Tian's ID algorithm requires one line by introducing the "fixing" operation, which generalizes conditioning and marginalization

- Fixable vertices

$$F(\mathcal{G}) := \{v \in V : \mathrm{dis}_{\mathcal{G}}(v) \cap \mathrm{de}_{\mathcal{G}}(v) = \{v\}\}$$

In words, v is fixable if no vertex $x \neq v$ s.t.

$$v \leftrightarrow \cdots \leftrightarrow x \text{ and } v \rightarrow \cdots \rightarrow x$$

Trivial implication: singletons are always fixable and vertices in a DAG are always fixable

Fixing operation

- Tian's ID algorithm requires one line by introducing the "fixing" operation, which generalizes conditioning and marginalization

- Fixable vertices

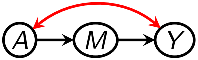
$$F(\mathcal{G}) := \{v \in V : \mathrm{dis}_{\mathcal{G}}(v) \cap \mathrm{de}_{\mathcal{G}}(v) = \{v\}\}$$

  In words, v is fixable if no vertex $x \neq v$ s.t.

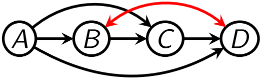
$$v \leftrightarrow \cdots \leftrightarrow x \text{ and } v \rightarrow \cdots \rightarrow x$$

  Trivial implication: singletons are always fixable and vertices in a DAG are always fixable

- Examples:



Front door: $\mathcal{D}_{\mathcal{G}} = \{\{M\},\{A,Y\}\}$,
$F(\mathcal{G}) = \{M,Y\}$



Verma: $\mathcal{D}_{\mathcal{G}} = \{\{A\},\{B,D\},\{C\}\}$,
$F(\mathcal{G}) = \{A,C,D\}$

Fixing operation: Graphical operation

For every $r \in F(\mathcal{G})$, graphically fixing operation is defined as

$$\phi_{\{r\}}(\mathcal{G}) := \mathcal{G}(V \setminus \{r\}, W \cup \{r\}, E')$$

where E' is edge set in the original ADMG $\mathcal{G}$ by removing all edges pointing towards $\{r\}$

Fixing operation: Graphical operation

For every $r \in F(\mathcal{G})$, graphically fixing operation is defined as

$$\phi_{\{r\}}(\mathcal{G}) \coloneqq \mathcal{G}(V \setminus \{r\}, W \cup \{r\}, E')$$

where E' is edge set in the original ADMG $\mathcal{G}$ by removing all edges pointing towards $\{r\}$

Fixing operation might introduce new vertices to the fixable sets

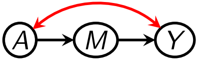
Fixing operation: Graphical operation

For every $r \in F(\mathcal{G})$, graphically fixing operation is defined as

$$\phi_{\{r\}}(\mathcal{G}) := \mathcal{G}(V \setminus \{r\}, W \cup \{r\}, E')$$

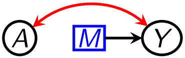
where E' is edge set in the original ADMG $\mathcal{G}$ by removing all edges pointing towards $\{r\}$

Fixing operation might introduce new vertices to the fixable sets



$F(\mathcal{G}) = \{M, Y\}$

$\xrightarrow{\text{Fixing } M}$



$\phi_M(\mathcal{G}), F(\phi_M(\mathcal{G})) = \{A, Y\}$

Fixing operation: Algebraic operation

$p(x_V|x_W)$ is the distribution of all the random vertices of a CADMG $\mathcal{G}(V, W, E)$; fixing a vertex $\{r\}$ means

$$\phi_{\{r\}}(p(x_V|x_W); \mathcal{G}) = \frac{p(x_V|x_W)}{p(x_r|x_{\mathrm{mb}_{\mathcal{G}}(r)})}$$

Fixing operation: Algebraic operation

$p(x_V|x_W)$ is the distribution of all the random vertices of a CADMG $\mathcal{G}(V, W, E)$; fixing a vertex $\{r\}$ means

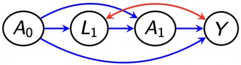
$$\phi_{\{r\}}(p(x_V|x_W); \mathcal{G}) = \frac{p(x_V|x_W)}{p(x_r|x_{\mathrm{mb}_{\mathcal{G}}(r)})}$$

