CSC2541: Introduction to Causality

Lecture 4 - Identification & Estimation

Instructor: Rahul G. Krishnan
TA: Vahid Balazadeh-Meresht

October 3, 2022

Learning directed acyclic graphs

- There are several score based hill-climbing algorithms for structure learning of directed acyclic graphs.

- They learn via the following optimization problem:

$$\min_{\mathcal{G}} \operatorname{loss}(\mathcal{G}) \text{ s.t. } \mathcal{G} \in \text{DAG}$$

- What constitutes a good score function?

- Number should be low if the model *explains* the data and high if it does not.
- When learning $p(y|x)$ we maximize the log-likelihood of labels y given features x to learn parameters of the conditional distribution.
- Posit a class of functions that generates the observations and use fit to data for learning *structure*.

Learning DAGs with linear structural causal models

- We can represent any d-dimensional graph of linear structural causal models in matrix notation as follows:

1. Let $W \in \mathbb{R}^{d \times d}$ be a weight matrix representing the strength of edges and $G(W)$ denote the graph,
2. $B \in \{0,1\}^{d \times d}$ where $B[i,j] = 0 \iff w_{ij} = 0$ is the (binary) adjacency matrix,
3. $x_j = w_j^\top X + \epsilon_j$ where $X = (X_1, \dots, X_d)$ are each dimensions of data (nodes in the graph) and $\epsilon = (\epsilon_1, \dots, \epsilon_d)$ are noise variables,
4. For data matrix D, we can measure fit to data via a least-squares loss $l(W,D) = \frac{1}{2n}||D - DW||_F^2$.
5. We can regularize the loss function to learn a sparse DAG fits the data: $F(W,D) = l(W,D) + \lambda||W||_1$.
6. Finding DAGs then reduces to $\min_{W \in \mathbb{R}^{d \times d}} F(W,D)$ s.t. $G(W) \in \text{DAGs}$

Searching over DAGs

- Optimization problem is NP hard. Challenging due to the constraint in the optimization problem,
- Acyclicity is a combinatorial constraint with the number of structures increasing super exponentially in d,
- DAGS with NO TEARS, Zheng et al., 2018, comes up with a creative solution to this problem!

Insight 1: Binary Adjacency Matrices and cycles

- Fact 1: $\operatorname{tr} B^k$ counts the number of length k closed paths (cycles) in a directed graph,
- Fact 2: DAG has no cycle iff $\sum_{k=1}^{\infty}\sum_{i=1}^{d}(B^k)_{ii} = 0$
- Consequence, B is a DAG iff $\operatorname{tr}(\mathbb{I} - B)^{-1} = d$

$$\begin{aligned}
\operatorname{tr}(\mathbb{I} - B)^{-1} &= \operatorname{tr} \sum_{k=0}^{\infty} B^k && \text{(Infinite geometric series)} \\
&= \operatorname{tr} \mathbb{I} + \operatorname{tr} \sum_{k=1}^{\infty} B^k \\
&= d + \sum_{k=1}^{\infty} \sum_{i=1}^{d} (B^k)_{ii} \\
&= d
\end{aligned}$$

However B^k is difficult to compute and represent in computer memory.

Insight 2: Matrix exponents and weighted graphs

- We can use the matrix exponential $\exp X = \sum_{k=0}^{\infty} \frac{1}{k!} X^k$ which is well-defined.
- Consequence, B is a DAG iff $\operatorname{tr} \exp B = d$, and its extension to the graph with weighted edges (Linear SCM) case yields:

Theorem - Characterizing DAGs with matrix exponents Zheng et al., 2018

A matrix $W \in \mathbb{R}^{d \times d}$ is a DAG iff:

$$h(W) = \operatorname{tr} \exp(W \circ W) - d = 0$$

where $\circ$ is the Hadamard product and

$$\nabla_W h(W) = \exp(W \circ W)^T \circ 2W$$

DAGS with NO TEARS

Smooth characterizations of acyclicity

- $h(W) = 0$ iff W is acyclic (i.e. G(W) represents a DAG),
- $h(W)$ quantifies the DAGness of a graph,
- h is smooth and has easy to compute derivatives.

Now, structure learning of a DAG (under a linear SCM) can be done via : $\min_{W \in \mathbb{R}^{d \times d}} F(W)$ s.t. $h(W) = 0$.

Extensions and future work

- There are non-linear extensions to this idea Lachapelle et al., 2019; Yu et al., 2021; may be interesting to explore for your projects!
- We learn structure and parameters jointly – should we?

Questions?

Question

Any questions on structure learning?

Backdoor criterion and the adjustment formula

Backdoor criterion

A set of variables X satisfies the backdoor criterion relative to sets of variables T and Y in a DAG $\mathcal{G}$ if

1. no node in X is a descendant of a node in T, and
2. X blocks/d-separates **every** path between T and Y that contains an arrow to T (backdoor paths)

In the previous example, sets $\{C\}$ or $\{W\}$ or $\{C, W\}$ all satisfy the backdoor criterion relative to T, Y (but not $\{M\}$).

