Causal Inference
CS 477-677

Causal Decision Theory

Ilya Shpitser

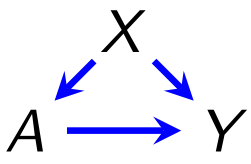
Outline

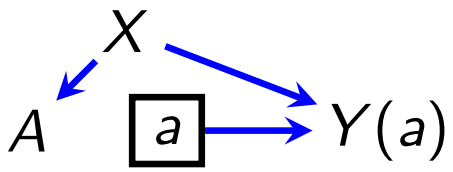
Review

- Causal models of a (hidden variable) DAG.
- Identification (the ID algorithm).
- Mediation (splitting causal effects).
- Estimation methods – IPW, parametric g-formula.
- Counterfactual experiment contrasts.
- Today: learning policies.

Conditional Ignorability (Review)

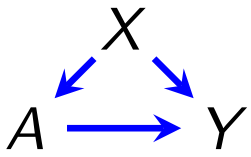
- Treatment A (usually binary, but not necessary).
- Outcome Y (discrete or continuous).
- A vector of baseline factors $\vec{X}$. Picture (observed and counterfactual):

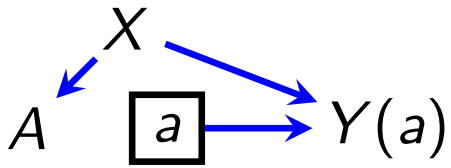




Conditional Ignorability (Review)

- Treatment A (usually binary, but not necessary).
- Outcome Y (discrete or continuous).
- A vector of baseline factors $\vec{X}$. Picture (observed and counterfactual):



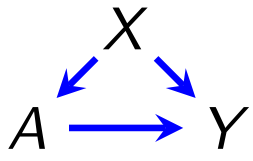


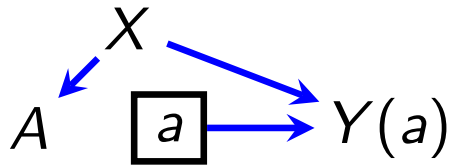
- Predicting what will happen to Y if $A = a$:

$$p(Y(a)) = \sum_{\vec{x}} p(Y \mid A = a, \vec{X} = \vec{x}) p(\vec{X} = \vec{x}).$$

Conditional Ignorability (Review)

- Treatment A (usually binary, but not necessary).
- Outcome Y (discrete or continuous).
- A vector of baseline factors $\vec{X}$. Picture (observed and counterfactual):





- Predicting what will happen to Y if $A = a$:

$$p(Y(a)) = \sum_{\vec{x}} p(Y \mid A = a, \vec{X} = \vec{x}) p(\vec{X} = \vec{x}).$$

- Average causal effect (ACE):

$$E[Y(1)] - E[Y(0)] = \sum_{\vec{x}} \left\{ E[Y \mid A = 1, \vec{X} = \vec{x}] - E[Y \mid A = 0, \vec{X} = \vec{x}] \right\} p(\vec{x}).$$

Using Causality To Act

- Say we learn $E[Y(1)] > E[Y(0)]$.
- On average, $A = 1$ helps (is "causally effective").
- A new patient comes in, should we set $A = 1$?

Using Causality To Act

- Say we learn $E[Y(1)] > E[Y(0)]$.
- On average, $A = 1$ helps (is "causally effective").
- A new patient comes in, should we set $A = 1$?
- Not necesarily, a patient has particular characteristics $\vec{X} = \vec{x}$.
- May well be $E[Y(1) \mid \vec{x}] < E[Y(0) \mid \vec{x}]$ (Simpson's reversal).

Using Causality To Act

- Say we learn $E[Y(1)] > E[Y(0)]$.
- On average, $A = 1$ helps (is "causally effective").
- A new patient comes in, should we set $A = 1$?
- Not necesarily, a patient has particular characteristics $\vec{X} = \vec{x}$.
- May well be $E[Y(1) \mid \vec{x}] < E[Y(0) \mid \vec{x}]$ (Simpson's reversal).
- Want to pick a "good" mapping (policy) $f_A(\vec{X})$ from patient state $\vec{X} = \vec{x}$ to action $A = a$.
- In medical contexts, related to "personalized medicine."
- How do we formalize the problem?

