

Some problems regarding Bayesian networks

পীযুষ শ্রীবাস্তব (Piyush Srivastava), TIFR, Mumbai

RTA Meeting, IACS Kolkata, July 2024

This is a “survey” talk, hoping to serve as an invitation to the combinatorics of Bayesian networks: there will be no proofs!

This is a “survey” talk, hoping to serve as an invitation to the combinatorics of Bayesian networks: there will be no proofs!



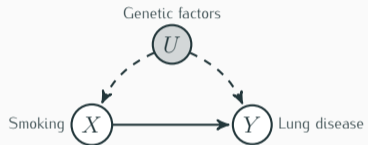
About half of this talk is based on the work of—and discussions with—**Vidya Sagar Sharma**, a PhD student at TIFR.

This is a “survey” talk, hoping to serve as an invitation to the combinatorics of Bayesian networks: there will be no proofs!

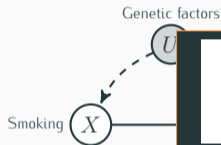


About half of this talk is based on the work of—and discussions with—**Vidya Sagar Sharma**, a PhD student at TIFR. So my role would be somewhat journalistic...

Observation



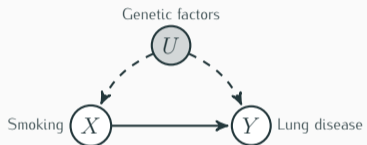
Observation



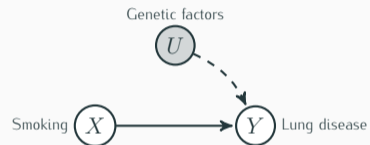
"THERE IS A THEORY OR A HYPOTHESIS THAT DEALS WITH PEOPLE'S PERSONALITIES. THE SAME THING THAT CAUSES PEOPLE TO SMOKE MAY PREDISPOSE THEM TO LUNG CANCER."

Ohlemeyer [1999], quoting an expert

Observation



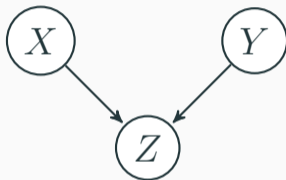
Intervention on X



Several interesting applications, but we focus on the theory in this talk

Bayesian networks (Directed graphical models)

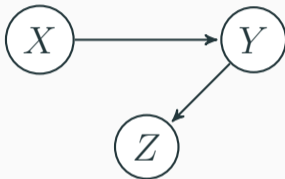
- A **directed acyclic graph** $G = (V, E)$ whose nodes are random variables
- **Absent** edges represent conditional independence assumptions



$$\begin{aligned} P(X, Y, Z) &= P(X)P(Y|X)P(Z|X, Y) \\ &= P(X)P(Y)P(Z|X, Y), \text{ due to model constraints} \end{aligned}$$

Bayesian networks (Directed graphical models)

- A **directed acyclic graph** $G = (V, E)$ whose nodes are random variables
- **Absent** edges represent conditional independence assumptions



$$\begin{aligned} P(X, Y, Z) &= P(X)P(Y|X)P(Z|X, Y) \\ &= P(X)P(Y|X)P(Z|Y), \text{ due to model constraints} \end{aligned}$$

Bayesian networks (Directed graphical models)

- A **directed acyclic graph** $G = (V, E)$ whose nodes are random variables
- **Absent** edges represent conditional independence assumptions



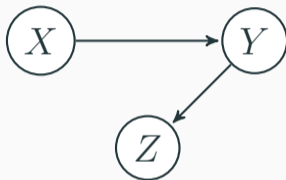
Vertices: Subsystems

Edges: "Causal" relationships between subsystems

$$\begin{aligned} P(X, Y, Z) &= P(X)P(Y|X)P(Z|X, Y) \\ &= P(X)P(Y|X)P(Z|Y), \text{ due to model constraints} \end{aligned}$$

Bayesian networks (Directed graphical models)

- A **directed acyclic graph** $G = (V, E)$ whose nodes are random variables
- **Absent** edges represent conditional independence assumptions



$$\begin{aligned} P(X, Y, Z) &= P(X)P(Y|X)P(Z|X, Y) \\ &= P(X)P(Y|X)P(Z|Y), \text{ due to model constraints} \end{aligned}$$

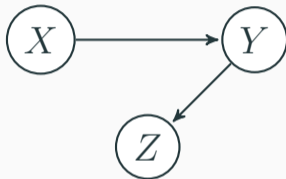
Markov property

If X_1, X_2, \dots, X_n is a **topological sort** of the variables then

X_i independent of $X_{[i-1]}$ **conditioned on its parents**

Bayesian networks (Directed graphical models)

- A **directed acyclic graph** $G = (V, E)$ whose nodes are random variables
- **Absent** edges represent conditional independence assumptions



$$P(X, Y, Z) = P(X)P(Y|X)P(Z|X, Y)$$

= i.e., ordering consistent with the partial order imposed by the DAG model constraints

Markov property

If X_1, X_2, \dots, X_n is a **topological sort** of the variables then

X_i independent of $X_{[i-1]}$ **conditioned on its parents**

(Simple) Examples

What are the conditional independence constraints implied by the following models?



(Simple) Examples

What are the conditional independence constraints implied by the following models?



$$X \perp\!\!\!\perp Z|Y$$



$$X \perp\!\!\!\perp Z|Y$$



(Simple) Examples

What are the conditional independence constraints implied by the following models?



$$X \perp\!\!\!\perp Z|Y$$



$$X \perp\!\!\!\perp Z|Y$$



$$X \perp\!\!\!\perp Z|Y$$



(Simple) Examples

What are the conditional independence constraints implied by the following models?



$$X \perp\!\!\!\perp Z|Y$$



$$X \perp\!\!\!\perp Z|Y$$



$$X \perp\!\!\!\perp Z|Y$$



$$X \perp\!\!\!\perp Z$$

(Simple) Examples

What are the conditional independence constraints implied by the following models?



$$X \perp\!\!\!\perp Z | Y$$



$$X \perp\!\!\!\perp Z | Y$$



$$X \perp\!\!\!\perp Z | Y$$



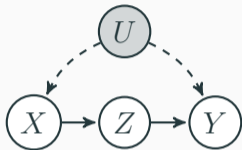
$$X \perp\!\!\!\perp Z$$

Markov equivalence

Bayesian networks are **Markov equivalent** if they entail the **same** set of conditional independence constraints.
(Pearl's **d-separation** method gives an elegant combinatorial characterization of these entailments.)

Semi-Markovian models

- A directed graphical model with some nodes **hidden**; hidden nodes have **no parents**



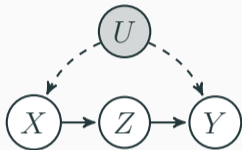
- Represented as $G = (\underbrace{V, E}_{\text{visible}}, \underbrace{U, D}_{\text{hidden}})$

Observed distribution

$$P(X, Y, Z) := \sum_u P(U = u)P(X|U = u)P(Z|X)P(Y|Z, U = u)$$

Semi-Markovian models

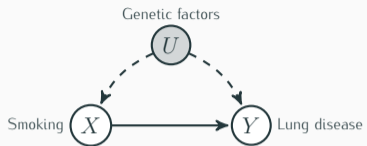
- A directed graphical model with some nodes **hidden**; hidden nodes have **no parents**



- Represented as $G = (\underbrace{V, E}_{\text{visible}}, \underbrace{U, D}_{\text{hidden}})$

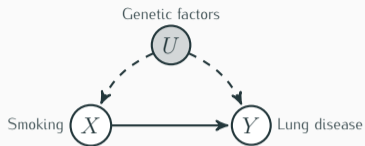
Observed distribution

$$\begin{aligned} P(X, Y, Z) &:= \sum_u P(U = u)P(X|U = u)P(Z|X)P(Y|Z, U = u) \\ &= P(X)P(Z|X)P(Y|Z, X) \end{aligned}$$