Theorem - Backdoor adjustment formula

If X satisfies the backdoor criterion relative to T, Y, then the interventional distribution $P(Y|do(T))$ is identifiable and is given by

$$P(Y = y|do(T = t)) = \sum_x P(Y = y|T = t, X = x)P(X = x)$$

Frontdoor criterion and adjustment formula

We were able to identify the causal effect even when the backdoor criterion was not satisfied

Frontdoor criterion

A set of variables M satisfies the frontdoor criterion relative to sets of variables T and Y in a DAG $\mathcal{G}$ if

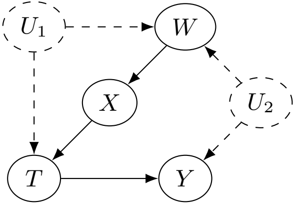
1. M blocks all directed paths from T to Y;
2. no unblocked backdoor path from T to M; and
3. all backdoor paths from M to Y are blocked by T.

Theorem - Frontdoor adjustment formula

If M satisfies the frontdoor criterion relative to T, Y, then the interventional distribution $P(Y|do(T))$ is identifiable and is given by

$$P(Y = y|do(T = t)) = \sum_{m} P(m|t) \sum_{t'} P(y|t', m)P(t')$$

What if backdoor and frontdoor criteria don't work?



We are interested in the causal effect of cardiac output (T) on the blood pressure (Y). X is the heart rate and W is catecholamine (a stress hormone). The levels of total peripheral resistance (U_1) and analgesia (U_2) are unobserved. [1]

- There is an unobserved backdoor path between T and Y, T, U_1, W, U_2, Y: ~~Backdoor criterion~~,
- There is no mediator between T and Y: ~~Frontdoor criterion~~,
- We can use *do*-calculus to decide if $P(Y|do(T))$ is identifiable.

[1] Figure 1.a in Jung, Tian, and Bareinboim, 2021.

Pearl's *do*-calculus

- *do*-calculus is a set of three inference rules that allows us to convert an interventional quantity into a probability expression involving observed quantities
- We'll consider general quantities $P(Y|do(T=t), X=x)$ for arbitrary (sets of) variables T, X, Y

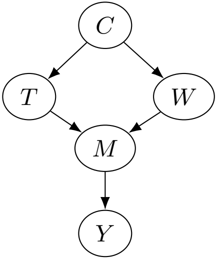
$$P(Y|do(T=t), X=x) := \frac{P(Y, X=x|do(T=t))}{P(X=x|do(T=t))}$$

Pearl's *do*-calculus

- *do*-calculus is a set of three inference rules that allows us to convert an interventional quantity into a probability expression involving observed quantities
- We'll consider general quantities $P(Y|do(T=t), X=x)$ for arbitrary (sets of) variables T, X, Y

$$P(Y|do(T=t), X=x) := \frac{P(Y, X=x|do(T=t))}{P(X=x|do(T=t))}$$

- Notation. Graph $\mathcal{G}$

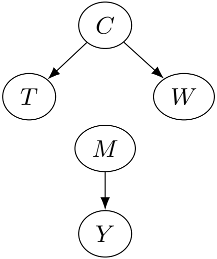


Pearl's *do*-calculus

- *do*-calculus is a set of three inference rules that allows us to convert an interventional quantity into a probability expression involving observed quantities
- We'll consider general quantities $P(Y|do(T=t), X=x)$ for arbitrary (sets of) variables T, X, Y

$$P(Y|do(T=t), X=x) := \frac{P(Y, X=x|do(T=t))}{P(X=x|do(T=t))}$$

- Notation. Graph $\mathcal{G}_{\overline{M}}$

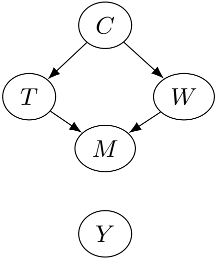


Pearl's *do*-calculus

- *do*-calculus is a set of three inference rules that allows us to convert an interventional quantity into a probability expression involving observed quantities
- We'll consider general quantities $P(Y|do(T=t), X=x)$ for arbitrary (sets of) variables T, X, Y

$$P(Y|do(T=t), X=x) := \frac{P(Y, X=x|do(T=t))}{P(X=x|do(T=t))}$$

- Notation. Graph $\mathcal{G}_{\underline{M}}$

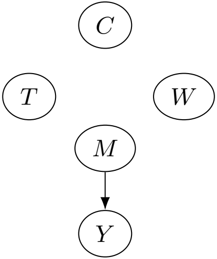


Pearl's *do*-calculus

- *do*-calculus is a set of three inference rules that allows us to convert an interventional quantity into a probability expression involving observed quantities
- We'll consider general quantities $P(Y|do(T=t), X=x)$ for arbitrary (sets of) variables T, X, Y

$$P(Y|do(T=t), X=x) := \frac{P(Y, X=x|do(T=t))}{P(X=x|do(T=t))}$$

- Notation. Graph $\mathcal{G}_{\underline{C},\overline{M}}$



Rule 1 of *do*-calculus - Insertion/deletion of observations

$$P(Y|do(T=t), X, W) = P(Y|do(T=t), W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T}}} X|T, W$$

