

Constructing Separators and Adjustment Sets in Ancestral Graphs



Benito van der Zander
Maciej Liśkiewicz
Theoretical Computer Science
Universität zu Lübeck, Germany



Johannes Textor
Theoretical Biology & Bioinformatics
Universiteit Utrecht, The Netherlands

Outline

What we do

We focus on algorithmic problems motivated by **confirmatory** applications of DAGs and other graphical problems.

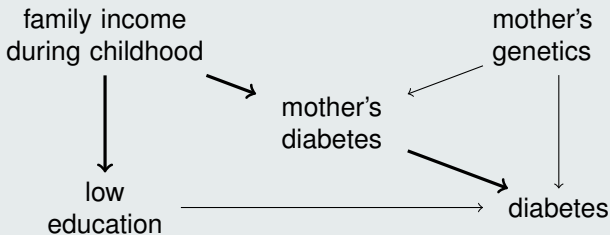
Outline of this talk:

- 1 Motivation
- 2 Algorithmic Framework
- 3 Covariate Adjustment in DAGs
- 4 Covariate Adjustment in MAGs

1 Motivation

Use of DAGs in Epidemiology

How big is the effect of low education on diabetes?



(Rothman, Greenland & Lash, Modern Epidemiology, 2008)

- Epidemiologists use DAGs to represent causal assumptions.
- These DAGs are drawn by hand (most often), generated from data (seldomly), or both (sometimes).
- The work presented in this talk is motivated by what Epidemiologists **do** with DAGs.

The DAGitty Project

- DAGitty is a simple web-based interface to draw and analyse DAGs.
- Focuses on computing adjustment sets and listing testable implications.
- Used mainly in teaching (medical schools) but also research (e.g. Epi, Psych).

Welcome to DAGitty!

 Launch Launch DAGitty online in your browser	 Learn Learn more about DAGs and DAGitty	 Communicate Join the mailing list or contact the author directly	 Download Download DAGitty's source for offline use
---	--	---	--

What is this?

DAGitty is a browser-based environment for creating, editing, and analyzing causal models (also known as directed acyclic graphs or causal Bayesian networks). The focus is on the use of causal diagrams for minimizing bias in empirical studies in epidemiology and other disciplines. For background information, see the "learn" page.

DAGitty is developed and maintained by [Johannes Textor \(Theoretical Biology & Bioinformatics group, University of Utrecht\)](#).

How can I get help?

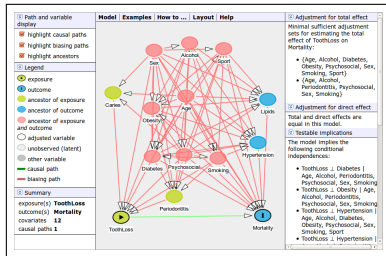
If you encounter any problems using DAGitty, or would like to have a certain feature implemented, please write to "johannes (dot) textor (at) gwu (dot) de". Your feedback and bug reports are very welcome and they contribute to making using DAGitty a better experience for everyone. Past contributors are acknowledged in the [manual](#).

Versions

The following versions of DAGitty are available:

- **Development version**
This is the current development snapshot. May contain new features, but is not thoroughly tested.
- **2.0.0** Released 2014-02-06
- **1.7.1** Released 2011-02-17
- **1.1.1** Released 2011-11-29
- **1.0.0** Released 2011-03-29
- **0.9a** Released 2010-11-24
- **0.9a** Released 2010-02-01

Changelog



The screenshot shows the DAGitty interface with a central causal diagram. The diagram includes nodes for Sex, Alcohol, Sport, Lymph, Hypertension, Mortality, Diabetes, Psychosocial, Smoking, Periodontitis, Obesity, Age, and ToothLoss. The interface is divided into several panels:

- Path and variable display:** Includes options to highlight causal paths, biasing paths, and ancestors.
- Legend:** Defines symbols for exposure, outcome, ancestor, adjusted variable, unobserved variable, other variable, causal path, and biasing path.
- Summary:** Shows exposure(s) as ToothLoss, outcome(s) as Mortality, and 12 covariates.
- Adjustment for total effect:** Lists minimal sufficient adjustment sets for estimating the total effect of ToothLoss on Mortality, including combinations of Age, Alcohol, Diabetes, Obesity, Psychosocial, Sex, Smoking, and Sport.
- Adjustment for direct effect:** Lists total and direct effects equal in this model, including combinations of Age, Alcohol, Diabetes, Obesity, Psychosocial, Sex, Smoking, and Sport.
- Testable implications:** Lists implications such as ToothLoss ⊥ Diabetes, ToothLoss ⊥ Obesity, and ToothLoss ⊥ Hypertension.

Questions for a Causal Diagram

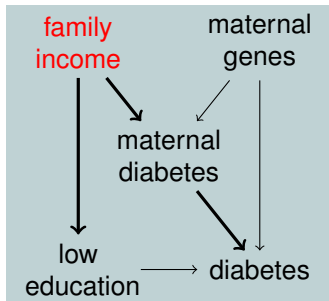
Hi Mr. Textor,

I am trying to learn more about causal diagrams. I want to see if DAGitty can be used for the attached causal diagram to answer a few of my questions. I am having problems with using the program to help answer these questions.

Can you give me some assistance?

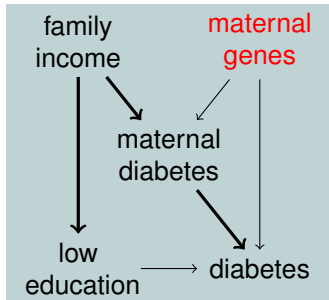
Questions for a Causal Diagram

- 1 Which variable would control for confounding and so reduce bias in estimating the causal effect of the exposure (E) on the disease (D)?



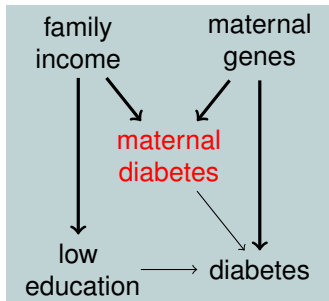
Questions for a Causal Diagram

- 2 Which variable would not impact on the bias in the estimate of causal effect of E on D?



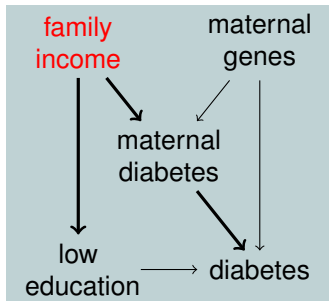
Questions for a Causal Diagram

- 3 Which variable in the model potentially introduces (additional) bias in the estimate of the causal effect of E on D?



Questions for a Causal Diagram

- 4 Which variables would be optimal to (a) estimate an unbiased causal effect of the exposure, (b) maximize the precision and (c) include no unneeded variables?



d-Separation To The Rescue?

*Tell us (...) if conditioning on Z will alter the association between X and Y or leave it intact. But, no cheating, do not use *d*-separation, do it “leaning on the concept of conditional independence, which you do understand.”*

(...)

Don't be surprised if, after 20 minutes of sweat – equations, expectations, covariances, integration, etc. – a student raises his/her hand and asks: Professor, I can see it in the graph!

(...)