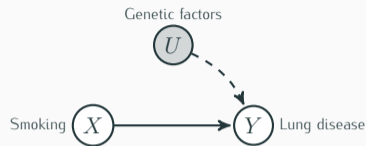
Observational distribution

$$P(X, Y) \\ = \sum_u P(U = u) P(X|u) P(Y|X, u)$$



Observational distribution

$$\begin{aligned}
 &P(X, Y) \\
 &= \sum_u P(U = u) P(X|u) P(Y|X, u)
 \end{aligned}$$



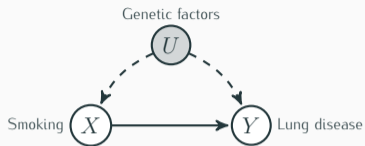
Intervention distribution

$$\begin{aligned}
 &P(Y | \mathbf{do}(X = x)) \\
 &= \sum_u P(U = u) P(Y|X = x, u)
 \end{aligned}$$

Identification problem

[Pearl, 1995]

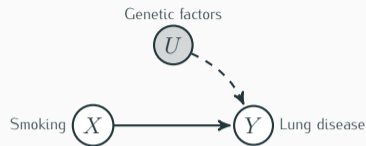
When is $P(Y = y | \mathbf{do}(X = x))$ computable given the observed distribution P ?



Observational distribution

$$P(X, Y)$$

$$= \sum_u P(U = u) P(X|u) P(Y|X, u)$$



Intervention distribution

$$P(Y | \mathbf{do}(X = x))$$

$$= \sum_u P(U = u) P(Y|X = x, u)$$

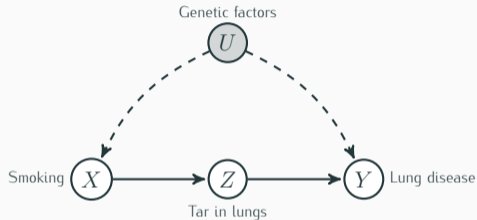
Identification problem

[Pearl, 1995]

When is $P(Y = y | \mathbf{do}(X = x))$ computable given the observed distribution P ?

Not always!

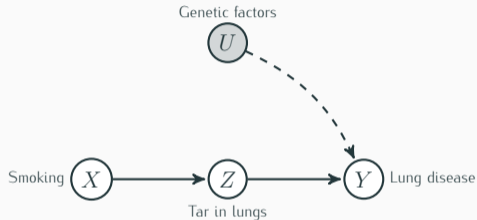
But sometimes it is...



Identification

$$P(Y = y \mid \text{do}(X = x)) = \sum_z P(Z = z \mid X = x) \cdot \sum_{x'} P(X = x') P(Y = y \mid Z = z, X = x').$$

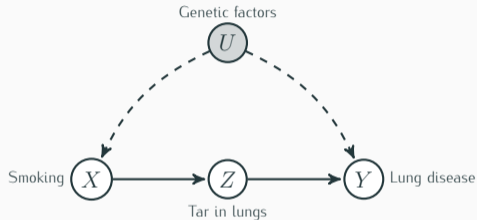
But sometimes it is...



Identification

$$P(Y = y \mid \text{do}(X = x)) = \sum_z P(Z = z \mid X = x) \cdot \sum_{x'} P(X = x') P(Y = y \mid Z = z, X = x').$$

But sometimes it is...

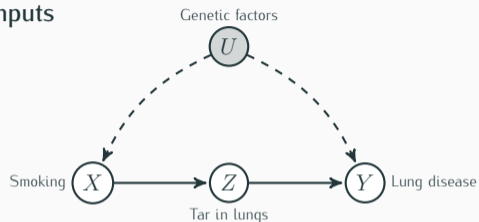


Identification

$$P(Y = y \mid \text{do}(X = x)) = \sum_z P(Z = z \mid X = x) \cdot \sum_{x'} P(X = x') P(Y = y \mid Z = z, X = x').$$

Causal inference: notions of robustness and approximation

Inputs

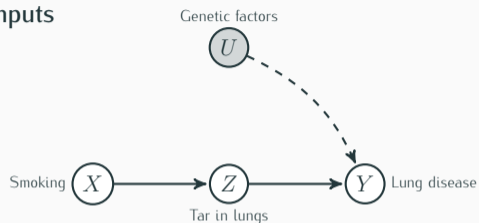


X	Y	Z	P
0	0	0	0.7012
0	0	1	0.0001
1	1	0	0.1001
...
1	1	1	0.0004

Output

$$\underbrace{\text{ID}(G, X, Y)}_{\text{a symbolic formula}} : \underbrace{P}_{\text{"observed marginals"}} \mapsto \underbrace{P(Y \mid \text{do}(X))}$$

Inputs

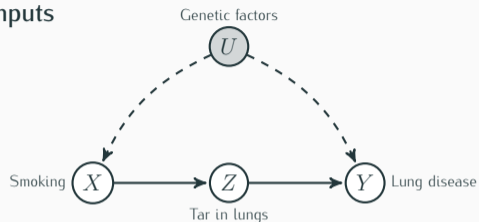


X	Y	Z	P
0	0	0	0.7012
0	0	1	0.0001
1	1	0	0.1001
...
1	1	1	0.0004

Output

$$\underbrace{\text{ID}(G, X, Y)}_{\text{a symbolic formula}} : \underbrace{P}_{\text{"observed marginals"}} \mapsto \underbrace{P(Y \mid \text{do}(X))}_{\substack{\text{"intervention distribution"} \\ \text{"Does smoking cause cancer?"}}}$$

Inputs



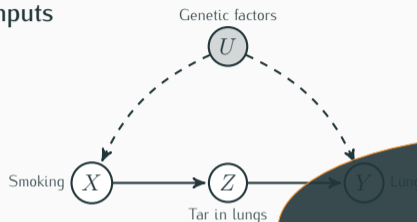
X	Y	Z	P
0	0	0	0.7012
0	0	1	0.0001
1	1	0	0.1001
...
1	1	1	0.0004

Output

$$\underbrace{\text{ID}(G, X, Y)}_{\text{a symbolic formula}} : \underbrace{P}_{\text{"observed marginals"}} \mapsto \underbrace{P(Y \mid \text{do}(X))}_{\substack{\text{"intervention distribution"} \\ \text{"Does smoking cause cancer?"}}}$$

$\text{ID}(G, X, Y)(P)$: the final numerical answer

Inputs



X	Y	Z	P
0	0	0	0.7012
0	0	1	0.0001
1	1	0	0.1001
...
1	1	1	0.0004

Both G and P
are assumed to
be known **exactly**

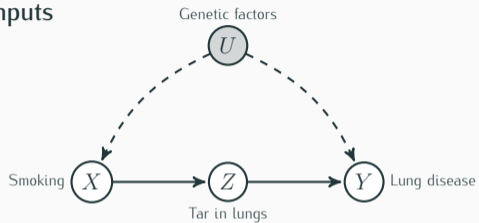
Output

$$\underbrace{\text{ID}(G, X, Y)}_{\text{a symbolic formula}} : \underbrace{P}_{\text{"observed marginals"}} \mapsto \underbrace{P(Y \mid \text{do}(X))}_{\substack{\text{"intervention distribution"} \\ \text{"Does smoking cause cancer?"}}}$$

$\text{ID}(G, X, Y)(P)$: the final numerical answer

Causal identification: Robustness

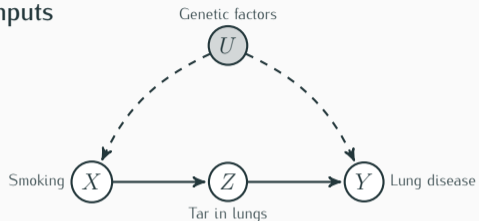
Inputs



X	Y	Z	P
0	0	0	0.7012
0	0	1	0.0001
1	1	0	0.1001
...
1	1	1	0.0004

Causal identification: Robustness

Inputs



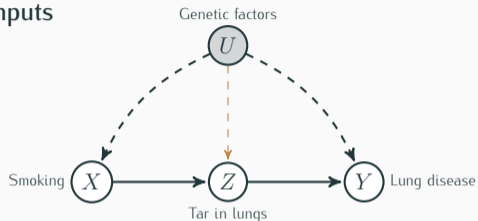
X	Y	Z	P
0	0	0	0.7010
0	0	1	0.0001
1	1	0	0.1003
...
1	1	1	0.0004

Question 1: Numerical stability of $ID(G, X, Y)(P)$

How do numerical errors in P affect causal identification?