Rule 1 of *do*-calculus - Insertion/deletion of observations

$$P(Y|do(T=t), X, W) = P(Y|do(T=t), W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T}}} X|T, W$$

Intuition:

- In the interventional/mutilated graph $\mathcal{G}_{\overline{T}}$, every path from T is causal. Therefore we can simplify the rule as:

$$P(Y|T=t, X, W) = P(Y|T=t, W) \text{ if } Y \perp\!\!\!\perp_{\mathcal{G}} X|T, W$$

Generalization of d-separation

Rule 1 of *do*-calculus - Insertion/deletion of observations

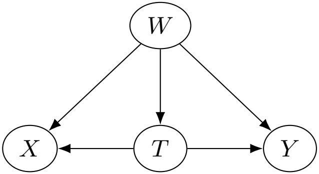
$$P(Y|do(T=t), X, W) = P(Y|do(T=t), W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T}}} X|T, W$$

Intuition:

- In the interventional/mutilated graph $\mathcal{G}_{\overline{T}}$, every path from T is causal. Therefore we can simplify the rule as:

$$P(Y|T=t, X, W) = P(Y|T=t, W) \text{ if } Y \perp\!\!\!\perp_{\mathcal{G}} X|T, W$$

Generalization of d-separation



Rule 1 of *do*-calculus - Insertion/deletion of observations

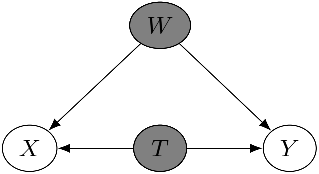
$$P(Y|do(T=t), X, W) = P(Y|do(T=t), W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T}}} X|T,W$$

Intuition:

- In the interventional/mutilated graph $\mathcal{G}_{\overline{T}}$, every path from T is causal. Therefore we can simplify the rule as:

$$P(Y|T=t, X, W) = P(Y|T=t, W) \text{ if } Y \perp\!\!\!\perp_{\mathcal{G}} X|T,W$$

Generalization of d-separation



Rule 2 of *do*-calculus - Action/observation exchange

$$P(Y|do(T=t), do(X=x), W) = P(Y|do(T=t), X=x, W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\underline{X}}} X|T, W$$

Rule 2 of *do*-calculus - Action/observation exchange

$$P(Y|do(T=t), do(X=x), W) = P(Y|do(T=t), X=x, W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\underline{X}}} X|T,W$$

Intuition:

- Removing all edges to T results in the interventional graph and:
  $$P(Y|T=t, do(X=x)), W) = P(Y|T=t, X=x, W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\underline{X}}} X|T,W$$
- If all backdoor paths from X to Y are blocked by T and W after removing the links between X and it's descendants, then conditioning on X = intervention on X

Generalization of backdoor criterion

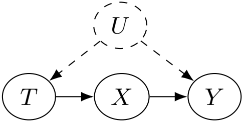
Rule 2 of *do*-calculus - Action/observation exchange

$$P(Y|do(T=t), do(X=x), W) = P(Y|do(T=t), X=x, W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\underline{X}}} X|T,W$$

Intuition:

- Removing all edges to T results in the interventional graph and:
  $$P(Y|T=t, do(X=x)), W) = P(Y|T=t, X=x, W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\underline{X}}} X|T,W$$
- If all backdoor paths from X to Y are blocked by T and W after removing the links between X and it's descendants, then conditioning on X = intervention on X

Generalization of backdoor criterion



$$P(Y|do(T=t), do(X=x)) =$$

Rule 2 of *do*-calculus - Action/observation exchange

$$P(Y|do(T=t), do(X=x), W) = P(Y|do(T=t), X=x, W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\underline{X}}} X|T,W$$

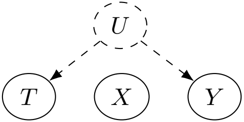
Intuition:

- Removing all edges to T results in the interventional graph and:

  $$P(Y|T=t, do(X=x)), W) = P(Y|T=t, X=x, W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\underline{X}}} X|T,W$$

- If all backdoor paths from X to Y are blocked by T and W after removing the links between X and it's descendants, then conditioning on X = intervention on X

Generalization of backdoor criterion



$$P(Y|do(T=t), do(X=x)) = P(Y|do(T=t), X=x) \;\; Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\underline{X}}} X$$

Rule 3 of *do*-calculus - Insertion/deletion of actions

Let $X = X_{\text{W-Anc}} \cup X_{\text{W-Rest}}$:

$$P(Y|do(T=t), do(X=x), W) = P(Y|do(T=t), W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T}, \overline{X_{\text{W-Rest}}}}} X|T, W$$

$X_{\text{W-Rest}}$ is the set of nodes in X that not ancestors of any node (e.g. descendants of some nodes) in set W in $\mathcal{G}_{\overline{T}}$.