If $r \in F(\mathcal{G})$, then $\phi_{\{r\}}(p(x_V|x_W); \mathcal{G}) \equiv p(x_{V\setminus\{r\}}|x_{W\cup\{r\}})$

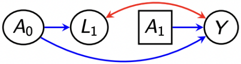
Sequential randomized trial example

Example: Sequential Randomization

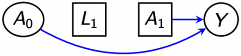
$\mathcal{G}$



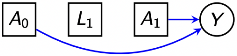
$\phi_{A_1}(\mathcal{G})$



$\phi_{L_1} \circ \phi_{A_1}(\mathcal{G})$

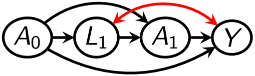


$\phi_{A_0} \circ \phi_{L_1} \circ \phi_{A_1}(\mathcal{G})$

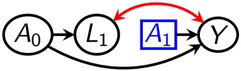


This establishes that $P(Y \mid \text{do}(A_0, A_1))$ is identified.

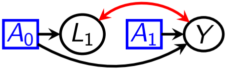
Using fixing to derive the ID formula



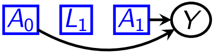
$p(a_0, \ell_1, a_1, y) \equiv$
$p(a_0)p(a_1|a_0, \ell_1)q(\ell_1, y|a_0, a_1)$,
where $q(\ell_1, y|a_0, a_1) =$
$\int p(\ell_1|u, a_0)p(y|u, a_0, \ell_1, a_1)p(u)\mathrm{d}u$



Fix A_1: $\dfrac{p(a_0, \ell_1, a_1, y)}{p(a_1|a_0, \ell_1)} \equiv$
$p(a_0)q(\ell_1, y|a_0, a_1) =: p^{(1)}(a_0, \ell_1, y|a_1)$

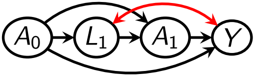


Fix A_0: $\dfrac{p^{(1)}(a_0, \ell_1, y|a_1)}{p(a_0)} \equiv q(\ell_1, y|a_0, a_1) =:$
$p^{(2)}(\ell_1, y|a_0, a_1)$

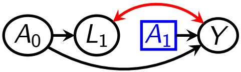


Fix L_1:
$\frac{p^{(2)}(\ell_1, y|a_0, a_1)}{q(\ell_1|a_0, a_1, y)} = q(y|a_0, a_1) =: p^{(3)}(y|a_0, a_1)$

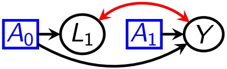
Using fixing to derive the ID formula



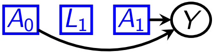
$p(a_0, \ell_1, a_1, y) \equiv$
$p(a_0)p(a_1|a_0, \ell_1)q(\ell_1, y|a_0, a_1)$,
where $q(\ell_1, y|a_0, a_1) =$
$\int p(\ell_1|u, a_0)p(y|u, a_0, \ell_1, a_1)p(u)\mathrm{d}u$



Fix A_1: $\dfrac{p(a_0, \ell_1, a_1, y)}{p(a_1|a_0, \ell_1)} \equiv$
$p(a_0)q(\ell_1, y|a_0, a_1) =: p^{(1)}(a_0, \ell_1, y|a_1)$



Fix A_0: $\dfrac{p^{(1)}(a_0, \ell_1, y|a_1)}{p(a_0)} \equiv q(\ell_1, y|a_0, a_1) =:$
$p^{(2)}(\ell_1, y|a_0, a_1)$



Fix L_1:
$\frac{p^{(2)}(\ell_1, y|a_0, a_1)}{q(\ell_1|a_0, a_1, y)} = q(y|a_0, a_1) =: p^{(3)}(y|a_0, a_1)$

$p^{(3)}(y|a_0, a_1) = q(y|a_0, a_1) = \int_{\ell_1} q(\ell_1, y|a_0, a_1)\mathrm{d}\ell_1$ is the g-formula because

$$q(\ell_1, y|a_0, a_1) = \frac{p(a_0, \ell_1, a_1, y)}{p(a_0)p(a_1|a_0, \ell_1)} = p(\ell_1|a_0)p(y|a_0, \ell_1, a_1)$$

The orders of fixing operations don't matter

- Comparing the above two slides, you will discover that they used two different fixing sequences but both lead to the same ID formula

The orders of fixing operations don't matter

- Comparing the above two slides, you will discover that they used two different fixing sequences but both lead to the same ID formula

- This is the main result that Richardson, Evans, Robins and Shpitser proved in 2017 (Theorem 32 of RERS17): otherwise using fixing operation would have been an absurd idea!