Causal identification: Robustness

Inputs



X	Y	Z	P
0	0	0	0.7010
0	0	1	0.0001
1	1	0	0.1003
...
1	1	1	0.0004

Question 1: Numerical stability of $ID(G, X, Y)(P)$

How do numerical errors in P affect causal identification?

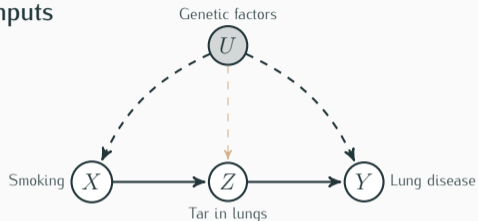
Question 2: Model stability of $ID(G, X, Y)(P)$

How do errors in specification of G affect causal identification?

- the ID algorithm would return FAIL if the **brown edge** is included
- ..but if the influence of the edge is "weak", we should perhaps just try to quantify the error rather than giving up

Causal identification: Robustness

Inputs



X	Y	Z	P
0	0	0	0.7010
0	0	1	0.0001
1	1	0	0.1003
...
1	1	1	0.0004

Question 1: Numerical stability of $ID(G, X, Y)(P)$

How do numerical errors in P affect causal identification?

Question 2: Model stability of $ID(G, X, Y)(P)$

How do errors in specification of G affect causal identification?

- the ID algorithm would return FAIL if the **brown edge** is included
- ..but if the influence of the edge is "weak", we should perhaps just try to quantify the error rather than giving up

Condition number: a classical measure of stability

$$\underbrace{\kappa(f, P)}_{\text{Condition number of } f \text{ at } P} \approx \frac{\text{Relative error in } f(P)}{\text{Relative error in } P} \approx \frac{\frac{|\Delta f(P)|}{|f(P)|}}{\frac{\|\Delta P\|}{\|P\|}}$$

Condition number: a classical measure of stability

A bit like an " ℓ_∞ " version of relative entropy

$$\underbrace{\kappa(f, P)}_{\text{Condition number of } f \text{ at } P} \approx \frac{\text{Relative error in } f(P)}{\text{Relative error in } P} \approx \frac{\frac{|\Delta f(P)|}{|f(P)|}}{\frac{\|\Delta P\|}{\|P\|}}$$

Condition number: a classical measure of stability

$$\underbrace{\kappa(f, P)}_{\text{Condition number of } f \text{ at } P} \approx \frac{\text{Relative error in } f(P)}{\text{Relative error in } P} \approx \frac{\frac{|\Delta f(P)|}{|f(P)|}}{\frac{\|\Delta P\|}{\|P\|}}$$

- Natural and widely used measure of stability in numerical analysis
 - $\log \kappa$ is a measure of **loss of precision**

Condition number: a classical measure of stability

$$\underbrace{\kappa(f, P)}_{\text{Condition number of } f \text{ at } P} \approx \frac{\text{Relative error in } f(P)}{\text{Relative error in } P} \approx \frac{\frac{|\Delta f(P)|}{|f(P)|}}{\frac{\|\Delta P\|}{\|P\|}}$$

- Natural and widely used measure of stability in numerical analysis
 - $\log \kappa$ is a measure of **loss of precision**

But also related to **model** stability

[Schulman and Srivastava, 2016]

Small $\kappa(ID(G, X, Y), P) \implies$ Effect of adding an extra “weak” edge is small

Condition number: a classical measure of stability

$$\underbrace{\kappa(f, P)}_{\text{Condition number of } f \text{ at } P} \approx \frac{\text{Relative error in } f(P)}{\text{Relative error in } P} \approx \frac{\frac{|\Delta f(P)|}{|f(P)|}}{\frac{\|\Delta P\|}{\|P\|}}$$

- Natural and widely used measure of stability in numerical analysis
 - $\log \kappa$ is a measure of **loss of precision**

But also related to **model** stability

[Schulman and Srivastava, 2016]

Small $\kappa(ID(G, X, Y), P) \implies$ Effect of adding an extra “weak” edge is small

Question: Condition number of causal identification

What is the condition number of causal identification of variables X on variables Y in a model G , with given observed marginals P ?

[see also Sankararaman, Louis, and Goyal [2020, 2022] for the “Gaussian” setting]

A quiz



Question

What is the condition number for computing $P(X = 0 \mid \mathbf{do}(S = 0))$?

(Grows linearly with n ? Independent of n ? Some other behavior?)

Assume that the observed marginal P is uniform

A quiz



Question

What is the condition number for computing $P(X = 0 \mid \mathbf{do}(S = 0))$?

(Grows linearly with n ? Independent of n ? Some other behavior?)

Assume that the observed marginal P is uniform

Results: Instance-specific condition number bounds

Goal 1

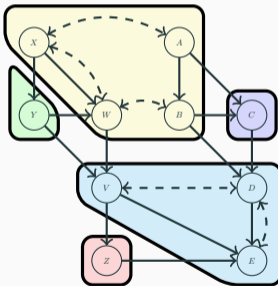
Are there any important classes where the condition number is small?

Results: Instance-specific condition number bounds

Goal 1

Are there any important classes where the condition number is small?

Yes, when the sizes of all **C-components** are small



Results: Instance-specific condition number bounds

Goal 1

Are there any important classes where the condition number is small?

Yes, when the sizes of all **C-components** are small

Theorem

Let G be a causal graph with n **observed nodes**, in which **each C-component has size at most c** . Then for positive P , and for any pair (X, Y) of sets of variables for which $P(Y \mid \mathbf{do}(X))$ is identifiable,

$$\kappa(\text{ID}(G, X, Y), P) \leq n \cdot c^{O(c)}$$

[Gordon, Kumar, Schulman, and Srivastava, 2021]

Results: Tools for instance-specific condition number bounds

Goal 2

How does one algorithmically estimate the condition number of a given causal identification problem?

Results: Tools for instance-specific condition number bounds

Goal 2

How does one algorithmically estimate the condition number of a given causal identification problem?

We have exact expressions for the condition number of causal identification

Results: Tools for instance-specific condition number bounds

Goal 2

How does one algorithmically estimate the condition number of a given causal identification problem?

We have exact expressions for the condition number of causal identification with simplified corollaries

Example corollary

Let G be a causal graph. Then for positive P , and for any pair (X, Y) of sets of variables for which $P(Y \mid \text{do}(X))$ is identifiable,

$$\kappa(\text{ID}(G, X, Y), P) \leq \sqrt{\text{Var}_{\omega \sim P}[\nabla(\log \text{ID}(G, X, Y))(P)]}$$

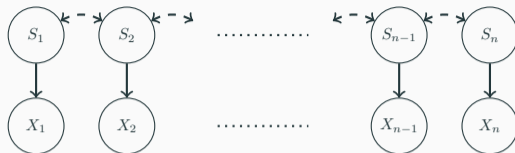


Question

What is the condition number for computing $P(X = 0 \mid \text{do}(S = 0))$?