Policies Under Conditional Ignorability

- Given: $p(Y, A, \vec{X})$, a policy class $\vec{F}_A$.
- Assume $Y(a) \perp\!\!\!\perp A \mid \vec{X}$.
- Find $f_A(\vec{X}) \in \vec{F}_A$ to optimize:

$$E[Y(A = f_A(\vec{X}))] \equiv \sum_{\vec{x}} E[Y(A = f_A(\vec{X} = \vec{x}))]p(\vec{X} = \vec{x})$$

- Reads as: "expected outcome had we counterfactually set A according to f_A, with expectation taken over $p(\vec{X})$."
- In other words, find

$$\arg\max_{f_A} E[Y(A = f_A(\vec{X}))].$$

Policies Under Conditional Ignorability

- Given: $p(Y, A, \vec{X})$, a policy class $\vec{F}_A$.
- Assume $Y(a) \perp\!\!\!\perp A \mid \vec{X}$.
- Find $f_A(\vec{X}) \in \vec{F}_A$ to optimize:

$$E[Y(A = f_A(\vec{X}))] \equiv \sum_{\vec{x}} E[Y(A = f_A(\vec{X} = \vec{x}))]p(\vec{X} = \vec{x})$$

- Reads as: "expected outcome had we counterfactually set A according to f_A, with expectation taken over $p(\vec{X})$."
- In other words, find

$$\arg\max_{f_A} E[Y(A = f_A(\vec{X}))].$$

- A is no longer a constant, but a function of $\vec{X}$!

Solving The Problem

- Steps for ACE:
  - Identify counterfactual distribution as function of observed data distribution.
  - Pick estimator (IPW, g-formula, 2SLS).
  - Pick statistical model (regression, Bayesian non-parametrics, neural nets, etc.)
  - Fit pieces by MLE.
  - Combine in the "right way" for ACE estimate.
  - Report intervals with bootstrap or posterior quantiles.
- Will do most of these steps for policies also.

Solving The Problem

- Steps for ACE:
  - Identify counterfactual distribution as function of observed data distribution.
  - Pick estimator (IPW, g-formula, 2SLS).
  - Pick statistical model (regression, Bayesian non-parametrics, neural nets, etc.)
  - Fit pieces by MLE.
  - Combine in the "right way" for ACE estimate.
  - Report intervals with bootstrap or posterior quantiles.
- Will do most of these steps for policies also.
- Three complications:
  - Identification is harder.
  - Have to search policy space $\vec{F}_A$.
  - More reliant on statistical model.

Identification

- Assume we could identify $p(Y(a) \mid \vec{X}(a)) = p(Y(a) \mid \vec{X})$.
- $E[Y(A = f_A(\vec{X}))]$ is a function of this and $p(\vec{X})$.
- $p(\vec{X})$ is always identified.
- Identifying $p(Y(a))$ is not enough! We need to consult $\vec{X}$ to determine A.

Identification (Under Conditional Ignorability)

- Since $Y(a) \perp\!\!\!\perp A \mid \vec{X}$,

$$p(Y(a) \mid \vec{X}) = p(Y(a) \mid A = a, \vec{X}) = p(Y \mid A = a, \vec{X}).$$

- Thus, for any f_A,

$$E[Y(A = f_A(\vec{X}))] = \left(\sum_{\vec{x}} E[Y(A = f_A(\vec{x})) \mid \vec{X} = \vec{x}] p(\vec{X} = \vec{x}) \right)$$
$$= \left(\sum_{\vec{x}} E[Y \mid A = f_A(\vec{x}), \vec{X} = \vec{x}] p(\vec{X} = \vec{x}) \right)$$

- f_A is known, other terms are functions of the observed data distribution.