Reachable Subgraphs and Intrinsic Sets

- Before stating Tian's ID algorithm in one line, we need one more definition

Reachable Subgraphs and Intrinsic Sets

- Before stating Tian's ID algorithm in one line, we need one more definition
- Intrinsic set: A set that is a district in a reachable subgraph derived from an ADMG $\mathcal{G}(V, W, E) \equiv \mathcal{G}$

Reachable Subgraphs and Intrinsic Sets

- Before stating Tian's ID algorithm in one line, we need one more definition
- Intrinsic set: A set that is a district in a reachable subgraph derived from an ADMG $\mathcal{G}(V, W, E) \equiv \mathcal{G}$
- Reachable subgraph: A subgraph $\mathcal{G}'$ of $\mathcal{G}$ is said to be reachable from $\mathcal{G}$ if there exists a sequence of fixable vertices $\boldsymbol{w} = (w_1, \cdots, w_T)$ such that $\mathcal{G}' = \phi_{w_T} \circ \cdots \circ \phi_{w_1}(\mathcal{G})$

Reachable Subgraphs and Intrinsic Sets

- Before stating Tian's ID algorithm in one line, we need one more definition

- Intrinsic set: A set that is a district in a reachable subgraph derived from an ADMG $\mathcal{G}(V, W, E) \equiv \mathcal{G}$

- Reachable subgraph: A subgraph $\mathcal{G}'$ of $\mathcal{G}$ is said to be reachable from $\mathcal{G}$ if there exists a sequence of fixable vertices $\boldsymbol{w} = (w_1, \cdots, w_T)$ such that $\mathcal{G}' = \phi_{w_T} \circ \cdots \circ \phi_{w_1}(\mathcal{G})$

- The set of all intrinsic sets in $\mathcal{G}$ is denoted as $\mathcal{I}(\mathcal{G})$

Tian's ID algorithm

Generalizing the above special case, Tian's ID algorithm can be formulated as follows

Theorem 4 (Theorem 49 of RERS17)

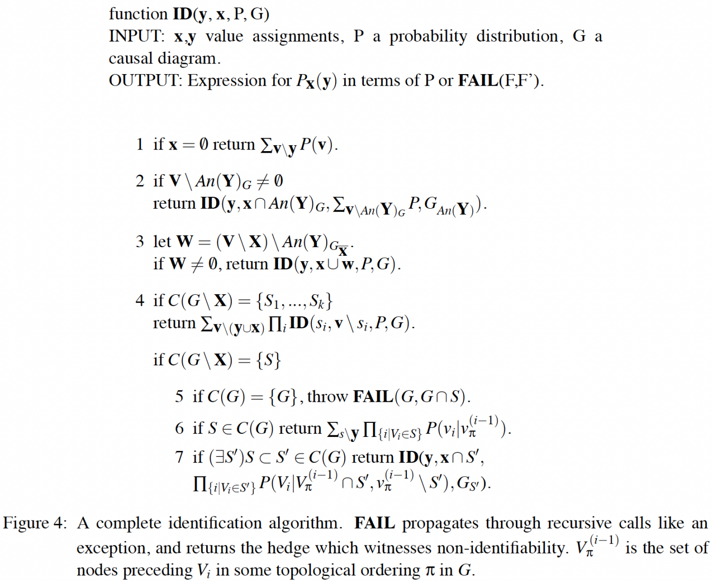
Given an ADMG $\mathcal{G}(V, E) \equiv \mathcal{G}$ and two disjoint subsets $A, Y \subseteq V$, let $\overleftarrow{Y} := \mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)$. If $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) \subseteq \mathcal{I}(\mathcal{G})$, then

$$\begin{aligned} p(X_Y(x_A) = x_Y) &= \int_{x_{\overleftarrow{Y}\setminus Y}} \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\mathrm{pa}_{\mathcal{G}}(D)\setminus D}) = x_D) \mathrm{d}x_{\overleftarrow{Y}\setminus Y} \\ &= \int_{x_{\overleftarrow{Y}\setminus Y}} \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} \phi_{V\setminus D}(p(x_V); \mathcal{G}) \mathrm{d}x_{\overleftarrow{Y}\setminus Y}. \end{aligned} \tag{1}$$

If not, there exists $D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})$ not in the intrinsic sets and $p(X_Y(x_A) = x_Y)$ is unidentifiable.