(Grows linearly with n ? Independent of n ? Some other behavior?)

Assume that the observed marginal P is uniform



Question

What is the condition number for computing $P(X = 0 \mid \text{do}(S = 0))$?

Two “equivalent” expressions

$$\begin{aligned}
 P(X = 0 \mid \text{do}(S = 0)) &= f_1(P) := P(X = 0 \mid S = 0) \\
 &= f_2(P) := \prod_{i=1}^n P(X_i = 0 \mid S_i = 0)
 \end{aligned}$$



Question

What is the condition number for computing $P(X = 0 \mid \text{do}(S = 0))$?

Two “equivalent” expressions

$$\begin{aligned}
 P(X = 0 \mid \text{do}(S = 0)) &= f_1(P) := P(X = 0 \mid S = 0) \\
 &= f_2(P) := \prod_{i=1}^n P(X_i = 0 \mid S_i = 0)
 \end{aligned}$$

But different (and a bit counter-intuitive) stability properties!

$$\kappa(f_1, \text{Uniform}) = \Theta(1) \quad (\text{easy})$$

$$\kappa(f_2, \text{Uniform}) \leq O(n) \quad (\text{easy})$$



Question

What is the condition number for computing $P(X = 0 \mid \text{do}(S = 0))$?

Two “equivalent” expressions

$$\begin{aligned}
 P(X = 0 \mid \text{do}(S = 0)) &= f_1(P) := P(X = 0 \mid S = 0) \\
 &= f_2(P) := \prod_{i=1}^n P(X_i = 0 \mid S_i = 0)
 \end{aligned}$$

But different (and a bit counter-intuitive) stability properties!

$$\kappa(f_1, \text{Uniform}) = \Theta(1) \quad (\text{easy})$$

$$\kappa(f_2, \text{Uniform}) \leq O(n) \quad (\text{easy})$$

$$\kappa(f_2, \text{Uniform}) \sim \sqrt{n} \quad (\text{using tools from [Gordon, Kumar, Schulman, and Srivastava, 2021]})$$



Question

What is the condition number for computing $P(X = 0 \mid \text{do}(S = 0))$?

Two “equivalent” expressions

$P(X = 0 \mid \text{do}(S = 0))$ “trivial” examples!

$$= f_2(P) := \prod_{i=1}^n P(X_i = 0 \mid S_i = 0)$$

But different (and a bit counter-intuitive) stability properties!

$$\kappa(f_1, \text{Uniform}) = \Theta(1) \quad (\text{easy})$$

$$\kappa(f_2, \text{Uniform}) \leq O(n) \quad (\text{easy})$$

$$\kappa(f_2, \text{Uniform}) \sim \sqrt{n} \quad (\text{using tools from [Gordon, Kumar, Schulman, and Srivastava, 2021]})$$

Another problem: Approximate inference (1)

Question

When exact identification is not possible, can we bound $P(Y \mid \mathbf{do}(X))$?

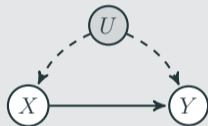
Another problem: Approximate inference (1)

Question

When exact identification is not possible, can we bound $P(Y \mid \mathbf{do}(X))$?

Example

[Tian and Pearl, 2000]



$$P(Y = 1, X = 1) \leq P(Y = 1 \mid \mathbf{do}(X = 1)) \leq 1 - P(Y = 0, X = 1)$$

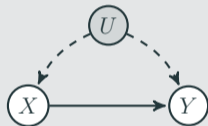
Another problem: Approximate inference (1)

Question

When exact identification is not possible, can we bound $P(Y \mid \mathbf{do}(X))$?

Example

[Tian and Pearl, 2000]



$$P(Y = 1, X = 1) \leq P(Y = 1 \mid \mathbf{do}(X = 1)) \leq 1 - P(Y = 0, X = 1)$$

Question

Can we get a tighter description of the region in which $P(Y \mid \mathbf{do}(X))$ lies when Y and X are not singletons?

Approximate inference (2)

$P(Y \mid \text{do}(X))$ can be seen as describing a **channel** from X to Y

- When $P(Y \mid \text{do}(X))$ is identifiable, the channel is completely determined by the observed distribution P

Approximate inference (2)

$P(Y | \text{do}(X))$ can be seen as describing a **channel** from X to Y

- When $P(Y | \text{do}(X))$ is identifiable, the channel is completely determined by the observed distribution P

Question

When $P(Y | \text{do}(X))$ is **not** identifiable, can we characterize/put bounds on this channel?

Models with hidden variables: recap and future directions

Recap

- $O(n)$ condition number when C-components are of constant size
- Tools for bounding the condition numbers of **any** given instance
- **A surprise:** Condition number may show non-intuitive behaviour even in “trivial” instances

Future directions

Models with hidden variables: recap and future directions

Recap

- $O(n)$ condition number when C-components are of constant size
- Tools for bounding the condition numbers of **any** given instance
- **A surprise:** Condition number may show non-intuitive behaviour even in “trivial” instances

Future directions

“Pre-conditioning” for causal inference?

- How is stability affected by the choice of simpler “equivalent” ID expressions?
e.g., Tikka and Karvanen [2017, 2018]
- How is stability affected by availability of extra information? e.g., Shpitser and Pearl [2008] and Tian and Pearl [2000]

Models with hidden variables: recap and future directions

Recap

- $O(n)$ condition number when C-components are of constant size
- Tools for bounding the condition numbers of **any** given instance
- **A surprise:** Condition number may show non-intuitive behaviour even in “trivial” instances

Future directions

“Pre-conditioning” for causal inference?

- How is stability affected by the choice of simpler “equivalent” ID expressions?
e.g., Tikka and Karvanen [2017, 2018]
- How is stability affected by availability of extra information? e.g., Shpitser and Pearl [2008] and Tian and Pearl [2000]

Approximate causal inference

- Algorithms for computing “channel regions” for interventions

Part 2: Models **without** hidden variables

(Simple) Examples

What are the conditional independence constraints implied by the following models?



$$X \perp\!\!\!\perp Z | Y$$



$$X \perp\!\!\!\perp Z | Y$$



$$X \perp\!\!\!\perp Z | Y$$



$$X \perp\!\!\!\perp Z$$

Markov equivalence

Bayesian networks are **Markov equivalent** if they entail the **same** set of conditional independence constraints.
(Pearl's **d-separation** method gives an elegant combinatorial characterization of these entailments.)

Markov equivalence classes

Markov equivalence

Bayesian networks are **Markov equivalent** if they entail the **same** set of conditional independence constraints.

Markov equivalence classes

Markov equivalence

Bayesian networks are **Markov equivalent** if they entail the **same** set of conditional independence constraints.

Learning the model

Thus, observational data about the system can determine the model...

...only up to its Markov equivalence class (MEC)

Markov equivalence classes

Markov equivalence

Bayesian networks are **Markov equivalent** if they entail the **same** set of conditional independence constraints.

Learning the model

Thus, observational data about the system can determine the model...

...only up to its Markov equivalence class (MEC)

Size of an MEC is a measure of model uncertainty, given only observational data

Aside: Representing an MEC

Partially directed graph representation: **Essential graphs**

An MEC can be represented as a **partially** directed graph, where an edge is **directed** if and only if it has the **same** direction in **every** member of the MEC.

Aside: Representing an MEC

Partially directed graph representation: **Essential graphs**

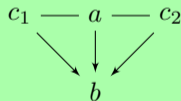
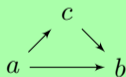
An MEC can be represented as a **partially** directed graph, where an edge is **directed** if and only if it has the **same** direction in **every** member of the MEC.