Estimation (Parametric g-formula)

Given n data points on $A, Y, \vec{X}$, and assuming conditional ignorablity and consistency:

1. Posit statistical model for $E[Y \mid A, \vec{X}; \alpha]$.
2. Fit model by MLE, yielding $\hat{\alpha}$.
3. For each $f_A \in \vec{F}_A$, estimate $E[Y(A = f_A(\vec{X}))]$ by:

$$\frac{1}{n}\left(\sum_i E[Y \mid A = f_A(\vec{x}_i), \vec{x}_i; \hat{\alpha}]\right)$$

4. Report best f_A found.

Estimation (IPW)

Given n data points on $A, Y, \vec{X}$, and assuming conditional ignorablity and consistency:

1. Posit statistical model for $p[A \mid \vec{X}; \alpha]$.
2. Fit model by MLE, yielding $\hat{\alpha}$.
3. For each $f_A \in \vec{F}_A$, estimate $E[Y(A = f_A(\vec{X}))]$ by:

$$\frac{1}{n}\left(\sum_i Y_i \frac{\mathbb{I}(A_i = f_A(\vec{x}_i))}{p(A_i = f_A(\vec{x}_i) \mid \vec{x}_i; \hat{\alpha})}\right)$$

4. Report best f_A found.

Pros and Cons

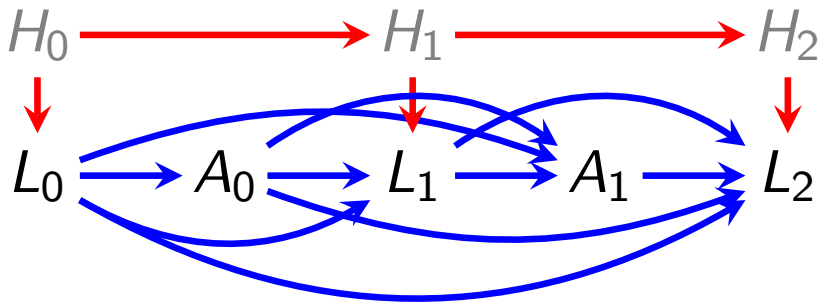
- Parametric g-formula is optimal if we know $E[Y \mid A, \vec{X}]$ (usual reasons).
- If we don't know $E[Y \mid A, \vec{X}]$, can get bad policy due to bias.
- IPW does not model Y, but high variance in policy estimate.

Pros and Cons

- Parametric g-formula is optimal if we know $E[Y \mid A, \vec{X}]$ (usual reasons).
- If we don't know $E[Y \mid A, \vec{X}]$, can get bad policy due to bias.
- IPW does not model Y, but high variance in policy estimate.
- Usually pick $\vec{F}_A$ to be small or easy to search.
- Example: when to switch from first line a to second line a' treatment based on unknown threshold $X \geq \alpha$.

Multiple Treatments (Review)

- Given the model



- Under sequential ignorability, we get:

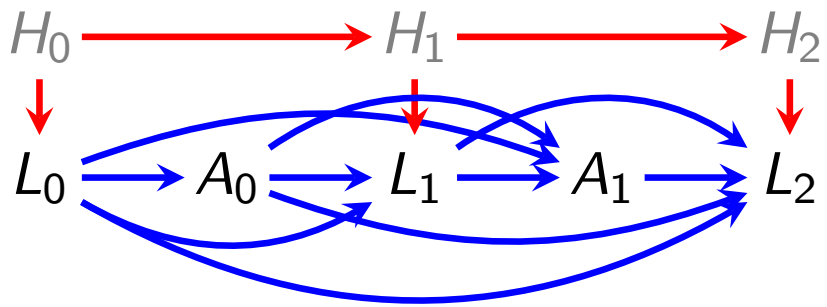
$$p(L_2(a_0, a_1)) = \sum_{l_0, l_1} p(L_2 \mid a_1, l_1, a_0, l_0) p(l_1 \mid a_0, l_0) p(l_0).$$

- Can get contrasts $E[L_2(a_0, a_1)] - E[L_2(a_0', a_1')]$ from this.
- How do we generalize evaluating policies to this setting?

Multistage policies

- Under model below, we want to pick $f_{A_0}(L_0)$, $f_{A_1}(L_1, A_0, L_0)$ to optimize

$$E[Y(A_1 = f_{A_1}(L_1(L_0, A_0 = f_{A_0}(L_0)), A_0 = f_{A_0}(L_0), L_0), A_0 = f_{A_0}(L_0))]$$



- Expectation taken with respect to $p(L_0, L_1(A_0 = f_{A_0}(L_0)))$.