Think about the following question: Try to translate the above theorem using SWIGs

For the sake of comparison



function **ID**($\mathbf{y}$, $\mathbf{x}$, P, G)
INPUT: $\mathbf{x}$,$\mathbf{y}$ value assignments, P a probability distribution, G a causal diagram.
OUTPUT: Expression for $P_{\mathbf{x}}(\mathbf{y})$ in terms of P or **FAIL**(F,F').

```
1 if $\mathbf{x} = \emptyset$ return $\sum_{\mathbf{v} \setminus \mathbf{y}} P(\mathbf{v})$.
2 if $\mathbf{V} \setminus An(\mathbf{Y})_G \neq \emptyset$
  return $\mathbf{ID}(\mathbf{y}, \mathbf{x} \cap An(\mathbf{Y})_G, \sum_{\mathbf{V} \setminus An(\mathbf{Y})_G} P, G_{An(\mathbf{Y})})$.
3 let $\mathbf{W} = (\mathbf{V} \setminus \mathbf{X}) \setminus An(\mathbf{Y})_{G_{\overline{\mathbf{X}}}}$.
  if $\mathbf{W} \neq \emptyset$, return $\mathbf{ID}(\mathbf{y}, \mathbf{x} \cup \mathbf{w}, P, G)$.
4 if $C(G \setminus \mathbf{X}) = \{S_1, ..., S_k\}$
  return $\sum_{\mathbf{v} \setminus (\mathbf{y} \cup \mathbf{x})} \prod_i \mathbf{ID}(s_i, \mathbf{v} \setminus s_i, P, G)$.
  if $C(G \setminus \mathbf{X}) = \{S\}$
    5 if $C(G) = \{G\}$, throw $\mathbf{FAIL}(G, G \cap S)$.
    6 if $S \in C(G)$ return $\sum_{s \setminus \mathbf{y}} \prod_{\{i|V_i \in S\}} P(v_i | v_\pi^{(i-1)})$.
    7 if $(\exists S') S \subset S' \in C(G)$ return $\mathbf{ID}(\mathbf{y}, \mathbf{x} \cap S',$
      $\prod_{\{i|V_i \in S'\}} P(V_i | V_\pi^{(i-1)} \cap S', v_\pi^{(i-1)} \setminus S'), G_{S'})$.
```

Figure 4: A complete identification algorithm. **FAIL** propagates through recursive calls like an exception, and returns the hedge which witnesses non-identifiability. $V_\pi^{(i-1)}$ is the set of nodes preceding V_i in some topological ordering π in G.

Intuition of Tian's ID algorithm

- Only the subgraph $\mathcal{G}^*$ of the ancestors of Y, with the causal path to Y not including A, needs to be considered for identifying $p(X_Y(x_A) = x_Y)$

Intuition of Tian's ID algorithm

- Only the subgraph $\mathcal{G}^*$ of the ancestors of Y, with the causal path to Y not including A, needs to be considered for identifying $p(X_Y(x_A) = x_Y)$
- Divide-and-Conquer: Get the set of districts of $\mathcal{G}^*$, then consider district by district

Intuition of Tian's ID algorithm

- Only the subgraph $\mathcal{G}^*$ of the ancestors of Y, with the causal path to Y not including A, needs to be considered for identifying $p(X_Y(x_A) = x_Y)$

- Divide-and-Conquer: Get the set of districts of $\mathcal{G}^*$, then consider district by district

- Then

$$p(X_Y(x_A) = x_Y) = \int_{x_{\overleftarrow{Y} \setminus Y}} \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\mathrm{pa}_{\mathcal{G}}(D) \setminus D}) = x_D) \mathrm{d}x_{\overleftarrow{Y} \setminus Y}$$

Intuition of Tian's ID algorithm

- Only the subgraph $\mathcal{G}^*$ of the ancestors of Y, with the causal path to Y not including A, needs to be considered for identifying $p(X_Y(x_A) = x_Y)$

- Divide-and-Conquer: Get the set of districts of $\mathcal{G}^*$, then consider district by district