But all partially directed graphs are not MECs: **essential graphs** are those partially directed graphs which can represent MECs

[Andersson, Madigan, and Perlman, 1997; Chickering, 1995; Meek, 1995]

A partially directed graph G represents an MEC (i.e., is an essential graph) if and only if all the following hold.

1. G is a **chain graph** and its undirected connected components are **chordal**
2. $a \rightarrow b - c$ does not appear as an induced subgraph of G
3. Every directed edge $a \rightarrow b$ in the graph appears as part of at least one of the following induced subgraphs

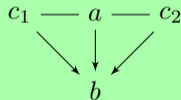
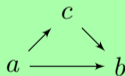


The graph has no par-

[Andersson, Madigan, and Perlman, 1997; Chickering, 1995; Meek, 1995]

A partially directed graph with no partially directed cycles is a MEC (i.e., is an essential graph) if and only if all the following hold.

1. G is a **chain graph** and its undirected connected components are **chordal**
2. $a \rightarrow b - c$ does not appear as an induced subgraph of G
3. Every directed edge $a \rightarrow b$ in the graph appears as part of at least one of the following induced subgraphs

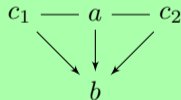
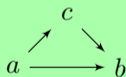


The graph has no partially directed cycles

Undirected graph in which every cycle of length at least four has a "chord"

A partially directed graph is an MEC (i.e., is an essential graph) if the following hold.

1. G is a **chain graph** and its undirected connected components are **chordal**
2. $a \rightarrow b - c$ does not appear as an induced subgraph of G
3. Every directed edge $a \rightarrow b$ in the graph appears as part of at least one of the following induced subgraphs



Recap: Size of MECs and model uncertainty

Markov equivalence

Bayesian networks are **Markov equivalent** if they entail the **same** set of conditional independence constraints.

Recap: Size of MECs and model uncertainty

Markov equivalence

Bayesian networks are **Markov equivalent** if they entail the **same** set of conditional independence constraints.

Learning the model

Thus, observational data about the system can determine the model...

...only up to its Markov equivalence class (MEC)

Recap: Size of MECs and model uncertainty

Markov equivalence

Bayesian networks are **Markov equivalent** if they entail the **same** set of conditional independence constraints.

Learning the model

Thus, observational data about the system can determine the model...

...only up to its Markov equivalence class (MEC)

Size of an MEC is a measure of model uncertainty, given only observational data

Problem 1: Size of an MEC

Size of an MEC: Algorithmic and Statistical perspectives

Algorithmic perspective

INPUT: An MEC G

OUTPUT: The number of DAGs consistent with G

Size of an MEC: Algorithmic and Statistical perspectives

Algorithmic perspective

INPUT: An MEC G

OUTPUT: The number of DAGs consistent with G

Statistical perspective

What is the size of a **typical** MEC?

Size of an MEC: Algorithmic and Statistical perspectives

Algorithmic perspective

INPUT: An MEC G

OUTPUT: The number of DAGs consistent with G

Statistical perspective

What is the size of a **typical** MEC?

What is a **typical** MEC?

Size of an MEC: Algorithmic viewpoint

The problem

INPUT: An MEC G (represented as an **essential graph**)

OUTPUT: The number of DAGs consistent with G .

First considered by Meek [1995], with a lot of recent work exploiting properties of chordal graphs

[Ganian, Hamm, and Talvitie, 2020; Ghassami, Salehkaleybar, Kiyavash, and Zhang, 2019; Talvitie and Koivisto, 2019]

Size of an MEC: Algorithmic viewpoint

The problem

INPUT: An MEC G (represented as an **essential graph**)

OUTPUT: The number of DAGs consistent with G .

First considered by Meek [1995], with a lot of recent work exploiting properties of chordal graphs

[Ganian, Hamm, and Talvitie, 2020; Ghassami, Salehkaleybar, Kiyavash, and Zhang, 2019; Talvitie and Koivisto, 2019]

Culmination

[Wienöbst, Bannach, and Likiewicz, 2021]

A polynomial time exact counting algorithm for the problem

Size of an MEC: Algorithmic viewpoint

The problem

INPUT: An MEC G (represented as an **essential graph**)

OUTPUT: The number of DAGs consistent with G .

First considered by Meek [1995], with a lot of recent work exploiting properties of chordal graphs

[Ganian, Hamm, and Talvitie, 2020; Ghassami, Salehkaleybar, Kiyavash, and Zhang, 2019; Talvitie and Koivisto, 2019]

Culmination

[Wienöbst, Bannach, and Likiewicz, 2021]

A polynomial time exact counting algorithm for the problem

Based on a beautiful augmentation of **lexicographic breadth first search**, a classical tool in the study of chordal graphs

[Rose, Tarjan, and Lueker, 1976]

Size of an MEC: Algorithmic viewpoint

The problem

INPUT: An MEC G (represented as an **essential graph**)

OUTPUT: The number of DAGs consistent with G .

First considered by Meek [1995], with a lot of recent work exploiting properties of chordal graphs

[Ganian, Hamm, and Talvitie, 2020; Ghassami, Salehkaleybar, Kiyavash, and Zhang, 2019; Talvitie and Koivisto, 2019]

Culmination

[Wienöbst, Bannach, and Likiewicz, 2021]

A polynomial time exact counting algorithm for the problem

Based on a beautiful augmentation of **lexicographic breadth first search**, a classical tool in the study of chordal graphs

[Rose, Tarjan, and Lueker, 1976]

Aside: LBFS (and its cousins) look like a “preferential attachment” version of BFS

Size of an MEC with side information: **Background knowledge**

Background knowledge

[Meek, 1995]

Model uncertainty given observational data \approx Size of MECs

Background knowledge: Directions of a few edges beyond those fixed by the MEC

Could be obtained through experimental intervention, or domain-specific knowledge

Size of an MEC with side information: **Background knowledge**

Background knowledge

[Meek, 1995]

Model uncertainty given observational data \approx Size of MECs

Background knowledge: Directions of a few edges beyond those fixed by the MEC

Could be obtained through experimental intervention, or domain-specific knowledge

Size of MECs with background knowledge

INPUT: An MEC M (represented as an **essential graph**), and the direction of a set S of undirected edges in M

OUTPUT: The number of DAGs consistent with G and S .

Size of an MEC with side information: **Background knowledge**

Background knowledge

[Meek, 1995]

Model uncertainty given observational data \approx Size of MECs

Background knowledge: Directions of a few edges beyond those fixed by the MEC

Could be obtained through experimental intervention, or domain-specific knowledge

Size of MECs with background knowledge

INPUT: An MEC M (represented as an **essential graph**), and the direction of a set S of undirected edges in M

OUTPUT: The number of DAGs consistent with G and S .

This problem is #P-hard!

[Wienöbst, Bannach, and Likiewicz, 2021]

Size of an MEC with side information: Background knowledge

Background knowledge

[Meek, 1995]

Model uncertainty given observational data \approx Size of MECs

Background knowledge: Directions of a few edges beyond those fixed by the MEC

Could be obtained through experimental intervention, or domain-specific knowledge

Size of MECs with background knowledge

INPUT: An MEC M (represented as an **essential graph**), and the direction of a set S of undirected edges

As hard as **counting** solutions to Boolean formulas

OUTPUT: The number of DAGs consistent with G and S .

This problem is #P-hard!

[Wienöbst, Bannach, and Likiewicz, 2021]

Size of an MEC with side information: Background knowledge

Size of MECs with background knowledge

INPUT: An MEC G (represented as an **essential graph**), and the direction of a set S of undirected edges in G

OUTPUT: The number of DAGs consistent with G and S .

This problem is #P-hard!!