Rule 3 of *do*-calculus - Insertion/deletion of actions

Let $X = X_{\text{W-Anc}} \cup X_{\text{W-Rest}}$:

$$P(Y|do(T=t), do(X=x), W) = P(Y|do(T=t), W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T}, \overline{X_{\text{W-Rest}}}}} X|T, W$$

$X_{\text{W-Rest}}$ is the set of nodes in X that not ancestors of any node (e.g. descendants of some nodes) in set W in $\mathcal{G}_{\overline{T}}$.

- Removing all edges to T results in the interventional graph and:

  $$P(Y|T=t, do(X=x), W) = P(Y|T=t, W) \quad \text{if} \quad Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{X_{\text{W-Rest}}}}} X|T, W$$

- We already know that $Y \perp\!\!\!\perp X_{\text{W-Anc}}|W$ (by definition),
- Now in $\mathcal{G}_{\overline{X_{\text{W-Rest}}}}$ we know that *if* there is a relationship between X and Y, it *must* be causal,
- Therefore the rule says that if $Y \perp\!\!\!\perp X|T, W$ in $\mathcal{G}_{\overline{X_{\text{W-Rest}}}}$ then interventions on $X_{\text{W-Rest}}$ can be freely inserted/deleted because we are guaranteed no causal paths and all non-causal paths are already blocked by W.

Rule 3 of *do*-calculus - Example

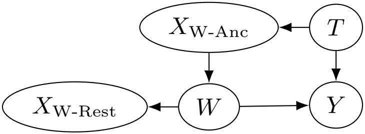


Figure: $\mathcal{G}$

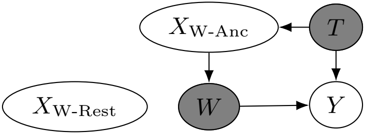


Figure: $\mathcal{G}_{\overline{T},\overline{X_{\text{W-Rest}}}}$

do-calculus is complete[1]

Theorem - Completeness of *do*-calculus

A causal effect $P(Y = y|do(T = t))$ is identifiable if and only if there exists a finite sequence of transformations, each conforming to one of the following inference rules that reduce $P(Y = y|do(T = t))$ into an expression involving observed quantities

1. Rule 1:

$$P(Y|do(T = t), X, W) = P(Y|do(T = t), W) \quad \text{if } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T}}} X|T, W$$

2. Rule 2:

$$P(Y|do(T = t), do(X = x), W) = P(Y|do(T = t), X = x, W)$$
$$\text{if } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\underline{X}}} X|T, W$$

3. Rule 3:

$$P(Y|do(T = t), do(X = x), W) = P(Y|do(T = t), W)$$
$$\text{if } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\overline{X_{\text{W-Rest}}}}} X|T, W$$

[1] Proof in Huang and Valtorta, 2012 and Shpitser and Pearl, 2012

Intuition for the rules of do-calculus

- Each rule first applies the intervention to the treatment resulting in $\mathcal{G}_{\overline{T}}$,
- Rule 1: Add/remove any variables that are d-separated in the interventional graph,
- Rule 2: We can replace conditioning with interventions whenever we are guaranteed that T, W block all backdoor paths,
- Rule 3: We can add/delete interventions over a set X as long as there are no direct causal paths between X and Y in the set of X that are non-ancestors of W (since W blocks the influence of the remaining set of X on Y).

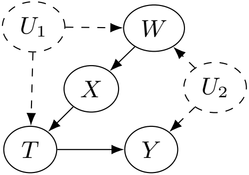
Questions?

Question

Any questions on do-calculus?

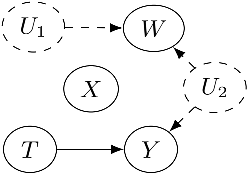
Example - Identification with *do*-calculus

$P(y|do(T=t))$



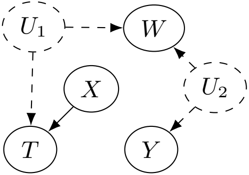
Example - Identification with *do*-calculus

$$
\begin{aligned}
&P(y|do(T=t))\\
&= P(y|do(T=t), do(X=x)) \quad \text{(Rule 3: insertion of actions - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\overline{X}}} X|T)
\end{aligned}
$$