*So, is it wise to quit, rather than investing 5 minutes in *d*-separation?*

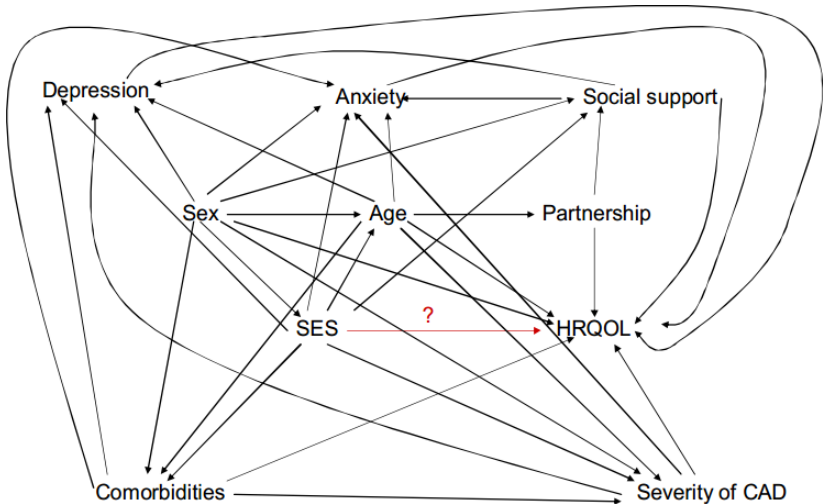
(Judea Pearl, in a discussion on SEMnet)

Back-Door Criterion

To remove bias in a causal effect estimates, find a set Z that *d*-separates all back-door paths from X to Y .

d-Separation To The Rescue?

Find a set Z that d -separates all back-door paths from X to Y .



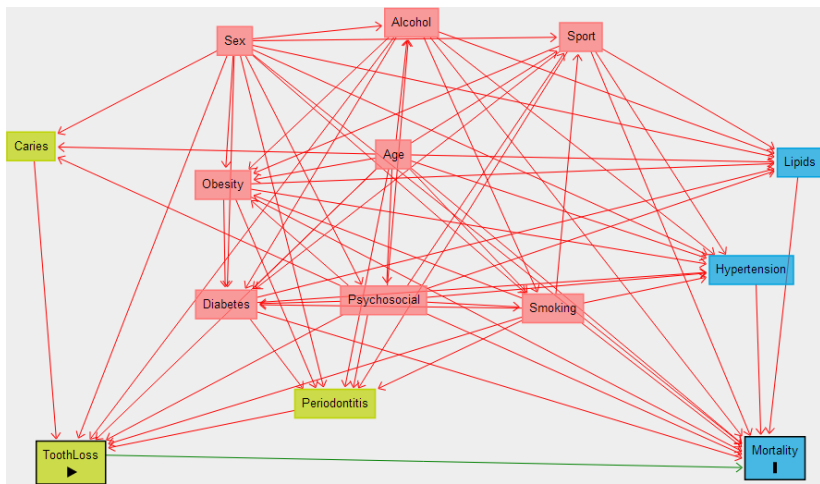
(Sehrndt et al., 2009)

d-Separation To The Rescue?

- For real-world DAGs, path analysis becomes cumbersome.
- In 2009, a German public health master student was assigned the analysis of the DAG on the previous slide.
- It took the person three whole months to find and analyze the ~1000 paths in this DAG.
- As a result, first software for analysing DAGs was developed:
 - *DAG program* (Knueppel & Stang, *Epidemiology* 2010)
 - *dagR* (Breitling, *Epidemiology* 2010) .
- These programs were direct implementations of procedures suggested in Pearl's *Causality* (e.g. back-door criterion).

d-Separation To The Rescue?

Explicit path analysis quickly becomes infeasible for software as well, even for hand-drawn DAGs.



(Polzer et al., personal communication)

2 Algorithmic Framework

Classes of Algorithmic Problems

Consider a relation $R \subseteq X \times Y$ (the input-output-relation).

Existence

i: $x \in X$

o: $\exists y \mid (x, y) \in R$

Complexity classes:

L, NL, P, NP

Counting

i: $x \in X$

o: $\#\{y \mid (x, y) \in R\}$

Complexity classes:

FP, #P

Enumeration

i: $x \in X$

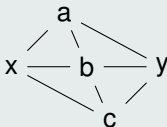
o: $\{y \mid (x, y) \in R\}$

Complexity classes:

n/a

Case I: undirected paths

- Finding one path: **very easy**
- Finding all paths: **very easy**
- Counting all paths: **very hard**



Classes of Algorithmic Problems

Consider a relation $R \subseteq X \times Y$ (the input-output-relation).