[Wienöbst, Bannach, and Likiewicz, 2021].

Size of an MEC with side information: Background knowledge

Size of MECs with background knowledge

INPUT: An MEC G (represented as an **essential graph**), and the direction of a set S of

undirected edges E of G .

OUTPUT: The number of DAGs consistent with G and S .



This problem is #P-hard!!

[Wienöbst, Bannach, and Likiewicz, 2021].

But...

[Sharma, 2023]

Size of an MEC with side information: Background knowledge

Size of MECs with background knowledge

INPUT: An MEC G (represented as an **essential graph**), and the direction of a set S of undirected edges in G

OUTPUT: The number of DAGs consistent with G and S .

This problem is #P-hard!!

[Wienöbst, Bannach, and Likiewicz, 2021].

But...

[Sharma, 2023]

- Define

$$k := \max_{\text{Clique } C \text{ in } G} \# \text{ of vertices in } C \cap S$$

Size of an MEC with side information: Background knowledge

Size of MECs with background knowledge

INPUT: An MEC G (represented as an **essential graph**), and the direction of a set S of undirected edges in G

OUTPUT: The number of DAGs consistent with G and S .

This problem is #P-hard!!

[Wienöbst, Bannach, and Likiewicz, 2021].

But...

[Sharma, 2023]

- Define

$$k := \max_{\text{Clique } C \text{ in } G} \# \text{ of vertices in } C \cap S$$

- Note that $k \leq 2|S|$, but it can be much smaller than $|S|$
- **Main result:** An algorithm for the above problem running in time $O(k! \cdot k^2 \cdot |G|^4)$.

So the hardness result can be circumvented, when k is small

Size of MECs with background knowledge: Open question

Problem

Is there a polynomial time **approximation** algorithm for computing size of MECs with background knowledge?

Problem 2: How many MECs?

How many MECs?

The question of how many MECs are there with a **given number of nodes** (and sometimes also with constraints on number of edges) has been studied from a variety of perspectives.

How many MECs?

The question of how many MECs are there with a **given number of nodes** (and sometimes also with constraints on number of edges) has been studied from a variety of perspectives.

- Exact enumeration formulas

[Gillispie and Perlman, 2002]

How many MECs?

The question of how many MECs are there with a **given number of nodes** (and sometimes also with constraints on number of edges) has been studied from a variety of perspectives.

- Exact enumeration formulas [Gillispie and Perlman, 2002]
- Sampling using Markov chains [Bernstein and Tetali, 2017; He, Jia, and Yu, 2013]

Problem

The mixing properties of the chains proposed in these references are not well understood

How many MECs?

The question of how many MECs are there with a **given number of nodes** (and sometimes also with constraints on number of edges) has been studied from a variety of perspectives.

- Exact enumeration formulas [Gillispie and Perlman, 2002]
- Sampling using Markov chains [Bernstein and Tetali, 2017; He, Jia, and Yu, 2013]

Problem

The mixing properties of the chains proposed in these references are not well understood

- Comparisons with the number of all DAGs on n vertices [Bernstein and Tetali, 2017; Schmid and Sly, 2022; Steinsky, 2003, 2013]

Number of MECs with a given skeleton

The problem of understanding the number of MECs with a given **undirected skeleton** has seen much less progress.

MECs with a given skeleton

INPUT: An **undirected** graph G

OUTPUT: The number of MECs with skeleton G

Number of MECs with a given skeleton

The problem of understanding the number of MECs with a given **undirected skeleton** has seen much less progress.

MECs with a given skeleton

INPUT: An **undirected** graph G

OUTPUT: The number of MECs with skeleton G

Previous work was restricted to understanding the problem in the setting of very special graphs G (special classes of trees etc.)

[Radhakrishnan, Solus, and Uhler, 2016, 2018]

Number of MECs with a given skeleton

The problem of understanding the number of MECs with a given **undirected skeleton** has seen much less progress.

MECs with a given skeleton

INPUT: An **undirected** graph G

OUTPUT: The number of MECs with skeleton G

Previous work was restricted to understanding the problem in the setting of very special graphs G (special classes of trees etc.)

[Radhakrishnan, Solus, and Uhler, 2016, 2018]

Recent progress

[Sharma, 2024]

- A polynomial time **exact counting** algorithm for graphs of **bounded degree and treewidth**
- The run time bound is $\exp(\text{poly}(\text{tw}(G), \text{deg}(G))) \cdot |G|^3$

MECs with a given skeleton

INPUT: An **undirected** graph G

OUTPUT: The number of MECs with skeleton G

Recent progress

[Sharma, 2024]

- A polynomial time **exact counting** algorithm for graphs of **bounded degree and treewidth**
- The run time bound is $\exp(\text{poly}(\text{tw}(G), \text{deg}(G))) \cdot |G|^3$

MECs with a given skeleton

INPUT: An **undirected** graph G

OUTPUT: The number of MECs with skeleton G

Recent progress

[Sharma, 2024]

- A polynomial time **exact counting** algorithm for graphs of **bounded degree and treewidth**
- The run time bound is $\exp(\text{poly}(\text{tw}(G), \text{deg}(G))) \cdot |G|^3$

Techniques: short account

1. The idea is to perform dynamic programming over a tree decomposition of the graph
2. Implementing this requires designing encoding extra information about MECs “projected” down to a subset of vertices in such a way that (1) the information is enough to count the corresponding “fibres” and (2) this encoding is succinct enough that it can be enumerated fast

MECs with a given skeleton

INPUT: An **undirected** graph G

OUTPUT: The number of MECs with skeleton G

Recent progress

[Sharma, 2024]

- A polynomial time **exact counting** algorithm for graphs of **bounded degree and treewidth**
- The run time bound is $\exp(\text{poly}(\text{tw}(G), \text{deg}(G))) \cdot |G|^3$

Techniques: short account

1. The idea is to perform dynamic programming over a tree decomposition of the graph
2. Implementing this requires designing encoding extra information about MECs “projected” down to a subset of vertices in such a way that (1) the information is enough to count the corresponding “fibres” and (2) this encoding is succinct enough that it can be enumerated fast
3. These conflicting requirements are captured in a (apparently new) notion that **Sharma [2024]** refers to as **shadow**

Number of MECs with a given skeleton: Open problems

MECs with a given skeleton

INPUT: An **undirected** graph G

OUTPUT: The number of MECs with skeleton G

Problem

What is the exact complexity of this problem? Is it in P, or #P-hard?

Problem

Are there **approximation** algorithms more efficient than the exact algorithm of Sharma [2024]?

Problem 3: MECs in random models

MECs and random graphs

As mentioned earlier, number of MECs on n nodes have been studied in a variety of ways. A recent strong result in this direction is the following

MECs on n nodes

[Schmid and Sly, 2022]

As $n \rightarrow \infty$,

$$\frac{\# \text{ of MECs on } n \text{ nodes}}{\# \text{ of DAGs on } n \text{ nodes}} \rightarrow \text{constant}$$

MECs and random graphs

As mentioned earlier, number of MECs on n nodes have been studied in a variety of ways. A recent strong result in this direction is the following

MECs on n nodes

[Schmid and Sly, 2022]

As $n \rightarrow \infty$,

$$\frac{\# \text{ of MECs on } n \text{ nodes}}{\# \text{ of DAGs on } n \text{ nodes}} \rightarrow \text{constant}$$

Problem

What happens in other models of DAGs? Sparse models?

MECs and random graphs

As mentioned earlier, number of MECs on n nodes have been studied in a variety of ways. A recent strong result in this direction is the following

MECs on n nodes

[Schmid and Sly, 2022]

As $n \rightarrow \infty$,

$$\frac{\# \text{ of MECs on } n \text{ nodes}}{\# \text{ of DAGs on } n \text{ nodes}} \rightarrow \text{constant}$$

Problem

What happens in other models of DAGs? Sparse models?