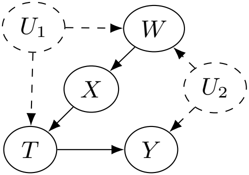
Example - Identification with *do*-calculus

$$\begin{aligned}
&P(y|do(T=t))\\
&= P(y|do(T=t), do(X=x)) \quad \text{(Rule 3: insertion of actions - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\overline{X}}} X|T)\\
&= P(y|T=t, do(X=x)) \quad \text{(Rule 2: action/observation exchange - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{X},\underline{T}}} T|X)
\end{aligned}$$



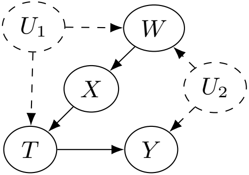
Example - Identification with *do*-calculus

$$
\begin{aligned}
&P(y|do(T=t)) \\
&= P(y|do(T=t), do(X=x)) \quad \text{(Rule 3: insertion of actions - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\overline{X}}} X|T) \\
&= P(y|T=t, do(X=x)) \quad \text{(Rule 2: action/observation exchange - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{X},\underline{T}}} T|X) \\
&= \frac{P(y,t|do(X=x))}{P(t|do(X=x))}
\end{aligned}
$$



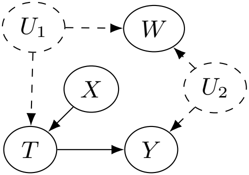
Example - Identification with *do*-calculus

$$
\begin{aligned}
&P(y|do(T=t))\\
&= P(y|do(T=t), do(X=x)) \quad \text{(Rule 3: insertion of actions - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\overline{X}}} X|T\text{)}\\
&= P(y|T=t, do(X=x)) \quad \text{(Rule 2: action/observation exchange - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{X},\underline{T}}} T|X\text{)}\\
&= \frac{P(y,t|do(X=x))}{P(t|do(X=x))}\\
&= \frac{\sum_w P(y,t|W=w, do(X=x))P(w|do(X=x))}{\sum_w P(t|W=w, do(X=x)P(w|do(X=x))} \quad \text{(Marginalization over } W\text{)}
\end{aligned}
$$



Example - Identification with *do*-calculus

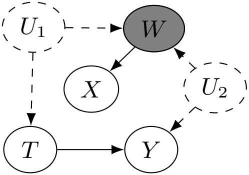
$$
\begin{aligned}
&P(y|do(T=t)) \\
&= P(y|do(T=t), do(X=x)) \quad \text{(Rule 3: insertion of actions - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\overline{X}}} X|T) \\
&= P(y|T=t, do(X=x)) \quad \text{(Rule 2: action/observation exchange - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{X},\underline{T}}} T|X) \\
&= \frac{P(y,t|do(X=x))}{P(t|do(X=x))} \\
&= \frac{\sum_w P(y,t|W=w, do(X=x))P(w|do(X=x))}{\sum_w P(t|W=w, do(X=x)P(w|do(X=x))} \quad \text{(Marginalization over } W) \\
&= \frac{\sum_w P(y,t|W=w, do(X=x))P(w)}{\sum_w P(t|W=w, do(X=x)P(w)} \quad \text{(Rule 3: deletion of actions - } W \perp\!\!\!\perp_{\mathcal{G}_{\overline{X}}} X)
\end{aligned}
$$



Example - Identification with *do*-calculus

$$
\begin{aligned}
&P(y|do(T=t))\\
&= P(y|do(T=t), do(X=x)) \quad \text{(Rule 3: insertion of actions - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{T},\overline{X}}} X|T\text{)}\\
&= P(y|T=t, do(X=x)) \quad \text{(Rule 2: action/observation exchange - } Y \perp\!\!\!\perp_{\mathcal{G}_{\overline{X},\underline{T}}} T|X\text{)}\\
&= \frac{P(y,t|do(X=x))}{P(t|do(X=x))}\\
&= \frac{\sum_w P(y,t|W=w, do(X=x))P(w|do(X=x))}{\sum_w P(t|W=w, do(X=x)P(w|do(X=x))} \quad \text{(Marginalization over } W\text{)}\\
&= \frac{\sum_w P(y,t|W=w, do(X=x))P(w)}{\sum_w P(t|W=w, do(X=x)P(w)} \quad \text{(Rule 3: deletion of actions - } W \perp\!\!\!\perp_{\mathcal{G}_{\overline{X}}} X\text{)}\\
&= \frac{\sum_w P(y,t|W=w, X=x)P(w)}{\sum_w P(t|W=w, X=x)P(w)}
\end{aligned}
$$

(Rule 2: action/observation exchange - $T, Y \perp\!\!\!\perp_{\mathcal{G}_{\underline{X}}} X|W$)



Questions?

Question

Any questions on do-calculus?