Existence

i: $x \in X$

o: $\exists y \mid (x, y) \in R$

Complexity classes:

L, NL, P, NP

Counting

i: $x \in X$

o: $\#\{y \mid (x, y) \in R\}$

Complexity classes:

FP, #P

Enumeration

i: $x \in X$

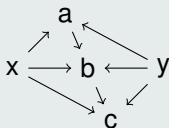
o: $\{y \mid (x, y) \in R\}$

Complexity classes:

n/a

Case II: directed paths

- Finding one path: *easy*
- Finding all paths: *easy*
- Counting all paths: *easy*



Classes of Algorithmic Problems

Consider a relation $R \subseteq X \times Y$ (the input-output-relation).

Existence

i: $x \in X$

o: $\exists y \mid (x, y) \in R$

Complexity classes:

L, NL, P, NP

Counting

i: $x \in X$

o: $\#\{y \mid (x, y) \in R\}$

Complexity classes:

FP, #P

Enumeration

i: $x \in X$

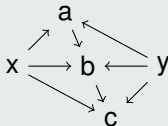
o: $\{y \mid (x, y) \in R\}$

Complexity classes:

n/a

Case III: d -connected paths

- Finding one path: **very easy**
- Finding all paths: **easy**
- Counting all paths: **hard**



Classes of Algorithmic Problems

Consider a relation $R \subseteq X \times Y$ (the input-output-relation).

Existence

i: $x \in X$

o: $\exists y \mid (x, y) \in R$

Complexity classes:
L, NL, P, NP

Counting

i: $x \in X$

o: $\#\{y \mid (x, y) \in R\}$

Complexity classes:
FP, #P

Enumeration

i: $x \in X$

o: $\{y \mid (x, y) \in R\}$

Complexity classes:
n/a

path type	existence	counting
undirected	L-complete	#P-complete
directed (DAGs)	NL-complete	\in FP
d -connected	L-complete	#P-complete

Overview of Our Algorithmic Results

Verification: Given disjoint $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ decide if ...

TESTSEP \mathbf{Z} m -separates \mathbf{X}, \mathbf{Y} $O(n + m)$

TESTMINSEP \mathbf{Z} , but no $\mathbf{Z}' \subsetneq \mathbf{Z}$, m -separates \mathbf{X}, \mathbf{Y} $O(n^2)$

Construction: Given disjoint \mathbf{X}, \mathbf{Y} , output **one** \mathbf{Z} s.t. $\mathbf{I} \subseteq \mathbf{Z} \subseteq \mathbf{R}$ and ...

FINDSEP \mathbf{Z} is an m -separator $O(n + m)$

FINDMINSEP \mathbf{Z} is a minimal m -separator $O(n^2)$

FINDMINCOSTS. \mathbf{Z} is a minimum-cost m -separator $O(n^3)$

Enumeration: Given disjoint \mathbf{X}, \mathbf{Y} , output **all** \mathbf{Z} s.t. $\mathbf{I} \subseteq \mathbf{Z} \subseteq \mathbf{R}$ and ...

LISTSEP \mathbf{Z} is an m -separator $O(n(n + m))$ delay

LISTMINSEP \mathbf{Z} is a minimal m -separator $O(n^3)$ delay

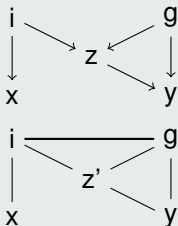
A Key Tool: Moralization

Many problems can be reduced to standard undirected graphs.

Input: AG $G = (V, E)$, vertex sets $X, Y \in V$

Output: A set $Z \subseteq V$ that m -separates X and Y .

The ancestor moral graph G_a^m



- Delete all nodes not in $An(X \cup Y)$
- Link vertices connected by collider paths (e.g. $x \rightarrow v_1 \leftrightarrow \dots \leftrightarrow v_k \leftarrow y$)
- Turn directed into undirected edges
- m -Separator in $\mathcal{G} \Leftrightarrow$ **vertex cut** in G_a^m

However: Moralization takes time $O(n^2)$, and needs to be avoided to achieve linear runtime.