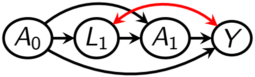
- Then

$$p(X_Y(x_A) = x_Y) = \int_{x_{\overleftarrow{Y} \setminus Y}} \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\mathrm{pa}_{\mathcal{G}}(D) \setminus D}) = x_D) \mathrm{d}x_{\overleftarrow{Y} \setminus Y}$$

- Find if $p(X_D(x_{\mathrm{pa}_{\mathcal{G}}(D) \setminus D}) = x_D)$ is identified

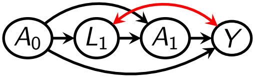
Exercise 1: Verma or sequential randomized trial

$\mathcal{G}$

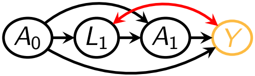


Exercise 1: Verma or sequential randomized trial

$\mathcal{G}$

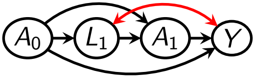


$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V \setminus A}}(Y)$

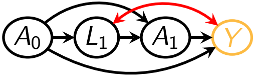


Exercise 1: Verma or sequential randomized trial

$\mathcal{G}$



$\overleftarrow{Y} = \text{an}_{\mathcal{G}_{V\setminus A}}(Y)$

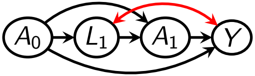


So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{Y\}$, $\text{pa}_{\mathcal{G}}(Y) = \{A_0, A_1\}$ and

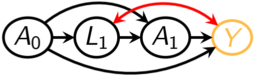
$$\begin{aligned} p(X_Y(x_{A_0}, x_{A_1}) = x_Y) &= \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\text{pa}_{\mathcal{G}}(D)\setminus D}) = x_D) \\ &= p(X_Y(x_{A_0}, x_{A_1}) = x_Y) \end{aligned}$$

Exercise 1: Verma or sequential randomized trial

$\mathcal{G}$



$\overleftarrow{Y} = \text{an}_{\mathcal{G}_{V\setminus A}}(Y)$



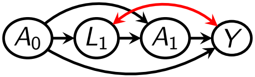
So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{Y\}$, $\text{pa}_{\mathcal{G}}(Y) = \{A_0, A_1\}$ and

$$p(X_Y(x_{A_0}, x_{A_1}) = x_Y) = \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\text{pa}_{\mathcal{G}}(D)\setminus D}) = x_D)$$
$$= p(X_Y(x_{A_0}, x_{A_1}) = x_Y)$$

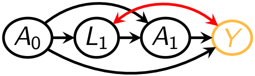
Identified?

Exercise 1: Verma or sequential randomized trial

$\mathcal{G}$



$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V \setminus A}}(Y)$

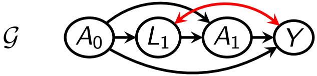


So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{Y\}$, $\mathrm{pa}_{\mathcal{G}}(Y) = \{A_0, A_1\}$ and

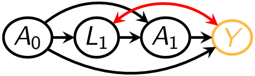
$$
\begin{aligned}
p(X_Y(x_{A_0}, x_{A_1}) = x_Y) &= \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\mathrm{pa}_{\mathcal{G}}(D) \setminus D}) = x_D) \\
&= p(X_Y(x_{A_0}, x_{A_1}) = x_Y)
\end{aligned}
$$

Identified? Is $\{Y\}$ an intrinsic set?

Exercise 1: Verma or sequential randomized trial



$\overleftarrow{Y} = \text{an}_{\mathcal{G}_{V \setminus A}}(Y)$



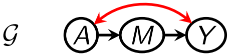
So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{Y\}$, $\text{pa}_{\mathcal{G}}(Y) = \{A_0, A_1\}$ and

$$p(X_Y(x_{A_0}, x_{A_1}) = x_Y) = \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\text{pa}_{\mathcal{G}}(D) \setminus D}) = x_D)$$

$$= p(X_Y(x_{A_0}, x_{A_1}) = x_Y)$$

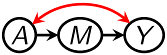
Identified? Is $\{Y\}$ an intrinsic set? Yes! We have seen $\{Y\}$ is reachable by fixing A_1, A_0, L_1