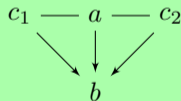
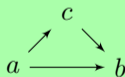
What are the appropriate random network models for this problem?

A short glimpse of basic techniques

[Andersson, Madigan, and Perlman, 1997; Chickering, 1995; Meek, 1995]

A partially directed graph G represents an MEC (i.e., is an essential graph) if and only if all the following hold.

1. G is a **chain graph** and its undirected connected components are **chordal**
2. $a \rightarrow b - c$ does not appear as an induced subgraph of G
3. Every directed edge $a \rightarrow b$ in the graph appears as part of at least one of the following induced subgraphs

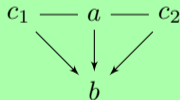
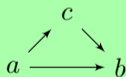


The graph has no par-

[Andersson, Madigan, and Perlman, 1997; Chickering, 1995; Meek, 1995]

A partially directed acyclic graph (PDAG) is a **MEC** (i.e., is an essential graph) if and only if all the following hold.

1. G is a **chain graph** and its undirected connected components are **chordal**
2. $a \rightarrow b - c$ does not appear as an induced subgraph of G
3. Every directed edge $a \rightarrow b$ in the graph appears as part of at least one of the following induced subgraphs



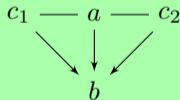
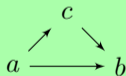
The graph has no par-

[Andersson, Madigan, and Perlman, 1997, Chapter 10]

Undirected graph in which every cycle of length at least four has a "chord"

A partially directed graph is called a MEC (i.e., is an essential graph) if the following hold.

1. G is a **chain graph** and its undirected connected components are **chordal**
2. $a \rightarrow b - c$ does not appear as an induced subgraph of G
3. Every directed edge $a \rightarrow b$ in the graph appears as part of at least one of the following induced subgraphs



Chordal graphs

Chordal graphs

An **undirected** graph G is said to be **chordal** if every simple cycle of length four or more in G has a “chord” (i.e., an edge between two vertices that are not adjacent along the cycle).

Chordal graphs

Chordal graphs

An **undirected** graph G is said to be **chordal** if every simple cycle of length four or more in G has a "chord" (i.e., an edge between two vertices that are not adjacent along the cycle).

Perfect elimination ordering (PEO)

An ordering v_1, v_2, \dots, v_n of vertices of G is a **perfect elimination ordering** if for every v_i , its neighbors in G that precede it in the ordering form a clique.

Chordal graphs

Chordal graphs

An **undirected** graph G is said to be **chordal** if every simple cycle of length four or more in G has a “chord” (i.e., an edge between two vertices that are not adjacent along the cycle).

Perfect elimination ordering (PEO)

An ordering v_1, v_2, \dots, v_n of vertices of G is a **perfect elimination ordering** if for every v_i , its neighbors in G that precede it in the ordering form a clique.

Chordal graphs and perfect elimination orderings

[Dirac, 1961]

A graph G is chordal if and only if it has a perfect elimination ordering

Methods for finding PEOs such as **lexicographic breadth first search** (LBFS) and **maximum cardinality search** (MCS) look like “preferential attachment” version of breadth first search

Chordal graphs: Maximum Cardinality Search

The **maximum cardinality search** algorithm is particularly simple.

Maximum cardinality search

[Tarjan and Yannakakis, 1984]

The algorithm:

1. Start BFS in G starting from any vertex v
2. When choosing which vertex to visit next from those in the BFS queue, choose one with the largest possible number of neighbors among already visited vertices
3. Output the "MCS" order in which vertices are visited

If G is chordal, any MCS ordering is a perfect elimination ordering (PEO).

Aside: special PEOs

In applications, it is sometimes useful to look for PEOs with special properties.

Example: A certain "clique block shared parent" property turned out to be useful in a problem about **learning** Bayesian networks [Porwal, Srivastava, and Sinha, 2022]

Summary

- Several interesting algorithmic and structural questions regarding Bayesian networks
 - Algorithms for counting and sampling
 - Behavior of random models
 - Correct choice of random models

Summary

- Several interesting algorithmic and structural questions regarding Bayesian networks
 - Algorithms for counting and sampling
 - Behavior of random models
 - Correct choice of random models
- This was a very biased survey: there are a whole host of other problems relating to
 - Learning Bayesian networks from data [too many refs. to list!]
 - Different axiomatizations of causal connections

[see, e.g., Richardson and Spirtes, 2002]

Summary

- Several interesting algorithmic and structural questions regarding Bayesian networks
 - Algorithms for counting and sampling
 - Behavior of random models
 - Correct choice of random models
- This was a very biased survey: there are a whole host of other problems relating to
 - Learning Bayesian networks from data [too many refs. to list!]
 - Different axiomatizations of causal connections

[see, e.g., Richardson and Spirtes, 2002]




Summary





- Several interesting algorithmic and structural questions regarding Bayesian networks
 - Algorithms for counting and sampling
 - Behavior of random models
 - Correct choice of random models
- This was a very biased survey: there are a whole host of other problems relating to
 - Learning Bayesian networks from data [too many refs. to list!]
 - Different axiomatizations of causal connections





[see, e.g., Richardson and Spirtes, 2002]




Thank you!

References





-  Andersson, S. A., D. Madigan, and M. D. Perlman (Apr. 1997). **“A Characterization of Markov Equivalence Classes for Acyclic Digraphs.”** In: *Annals of Statistics* 25.2, pp. 505–541. ISSN: 0090-5364, 2168-8966. DOI: 10.1214/aos/1031833662. URL: <https://projecteuclid.org/journals/annals-of-statistics/volume-25/issue-2/A-characterization-of-Markov-equivalence-classes-for-acyclic-digraphs/10.1214/aos/1031833662.full>.
-  Bernstein, M. and P. Tetali (2017). **“On Sampling Graphical Markov Models.”** In: *arXiv:1705.09717*.
-  Chickering, D. M. (1995). **“A Transformational Characterization of Equivalent Bayesian Network Structures.”** In: *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI 1995)*. arXiv:1302.4938, pp. 87–98. URL: <http://arxiv.org/abs/1302.4938>.





-  Dirac, G. A. (Apr. 1961). **“On rigid circuit graphs.”** en. In: *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* 25.1–2, pp. 71–76. ISSN: 0025–5858, 1865–8784. DOI: 10.1007/BF02992776. URL: <http://link.springer.com/10.1007/BF02992776>.
-  Ganian, R., T. Hamm, and T. Talvitie (2020). **“An Efficient Algorithm for Counting Markov Equivalent DAGs.”** In: *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI 2020)*. Vol. 34, pp. 10136–10143.
-  Ghassami, A., S. Salehkaleybar, N. Kiyavash, and K. Zhang (2019). **“Counting and Sampling from Markov Equivalent DAGs using Clique Trees.”** In: *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI 2019)*. Vol. 33, pp. 3664–3671.
-  Gillispie, S. B. and M. D. Perlman (2002). **“The Size Distribution for Markov Equivalence Classes of Acyclic Digraph Models.”** In: *Artificial Intelligence* 141.1–2, pp. 137–155.