For instance, m -connectedness is solved optimally $O(n + m)$ by a modification of Shachter's "Bayes-Ball" algorithm.

Enumerating m -Separating Sets

Problem

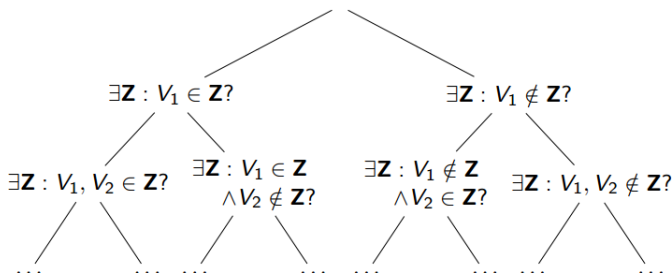
Input: DAG $G = (V, E)$, vertex sets $X, Y \in V$

Output: All minimal sets $Z \subseteq V$ that d -separate X and Y .

- This problem can be solved with **polynomial delay**.
- A polynomial delay algorithm (think Google) outputs each solution after a polynomial waiting time.
- It can be stopped and resumed at any time.
- If no further solution exists, it terminates in polynomial time.
- A polynomial delay algorithm for vertex cuts in undirected graphs was recently presented (*Takata, Discrete Applied Mathematics, 2010*) .

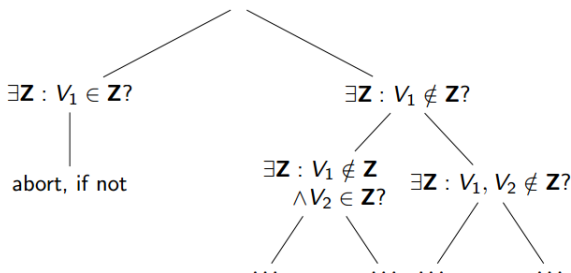
Enumerating all m -Separators with Polynomial Delay

```
function LISTSEP( $\mathcal{G}, \mathbf{X}, \mathbf{Y}, \mathbf{I}, \mathbf{R}$ )  
  if FINDSEP( $\mathcal{G}, \mathbf{X}, \mathbf{Y}, \mathbf{I}, \mathbf{R}$ )  $\neq \perp$  then  
    if  $\mathbf{I} = \mathbf{R}$  then Output  $\mathbf{I}$   
    else  
       $V \leftarrow$  an arbitrary node of  $\mathbf{R} \setminus \mathbf{I}$   
      LISTSEP( $\mathcal{G}, \mathbf{X}, \mathbf{Y}, \mathbf{I} \cup \{V\}, \mathbf{R}$ )  
      LISTSEP( $\mathcal{G}, \mathbf{X}, \mathbf{Y}, \mathbf{I}, \mathbf{R} \setminus \{V\}$ )
```



Enumerating all m -Separators with Polynomial Delay

```
function LISTSEP( $\mathcal{G}, \mathbf{X}, \mathbf{Y}, \mathbf{I}, \mathbf{R}$ )  
  if FINDSEP( $\mathcal{G}, \mathbf{X}, \mathbf{Y}, \mathbf{I}, \mathbf{R}$ )  $\neq \perp$  then  
    if  $\mathbf{I} = \mathbf{R}$  then Output  $\mathbf{I}$   
    else  
       $V \leftarrow$  an arbitrary node of  $\mathbf{R} \setminus \mathbf{I}$   
      LISTSEP( $\mathcal{G}, \mathbf{X}, \mathbf{Y}, \mathbf{I} \cup \{V\}, \mathbf{R}$ )  
      LISTSEP( $\mathcal{G}, \mathbf{X}, \mathbf{Y}, \mathbf{I}, \mathbf{R} \setminus \{V\}$ )
```



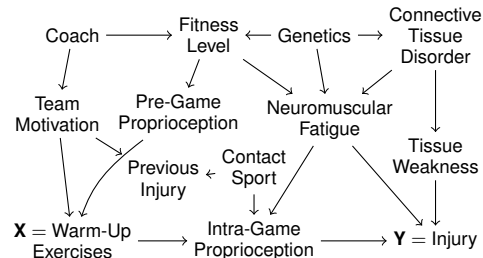
3 Covariate Adjustment in DAGs

Covariate Adjustment

In practice, covariate adjustment is by far the most commonly used technique to estimate causal effects (regression models).