The story thus far

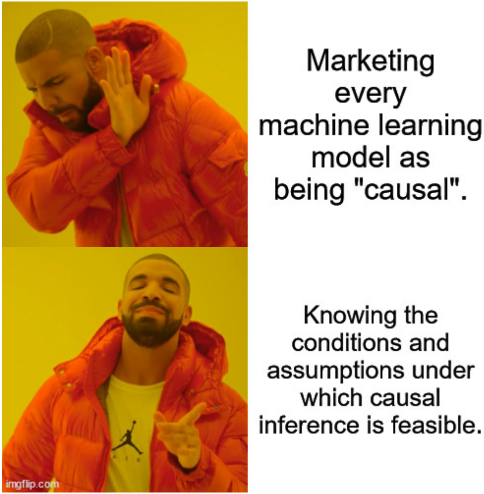
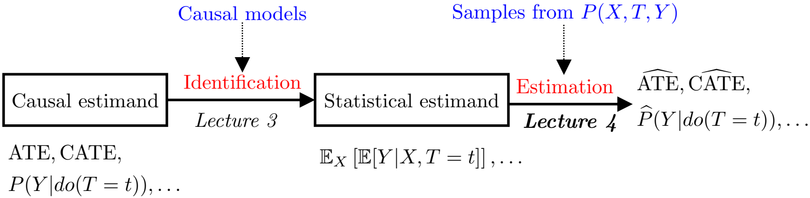

Marketing every machine learning model as being "causal".
Knowing the conditions and assumptions under which causal inference is feasible.
imgflip.com

Figure: On the feasibility of causal inference

Estimation

- Thus far we have studied how to map from causal quantities onto statistical estimands.
- We'll turn to *estimation* - how to map from *data* onto a statistical estimand.
- One of the areas where ideas from machine learning can play a big role in causal inference.



Causal models

Samples from $P(X, T, Y)$

Causal estimand

Identification

Lecture 3

Statistical estimand

Estimation

Lecture 4

$\widehat{\text{ATE}}, \widehat{\text{CATE}},$
$\widehat{P}(Y|do(T=t)), \ldots$

$\text{ATE}, \text{CATE},$
$P(Y|do(T=t)), \ldots$

$\mathbb{E}_X\left[\mathbb{E}[Y|X, T=t]\right], \ldots$

Estimation in supervised learning

Consider the following regression model:

- Data: $\boldsymbol{X} \in \mathbb{R}^{N \times D}; \boldsymbol{Y} \in \mathbb{R}^{N \times 1}$; x_i, y_i denote rows of each matrix.
- Model (trained): $f(x;\theta^*) = W^*x$, or $f(x;\theta^*) = W_2^*(\sigma(W_1^*x))$
- Estimating the risk of a regression model:
  - Estimand for risk: $\mathbb{E}[\mathcal{R}[(f(X,\theta^*),Y]]$; $\mathcal{R}(\hat{y},y) = \frac{1}{2}(y-\hat{y})^2$
  - Estimator: $\mathbb{E}[\mathcal{R}(f(X,\theta^*),Y)] = \frac{1}{N}\sum_{i=1}^{N}\mathcal{R}(f(x_i,\theta^*),y_i)$
- Conditional expectation of outcomes:
  - Estimand for conditional expectation: $\mathbb{E}[Y|X=x]$
  - Non-parametric estimator: $\mathbb{E}[Y|X=x] = \frac{1}{\sum_{j=1}^{N}\mathbb{I}[x_j=x]}\sum_{i=1}^{N}y_i\mathbb{I}[x_i=x]$
  - Parametric estimator: $\mathbb{E}[Y|X=x] = f(x,\theta^*)$

We can use a predictive model to get an estimate of a conditional expectation!

Estimation of the G-formula/Backdoor adjustment

Focus on estimation in the backdoor setting today! Assuming positivity/unconfoundedness/graphical criteria for identifiability we obtain the following estimands for Average Treatment Effects:

- Let X be the adjustment set/backdoor path in the causal Bayesian network.
- Potential outcomes / Backdoor adjustment: $\mathbb{E}[Y_1 - Y_0] = \mathbb{E}_X[\mathbb{E}[Y|T = 1, X] - \mathbb{E}[Y|T = 0, X]]$

Strategy: Use predictive models to approximate Estimand 1 and 2.

$$\mathbb{E}_W[\underbrace{\mathbb{E}[Y|T = 1, X]}_{\textit{Estimand 1}} - \underbrace{\mathbb{E}[Y|T = 0, X]}_{\textit{Estimand 2}}]$$

Using models to estimate the G-formula

The use of parameteric methods to estimate the effect of interventions goes by many names:

- G-computation estimators
- Parametric G-formula
- Standardization
- S-learner

Conditional outcome modeling

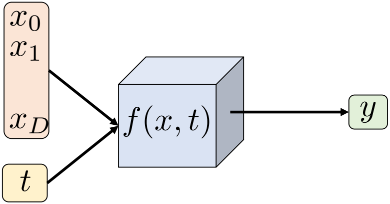


Figure: Using machine learning to fit conditional expectations

- $\mathcal{D} = \{(x_1, t_1, y_1), \ldots, (x_N, t_N, y_N), \ldots, (x_{N+\tilde{N}}, t_{N+\tilde{N}}, y_{N+\tilde{N}}\}$,
- Fit $f(x,t) \approx \mathbb{E}[Y|X,T]$ using $\{(x_N, t_N, y_N), \ldots, (x_{N+\tilde{N}}, t_{N+\tilde{N}}, y_{N+\tilde{N}})\}$,
- $\widehat{\text{CATE}}(x) = f(x,1) - f(x,0)$,
- $\widehat{\text{ATE}} = \frac{1}{N} \sum_{i=1}^{N} f(x_i, 1) - f(x_i, 0)$

Grouped conditional outcome modeling

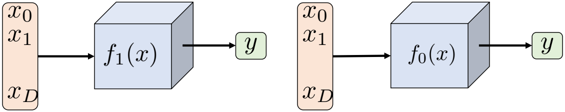
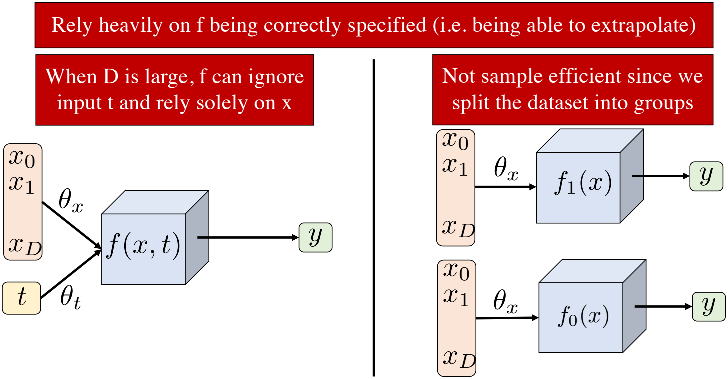


Figure: Using machine learning to fit grouped conditional expectations

- Let $\mathcal{D}_{tr} = \{(x_N, t_N, y_N), \ldots, (x_{N+\tilde{N}}, t_{N+\tilde{N}}, y_{N+\tilde{N}})\} = \mathcal{D}_1 \cup \mathcal{D}_0$,
- $\mathcal{D}_1 = \{(x_1, 1, y_1), \ldots, (x_k, 1, y_k)\}$ & $\mathcal{D}_0 = \{(x'_1, 0, y'_1), \ldots, (x'_{k'}, 0, y'_{k'})\}$,
- Fit $f_1(x) \approx \mathbb{E}[Y|X]$ using $\mathcal{D}_1$ and $f_0(x) \approx \mathbb{E}[Y|X]$ using $\mathcal{D}_0$,
- $\widehat{\text{CATE}}(x) = f_1(x) - f_0(x)$,
- $\widehat{\text{ATE}} = \frac{1}{N}\sum_{i=1}^{N} f_1(x_i) - f_0(x_i)$

Tradeoffs in the parametric G-formula



Rely heavily on f being correctly specified (i.e. being able to extrapolate)

When D is large, f can ignore input t and rely solely on x

Not sample efficient since we split the dataset into groups

Figure: Tradeoffs in estimation

Covariate adjustment with linear models

- Lets assume that we model conditional expectations with linear models,
- Then $Y_t(x) = f(x,t) = \beta x + \gamma t + \epsilon_t, \;\; \mathbb{E}[\epsilon_t] = 0$,
- We can write out a closed form solution for CATE as follows:

$$\begin{aligned}\text{CATE}(x) &= \mathbb{E}[(\beta x + \gamma + \epsilon_1) - (\beta x + \epsilon_0)] \\ &= \mathbb{E}[\cancel{\beta x} + \gamma - \cancel{\beta x}] + \underbrace{\mathbb{E}[\epsilon_1] - \mathbb{E}[\epsilon_0]}_{0} \\ &= \gamma\end{aligned}$$

- $\text{ATE} = \mathbb{E}_x[\text{CATE}(x)] = \gamma$

1. Takeaway 1: Goal in causal inference is to estimate γ well! f is a tool to get us there.
2. Takeaway 2: Often β (coefficients of adjustment set) are referred to as *nuisance parameters*.

Cost of model mis-specification

Consider the following *true* data generating process:

- $Y_t(x) = f^*(x,t) = \beta x + \gamma t + \delta x^2 + \epsilon_t, \ \ \mathbb{E}[\epsilon_t] = 0,$
- $ATE = \gamma$

Now, lets say we estimate the following *hypothesized* predictive model:

- $\hat{Y}_t(x) = \hat{\beta} x + \hat{\gamma} t,$
- $\hat{\gamma} = \gamma + \delta \frac{\mathbb{E}[xt]\mathbb{E}[x^2] - \mathbb{E}[t^2]\mathbb{E}[x^2 t]}{\mathbb{E}[xt^2] - \mathbb{E}[x^2]\mathbb{E}[t^2]}$

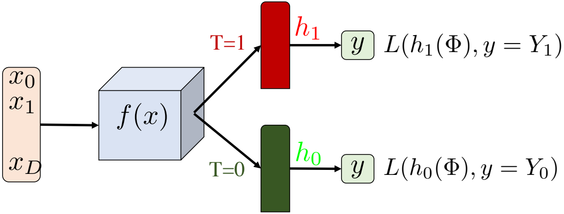
Mis-specification can result in bias: δ can result in an arbitrarily large bias in our causal estimate!