-  Gordon, S. L., V. M. Kumar, L. J. Schulman, and P. Srivastava (2021). **“Condition number bounds for causal inference.”** en. In: *Proc. 37th Uncertainty in Artificial Intelligence Conference (UAI)*. PMLR, pp. 1948–1957. URL: <https://proceedings.mlr.press/v161/gordon21a.html>.
-  He, Y., J. Jia, and B. Yu (Aug. 2013). **“Reversible MCMC on Markov equivalence classes of sparse directed acyclic graphs.”** In: *The Annals of Statistics* 41.4, pp. 1742–1779. ISSN: 0090-5364, 2168-8966. DOI: 10.1214/13-AOS1125. URL: <https://projecteuclid.org/journals/annals-of-statistics/volume-41/issue-4/Reversible-MCMC-on-Markov-equivalence-classes-of-sparse-directed-acyclic/10.1214/13-AOS1125.full>.
-  Huang, Y. and M. Valtorta (2006). **“Pearls Calculus of Intervention Is Complete.”** In: *Proc. 22nd Uncertainty in Artificial Intelligence Conference (UAI)*, pp. 217–224.
-  — (2008). **“On the completeness of an identifiability algorithm for semi-Markovian models.”** en. In: *Annals of Mathematics and Artificial Intelligence* 54, pp. 363–408.





-  Meek, C. (1995). **“Causal Inference and Causal Explanation with Background Knowledge.”**
In: *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI 1995)*.
arXiv:1302.4972, pp. 403–410. URL: <http://arxiv.org/abs/1302.4972>.
-  Ohlemeyer, W. S. (1999). **“Closing statement.”** Henley v. Philip Morris Inc. Case No. 995172. 3
February 1999. Superior Court of the State of California. Page 88 in the original, p. 42 in the
digitization. Accessed Nov. 17, 2015. URL:
<https://industrydocuments.library.ucsf.edu/tobacco/docs/#id=frx10001>.
-  Pearl, J. (Dec. 1995). **“Causal diagrams for empirical research.”** en. In: *Biometrika* 82.4,
pp. 669–688. URL: <http://biomet.oxfordjournals.org/content/82/4/669>.





Bibliography v



-  Porwal, V., P. Srivastava, and G. Sinha (May 2022). **“Almost Optimal Universal Lower Bound for Learning Causal DAGs with Atomic Interventions.”** en. In: *Proc. 25th International Conference on Artificial Intelligence and Statistics (AISTATS)*. An extended version is available at arXiv:2111.05070. PMLR, pp. 5583–5603. URL: <https://proceedings.mlr.press/v151/porwal22a.html>.
-  Radhakrishnan, A., L. Solus, and C. Uhler (2016). **“Counting Markov Equivalence Classes by Number of Immoralities.”** In: *arXiv:1611.07493*.
-  — (2018). **“Counting Markov equivalence classes for DAG models on trees.”** In: *Discrete Applied Mathematics* 244, pp. 170–185.
-  Richardson, T. and P. Spirtes (Aug. 2002). **“Ancestral graph Markov models.”** en. In: *The Annals of Statistics* 30.4. ISSN: 0090-5364. DOI: 10.1214/aos/1031689015. URL: <https://projecteuclid.org/journals/annals-of-statistics/volume-30/issue-4/Ancestral-graph-Markov-models/10.1214/aos/1031689015.full> (visited on 07/01/2024).

-  Rose, D. J., R. E. Tarjan, and G. S. Lueker (1976). **“Algorithmic Aspects of Vertex Elimination on Graphs.”** In: *SIAM Journal on Computing* 5.2, pp. 266–283.
-  Sankararaman, K. A., A. Louis, and N. Goyal (2020). **“Stability of Linear Structural Equation Models of Causal Inference.”** en. In: *Proc. 35th Uncertainty in Artificial Intelligence Conference (UAI)*. Vol. 115. PMLR, pp. 323–333. URL: <http://proceedings.mlr.press/v115/sankararaman20a.html>.
-  — (2022). **“Robust Identifiability in Linear Structural Equation Models of Causal Inference.”** In: *Proc. 35th Uncertainty in Artificial Intelligence Conference (UAI)*. Vol. 180. PMLR, pp. 1728–1737. URL: <https://proceedings.mlr.press/v180/sankararaman22a.html>.
-  Schmid, D. and A. Sly (Sept. 2022). ***On the number and size of Markov equivalence classes of random directed acyclic graphs.*** arXiv:2209.04395 [math, stat]. DOI: 10.48550/arXiv.2209.04395. URL: <http://arxiv.org/abs/2209.04395>.

-  Schulman, L. J. and P. Srivastava (June 2016). **“Stability of Causal Inference.”** In: *Proc. 32nd Uncertainty in Artificial Intelligence Conference (UAI)*. An erratum is available at https://www.auai.org/uai2016/proceedings/errata/214_erratum.pdf, pp. 666–675. URL: <http://auai.org/uai2016/proceedings/papers/214.pdf>.
-  Sharma, V. S. (July 2023). **“Counting Background Knowledge Consistent Markov Equivalent Directed Acyclic Graphs.”** en. In: *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*. ISSN: 2640-3498. PMLR, pp. 1911–1920. URL: <https://proceedings.mlr.press/v216/sharma23b.html>.
-  — (2024). **“A Fixed-Parameter Tractable Algorithm for Counting Markov Equivalence Classes with the Same Skeleton.”** In: *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*. Ed. by M. J. Wooldridge, J. G. Dy, and S. Natarajan. arXiv:2310.04218. AAAI Press, pp. 20532–20539. DOI: 10.1609/AAAI.V38I18.30038.

-  Shpitser, I. and J. Pearl (July 2006). **“Identification of Joint Interventional Distributions in Recursive Semi-Markovian Causal Models.”** In: *Proc. 20th AAAI Conference on Artificial Intelligence*, pp. 1219–1226.
-  — (June 2008). **“Complete Identification Methods for the Causal Hierarchy.”** In: *Journal of Machine Learning Research* 9, pp. 1941–1979. URL:
<http://dl.acm.org/citation.cfm?id=1390681.1442797>.
-  Steinsky, B. (Aug. 2003). **“Enumeration of labelled chain graphs and labelled essential directed acyclic graphs.”** en. In: *Discrete Mathematics* 270.1, pp. 267–278. ISSN: 0012–365X. DOI: 10.1016/S0012-365X(02)00838-5. URL:
<http://www.sciencedirect.com/science/article/pii/S0012365X02008385>.
-  — (July 2013). **“Enumeration of Labelled Essential Graphs.”** In: *Ars Combinatoria -Waterloo then Winnipeg-* 111, pp. 485–494.

-  Talvitie, T. and M. Koivisto (2019). **“Counting and Sampling Markov Equivalent Directed Acyclic Graphs.”** In: *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI 2019)*. Vol. 33, pp. 7984–7991.
-  Tarjan, R. E. and M. Yannakakis (Aug. 1984). **“Simple Linear-Time Algorithms to Test Chordality of Graphs, Test Acyclicity of Hypergraphs, and Selectively Reduce Acyclic Hypergraphs.”** In: *SIAM Journal on Computing* 13.3, pp. 566–579. ISSN: 0097-5397. DOI: 10.1137/0213035. URL: <https://epubs.siam.org/doi/abs/10.1137/0213035>.
-  Tian, J. and J. Pearl (2000). **“Probabilities of Causation: Bounds and Identification.”** In: *Proc. 16th Uncertainty in Artificial Intelligence Conference (UAI)*, pp. 589–598.
-  Tikka, S. and J. Karvanen (2017). **“Simplifying Probabilistic Expressions in Causal Inference.”** In: *Journal of Machine Learning Research* 18.36, pp. 1–30. URL: <http://jmlr.org/papers/v18/16-166.html>.

-  Tikka, S. and J. Karvanen (2018). **“Enhancing Identification of Causal Effects by Pruning.”**
In: *Journal of Machine Learning Research* 18.194, pp. 1–23. URL:
<http://jmlr.org/papers/v18/17-563.html>.
-  Wienöbst, M., M. Bannach, and M. Likiewicz (2021). **“Polynomial-Time Algorithms for Counting and Sampling Markov Equivalent DAGs.”** In: *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI 2021)*, pp. 12198–12206.