Adjustment Set Construction

Given a graphical model, find sets \mathbf{Z} that fulfill the condition $P(y | \text{do}(x)) = \sum_z P(y | x, z)P(z)$.



8 minimal adjustment sets:

- {Coach, FitnessLevel}
- {Coach, PreGameProprioception}
- {ConnectiveTissueDisorder, NeuromuscularFatigue}
- {FitnessLevel, Genetics}
- {FitnessLevel, TeamMotivation}
- {NeuromuscularFatigue, TissueWeakness}
- {PreGameProprioception, TeamMotivation}

Shrier & Platt, BMC Med Res Meth 2008

Simple Adjustment Criteria

Back-Door Criterion

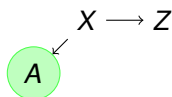
If \mathbf{Z} contains no descendants of \mathbf{X} and m -separates all back-door paths from \mathbf{X} to \mathbf{Y} , then \mathbf{Z} is an adjustment set.

- (+) very intuitive
- (-) not complete

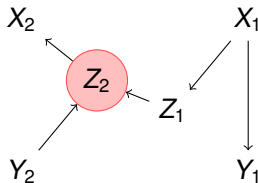
Adjustment for Parents

If all parents of \mathbf{X} (or \mathbf{Y}) are observed variables, then they are an adjustment set.

- (+) very simple
- (-) does not work for $|\mathbf{X}| > 1$



$\{A\}$ is an adjustment set



$Pa(\mathbf{X})$ is not an adjustment set

A Proper Back-Door Criterion

A non-constructive version of the back-door criterion was given by Shpitser et al (UAI 2010).

Adjustment Criterion

\mathbf{Z} is an adjustment set for the causal effect of \mathbf{X} on \mathbf{Y} if and only if

- (a) no element in \mathbf{Z} is a descendant of any $W \in \mathbf{V} \setminus \mathbf{X}$ which lies on a proper causal path from \mathbf{X} to \mathbf{Y} and
- (b) all proper non-causal paths in \mathcal{G} from \mathbf{X} to \mathbf{Y} are blocked by \mathbf{Z} .

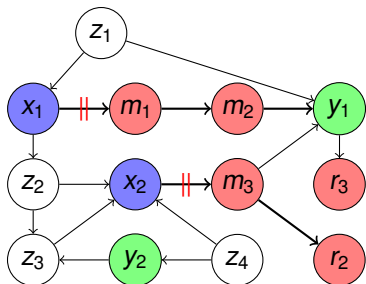
Proper causal path: $x \rightarrow y$

Improper causal path: $x_1 \rightarrow a \rightarrow x_2 \rightarrow y$

Characterizing Separators as Adjustment Sets

Constructive Adjustment Criterion

- Remove the first edge of every proper causal path
- Set $\mathbf{R} = De(\text{True Outcomes} \cup \text{Mediators})$
- \mathbf{Z} is adjustment set if and only if $\mathbf{Z} \subseteq \mathbf{V} \setminus \mathbf{R}$ and \mathbf{Z} *m*-separates \mathbf{X}, \mathbf{Y}

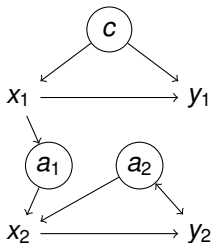


Reduces adjustment set construction to *m*-separation.

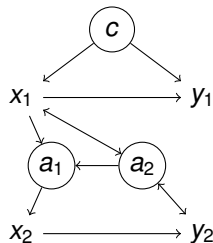
This means we can apply our algorithmic framework to find all adjustment sets.

Constructing a Simple Adjustment Set

Use $\mathbf{Z} = An(\mathbf{X} \cup \mathbf{Y}) \setminus De(\text{True Outcomes} \cup \text{Mediators})$.