Exercise 2: Front door

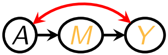


Exercise 2: Front door

$\mathcal{G}$

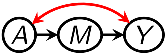


$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V \setminus A}}(Y)$

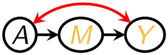


Exercise 2: Front door

$\mathcal{G}$



$\overleftarrow{Y} = \text{an}_{\mathcal{G}_{V\setminus A}}(Y)$



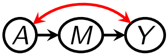
So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{\{M\}, \{Y\}\}$, $\text{pa}_{\mathcal{G}}(M) = \{A\}$, $\text{pa}_{\mathcal{G}}(Y) = \{M\}$ and

$$p(X_Y(x_A) = x_Y) = \int_{x_{\overleftarrow{Y}\setminus Y}} \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\text{pa}_{\mathcal{G}}(D)\setminus D}) = x_D)\mathrm{d}x_{\overleftarrow{Y}\setminus Y}$$

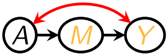
$$= \int_{x_M} p(X_Y(x_M) = x_Y)p(X_M(x_A) = x_M)\mathrm{d}x_M$$

Exercise 2: Front door

$\mathcal{G}$



$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)$



So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{\{M\},\{Y\}\}$, $\mathrm{pa}_{\mathcal{G}}(M) = \{A\}$, $\mathrm{pa}_{\mathcal{G}}(Y) = \{M\}$ and

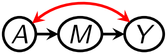
$$p(X_Y(x_A) = x_Y) = \int_{x_{\overleftarrow{Y}\setminus Y}} \prod_{D\in\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\mathrm{pa}_{\mathcal{G}}(D)\setminus D}) = x_D)\mathrm{d}x_{\overleftarrow{Y}\setminus Y}$$

$$= \int_{x_M} p(X_Y(x_M) = x_Y)p(X_M(x_A) = x_M)\mathrm{d}x_M$$

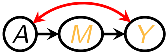
Identified?

Exercise 2: Front door

$\mathcal{G}$



$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V \setminus A}}(Y)$



So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{\{M\}, \{Y\}\}$, $\mathrm{pa}_{\mathcal{G}}(M) = \{A\}$, $\mathrm{pa}_{\mathcal{G}}(Y) = \{M\}$ and

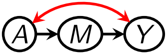
$$p(X_Y(x_A) = x_Y) = \int_{x_{\overleftarrow{Y} \setminus Y}} \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\mathrm{pa}_{\mathcal{G}}(D) \setminus D}) = x_D) \mathrm{d}x_{\overleftarrow{Y} \setminus Y}$$

$$= \int_{x_M} p(X_Y(x_M) = x_Y) p(X_M(x_A) = x_M) \mathrm{d}x_M$$

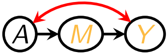
Identified? Are $\{M\}$ and $\{Y\}$ intrinsic sets?

Exercise 2: Front door

$\mathcal{G}$



$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)$

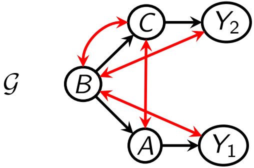


So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{\{M\},\{Y\}\}$, $\mathrm{pa}_{\mathcal{G}}(M) = \{A\}$, $\mathrm{pa}_{\mathcal{G}}(Y) = \{M\}$ and

$$
\begin{aligned}
p(X_Y(x_A) = x_Y) &= \int_{x_{\overleftarrow{Y}\setminus Y}} \prod_{D \in \mathcal{D}(\mathcal{G}_{\overleftarrow{Y}})} p(X_D(x_{\mathrm{pa}_{\mathcal{G}}(D)\setminus D}) = x_D)\mathrm{d}x_{\overleftarrow{Y}\setminus Y} \\
&= \int_{x_M} p(X_Y(x_M) = x_Y) p(X_M(x_A) = x_M)\mathrm{d}x_M
\end{aligned}
$$

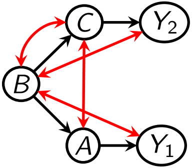
Identified? Are $\{M\}$ and $\{Y\}$ intrinsic sets? Yes for M; Yes for Y by fixing M and A

Exercise 3: causal effect of A on $Y = \{Y_1, Y_2\}$

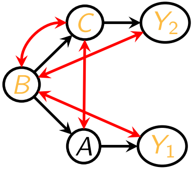


Exercise 3: causal effect of A on $Y = \{Y_1, Y_2\}$

$\mathcal{G}$

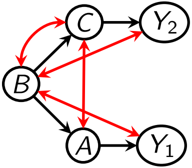


$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V \setminus A}}(Y)$

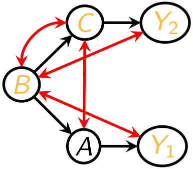


Exercise 3: causal effect of A on $Y = \{Y_1, Y_2\}$

$\mathcal{G}$



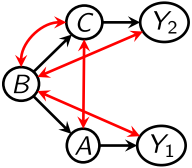
$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V \setminus A}}(Y)$



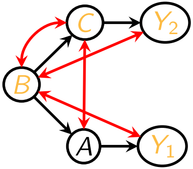
So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{\{B, C, Y_1, Y_2\}\}$ with the only district $D = \{B, C, Y_1, Y_2\}$.