Slide credit to David Sontag

Non-linear functions

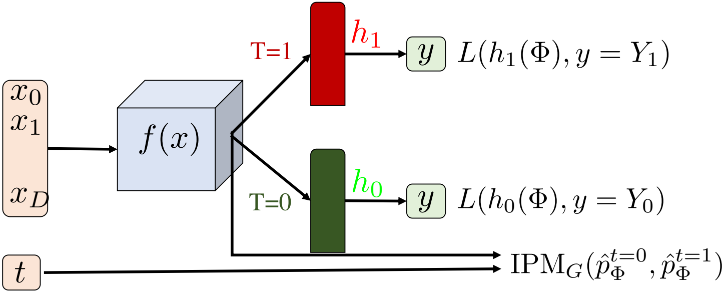
- Nonlinear functions have a rich history of being used in conditional outcome modeling in statistics and machine learning:
- Random forests and Bayesian Trees (J. L. Hill, 2011; J. Hill, Linero, and Murray, 2020),
- Gaussian processes (Alaa and Van Der Schaar, 2017; Schulam and Saria, 2017),
- Neural Networks (Johansson, Shalit, and Sontag, 2016),

TAR-Net (Johansson, Shalit, and Sontag, 2016)



- Grouped conditional outcome model is inefficient → TAR-Net uses a neural network $f(x)$ to learn a shared low-dimensional representation of high-dimensional data x for both treatment and control,
- Treatment head and control head are responsible for modeling outcomes under different treatment assignments.
- In finite samples, what happens when treatment assignment is predictive of outcome? → Model's representation can rely solely on predicting treatment assignment i.e. it learns $f(x) = [f_1(x), f_0(x)]$.

TAR-Net (Johansson, Shalit, and Sontag, 2016)



- Additional regularization penalty using an integral probability metric to ensure that the representation space $h(x)$ is *aligned* for both treatment and control groups.

Questions?

Question

Any questions on parametric estimation?

Recap - Lecture 4

- Identification
  - Backdoor criteria: Identical to adjustment via the G-formula,
  - Frontdoor criteria: Using mediators to identify causal effect on outcomes.
- Do-Calculus: Three rules to identify causal effects:
  1. Insertion or deletion of observations : Generalization of d-separation,
  2. Interchanging actions with observations : Generalization of the backdoor criteria,
  3. Insertion or deletion of actions
- Parametric Estimation:
  - Conditional outcome models
  - Grouped conditional outcome models
  - TAR-Net

- Zheng, Xun et al. (2018). “Dags with no tears: Continuous optimization for structure learning”. In: *Advances in Neural Information Processing Systems* 31.
- Lachapelle, Sébastien et al. (2019). “Gradient-based neural dag learning”. In: *arXiv preprint arXiv:1906.02226.*
- Yu, Yue et al. (2021). “DAGs with no curl: An efficient DAG structure learning approach”. In: *International Conference on Machine Learning.* PMLR, pp. 12156–12166.
- Jung, Yonghan, Jin Tian, and Elias Bareinboim (2021). “Estimating identifiable causal effects through double machine learning”. In: *Proceedings of the AAAI Conference on Artificial Intelligence.* Vol. 35. 13, pp. 12113–12122.
- Huang, Yimin and Marco Valtorta (2012). “Pearl’s calculus of intervention is complete”. In: *arXiv preprint arXiv:1206.6831.*
- Shpitser, Ilya and Judea Pearl (2012). “Identification of conditional interventional distributions”. In: *arXiv preprint arXiv:1206.6876.*
- Hill, Jennifer L (2011). “Bayesian nonparametric modeling for causal inference”. In: *Journal of Computational and Graphical Statistics* 20.1, pp. 217–240.

Hill, Jennifer, Antonio Linero, and Jared Murray (2020). "Bayesian additive regression trees: A review and look forward". In: *Annual Review of Statistics and Its Application* 7.1.

Alaa, Ahmed M and Mihaela Van Der Schaar (2017). "Bayesian inference of individualized treatment effects using multi-task gaussian processes". In: *Advances in neural information processing systems* 30.

Schulam, Peter and Suchi Saria (2017). "What-if reasoning using counterfactual gaussian processes". In: *NIPS.*

Johansson, Fredrik, Uri Shalit, and David Sontag (2016). "Learning representations for counterfactual inference". In: *International conference on machine learning.* PMLR, pp. 3020–3029.