Either \mathbf{Z} is an adjustment set,



or no adjustment set exists.

4 Covariate Adjustment in MAGs

DAG Representation by MAGs

Maximum Ancestral Graphs (Richardson & Spirtes, 2002)

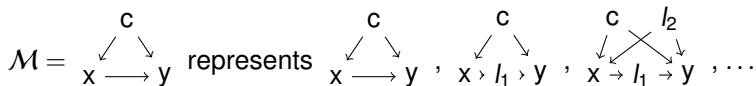
Let $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ be a DAG, and let $\mathbf{S}, \mathbf{L} \subseteq \mathbf{V}$. The MAG $\mathcal{M} = \mathcal{G}_{\mathbf{S}}^{\mathbf{L}}$ is a graph with nodes $\mathbf{V} \setminus (\mathbf{S} \cup \mathbf{L})$ and defined as follows. (1) Two nodes U and V are adjacent in $\mathcal{G}_{\mathbf{S}}^{\mathbf{L}}$ if they cannot be m -separated by any \mathbf{Z} with $\mathbf{S} \subseteq \mathbf{Z} \subseteq \mathbf{V} \setminus \mathbf{L}$ in \mathcal{G} . (2) The edge between U and V is

$U - V$ if $U \in \text{An}(\mathbf{S} \cup V)$ and $V \in \text{An}(\mathbf{S} \cup U)$;

$U \rightarrow V$ if $U \in \text{An}(\mathbf{S} \cup V)$ and $V \notin \text{An}(\mathbf{S} \cup U)$;

$U \leftrightarrow V$ if $U \notin \text{An}(\mathbf{S} \cup V)$ and $V \notin \text{An}(\mathbf{S} \cup U)$.

\mathbf{L} = latent variables; \mathbf{S} = selection variables.



DAG Representation by MAGs

Maximum Ancestral Graphs (Richardson & Spirtes, 2002)

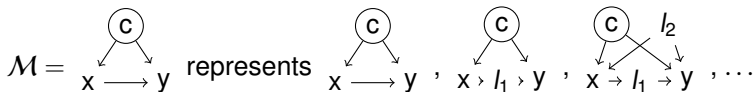
Let $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ be a DAG, and let $\mathbf{S}, \mathbf{L} \subseteq \mathbf{V}$. The MAG $\mathcal{M} = \mathcal{G}_{\mathbf{S}}^{\mathbf{L}}$ is a graph with nodes $\mathbf{V} \setminus (\mathbf{S} \cup \mathbf{L})$ and defined as follows. (1) Two nodes U and V are adjacent in $\mathcal{G}_{\mathbf{S}}^{\mathbf{L}}$ if they cannot be m -separated by any \mathbf{Z} with $\mathbf{S} \subseteq \mathbf{Z} \subseteq \mathbf{V} \setminus \mathbf{L}$ in \mathcal{G} . (2) The edge between U and V is

$U - V$ if $U \in \text{An}(\mathbf{S} \cup V)$ and $V \in \text{An}(\mathbf{S} \cup U)$;

$U \rightarrow V$ if $U \in \text{An}(\mathbf{S} \cup V)$ and $V \notin \text{An}(\mathbf{S} \cup U)$;

$U \leftrightarrow V$ if $U \notin \text{An}(\mathbf{S} \cup V)$ and $V \notin \text{An}(\mathbf{S} \cup U)$.

\mathbf{L} = latent variables; \mathbf{S} = selection variables.



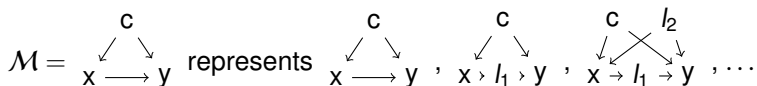
$\mathbf{Z} = \{c\}$ is an adjustment set in some, but not all, represented DAGs.

We consider only MAGs without undirected edges (no selection bias).
Working around selection bias: see Barenboim et al, AAI 2014.