Exercise 3: causal effect of A on $Y = \{Y_1, Y_2\}$

$\mathcal{G}$



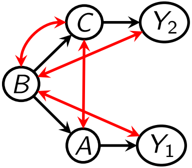
$\overleftarrow{Y} = \text{an}_{\mathcal{G}_{V\setminus A}}(Y)$



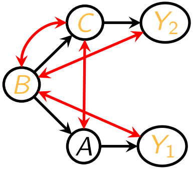
So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{\{B, C, Y_1, Y_2\}\}$ with the only district $D = \{B, C, Y_1, Y_2\}$.
Identified? Is D intrinsic?

Exercise 3: causal effect of A on $Y = \{Y_1, Y_2\}$

$\mathcal{G}$



$\overleftarrow{Y} = \mathrm{an}_{\mathcal{G}_{V\setminus A}}(Y)$



So $\mathcal{D}(\mathcal{G}_{\overleftarrow{Y}}) = \{\{B, C, Y_1, Y_2\}\}$ with the only district $D = \{B, C, Y_1, Y_2\}$.
Identified? Is D intrinsic? No! Because A is not fixable in $\mathcal{G}$!

Optimal ID formula for ADMG?

- All current results focus solely on adjustment formula (i.e. only for single time point case)

Optimal ID formula for ADMG?

- All current results focus solely on adjustment formula (i.e. only for single time point case)

- Smucler, Sapienza, Rotnitzky (SSR) 2020 developed a sound algorithm for ADMGs

Optimal ID formula for ADMG?

- All current results focus solely on adjustment formula (i.e. only for single time point case)
- Smucler, Sapienza, Rotnitzky (SSR) 2020 developed a sound algorithm for ADMGs
- Runge proposed a sound and complete algorithm for ADMGs

Software

- Ilya Shpitser's ananke looks incredible
- We will see some python code using ananke if time permitted

Software

- Ilya Shpitser's ananke looks incredible
- We will see some python code using ananke if time permitted
- There are tons of examples you can try out and its functions include

Software

- Ilya Shpitser's ananke looks incredible
- We will see some python code using ananke if time permitted
- There are tons of examples you can try out and its functions include
  - Differentiable causal discovery/structure learning with linear SEM allowing latent variables to recover certain ADMGs
  - Given an ADMG, whether a causal query is identifiable
  - If over-identified, which formula should we use (Shpitser's group is working on a symbolic computation software just like mathematica or maple)

Summary and Outlook

- ADMG preserves all the Markovian and nested Markovian properties (dormant conditional independencies by fixing) of the underlying latent-variable DAG: these are equality/algebraic constraints

Summary and Outlook

- ADMG preserves all the Markovian and nested Markovian properties (dormant conditional independencies by fixing) of the underlying latent-variable DAG: these are equality/algebraic constraints
- In fact, for certain latent variable DAGs, they also induce certain inequality/semi-algebraic constraints that may be helpful for partial identification

Summary and Outlook

- ADMG preserves all the Markovian and nested Markovian properties (dormant conditional independencies by fixing) of the underlying latent-variable DAG: these are equality/algebraic constraints
- In fact, for certain latent variable DAGs, they also induce certain inequality/semi-algebraic constraints that may be helpful for partial identification
- Except probability and graph/combinatorics, algebraic geometry is another subfield of pure math that is extremely useful for causal inference

Summary and Outlook

- ADMG preserves all the Markovian and nested Markovian properties (dormant conditional independencies by fixing) of the underlying latent-variable DAG: these are equality/algebraic constraints
- In fact, for certain latent variable DAGs, they also induce certain inequality/semi-algebraic constraints that may be helpful for partial identification
- Except probability and graph/combinatorics, algebraic geometry is another subfield of pure math that is extremely useful for causal inference
- We will discuss inequality constraints in next chapter

Any Questions?