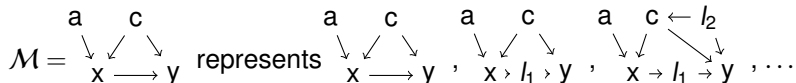
Edge Visibility

Only some directed edges in MAGs are ambiguous.

Invisible edges $x \rightarrow$ can represent non-causal paths.



Visible edges $x \rightarrow$ can only represent causal paths.



Using graphical criteria by Zhang (JMLR 2008), edge visibility of all “first edges” $x \rightarrow$ can be tested in time $O(|\text{children of } \mathbf{X}|(n + m))$.

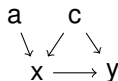
Adjustment in MAGs

If some first edge $x \rightarrow$ on a proper causal path is not visible, then there exists no adjustment set that holds for all represented DAGs: That edge may represent a non-causal path that we can't block.

If all first edges on proper causal paths are visible, we call the MAG **adjustment amenable**.



not adjustment amenable



adjustment amenable

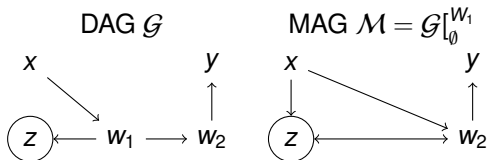
adjustment set: $\{c\}$

For adjustment amenable graphs, we can simply apply the same procedure as for DAGs!

One Bit of the Proof

There is no bijective mapping between non-causal paths in MAGs and their represented DAGs.

Below, \mathbf{Z} contains a descendant of a mediator in the DAG, but not in the corresponding MAG.



We need to show that this leads to an unblockable proper non-causal path.

Towards Robust Adjustment Sets

What's our MAG result worth in confirmatory research? Researchers won't normally draw MAGs due to the causal ambiguities.

A frequent criticism of DAGs:

“Pearl assumes that all plausible models (DAGs) have been properly specified and included among the set of models that are considered.”

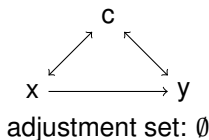
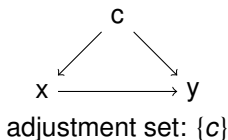
Koch and West, Structural Equation Modelling, 2014

But computed adjustment sets are often valid for many more DAGs than those that were explicitly considered. It is not very easy to determine for which ones exactly.

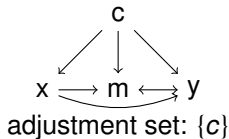
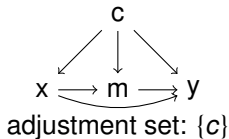
Markov Equivalence Versus Adjustment Equivalence

Let us call two graphs **adjustment equivalent** if they admit exactly the same adjustment sets w.r.t. \mathbf{X}, \mathbf{Y} .

Markov equivalence (being statistically indistinguishable) is not sufficient for adjustment equivalence:



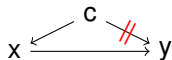
It is also not necessary for adjustment equivalence:



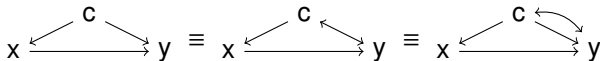
Latent Confounding Robustness

A frequent concern with DAG models is the presence of unobserved confounders. For instance, if we draw only $x \rightarrow y$, how do we know that there is no unobserved variable influencing both?

Let the **transitive reduction** be the unique subgraph of a DAG with the same ancestral relationship.



For all invisible edges $x \rightarrow y$ that are *not* in the transitive reduction, latent confounders do not affect the computation of adjustment sets. This follows simply by reading the DAG as a MAG.



Conclusions

Done:

- We have shown algorithms and constructive criteria to solve various problems in confirmatory causal modelling.
- Most of the algorithms are implemented for DAGs in the open-source tool dagitty.net.

Work in Progress:

- Implementation of the algorithms for MAG.
- We are working on an R package. (Suggestions?)

Future work:

- We think that generalization to CPDAGs and PAGs should be possible (exploratory research).