General Definition

- For $i = 1, \ldots, k$, will define $\vec{F}_{A_{<i}}(\vec{L}_{A_{<i-1}})$, $\vec{L}_{A_{<i}}$ recursively:

$$\begin{aligned}
\vec{L}_{A_{<0}} &\equiv \{L_0\} \\
\vec{F}_{A_{<1}}(\vec{L}_{A_{<0}}) &\equiv \{f_{A_0}(L_0)\} \\
\vec{L}_{A_{<i}} &\equiv L_i(A_{<i} = \vec{F}_{A_{<i}}(\vec{L}_{A_{<(i-1)}}), \vec{L}_{A_{<(i-1)}}) \cup \vec{L}_{A_{<(i-1)}} \\
\vec{F}_{A_{<i}}(\vec{L}_{A_{<(i-1)}}) &\equiv f_{A_{i-1}}(\vec{L}_{A_{<i}}) \cup \vec{F}_{A_{<(i-1)}}(\vec{L}_{A_{<(i-2)}})
\end{aligned}$$

- Example:

$$\begin{aligned}
\vec{F}_{A_{<2}} &\equiv \{f_{A_0}(L_0), f_{A_1}(L_0, A_0 = f_{A_0}(L_0), L_1(A_0 = f_{A_0}(L_0), L_0))\} \\
\vec{L}_{A_{<1}} &\equiv \{L_0, L_1(A_0 = f_{A_0}(L_0), L_0)\}
\end{aligned}$$

- Like recursive substitution, but we use f_{A_i} instead of setting A_i to a constant.
- How do we optimize $\vec{F}_{A_{<k}}$ with respect to $E[L_k(A_{<k} = \vec{F}_{A_{<k}})]$?

Steps For Multistage Policy Problems

- Identify counterfactual distribution. In this case

$$\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a})).$$

- Pick estimator (marginal structural models, sequential parametric g-formula, Q-learning).
- Pick statistical models (regressions, etc.)
- Fit by MLE.
- Combine in the "right way."
- Report best policy set found.

Identification

- If we can identify $\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a}))$, we can identify $E[L_k(\vec{A} = \vec{F}_{\vec{A}})]$ (a function of above and $\vec{F}_{\vec{A}}$).
- When is $\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a}))$ identifiable?

Identification

- If we can identify $\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a}))$, we can identify $E[L_k(\vec{A} = \vec{F}_{\vec{A}})]$ (a function of above and $\vec{F}_{\vec{A}}$).
- When is $\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a}))$ identifiable?
- Under sequential ignorability (remember we could always do this, we just summed out $L_{<k}$ before).
- Identifying formula:

$$\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a})) = \left(\prod_{i=0}^{k} p(L_i \mid L_{<i}, a_{<i}) \right)$$

- $\vec{a} \equiv (a_k, a_{<k}) \equiv (a_k, a_{k-1}, a_{<k-1}) \equiv \ldots \equiv (a_k, a_{k-1}, \ldots, a_1)$.

Identification

- If we can identify $\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a}))$, we can identify $E[L_k(\vec{A} = \vec{F}_{\vec{A}})]$ (a function of above and $\vec{F}_{\vec{A}}$).
- When is $\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a}))$ identifiable?
- Under sequential ignorability (remember we could always do this, we just summed out $L_{<k}$ before).
- Identifying formula:

$$\prod_{i=0}^{k} p(L_i(\vec{a}) \mid L_{<i}(\vec{a})) = \left(\prod_{i=0}^{k} p(L_i \mid L_{<i}, a_{<i}) \right)$$

- $\vec{a} \equiv (a_k, a_{<k}) \equiv (a_k, a_{k-1}, a_{<k-1}) \equiv \ldots \equiv (a_k, a_{k-1}, \ldots, a_1)$.
- We get $E[L_k(A_{<k} = \vec{F}_{A_{<k}})]$ in stages.

Learning Multi-Stage Policies (Last Stage)

- Assume we already chose $A_0, \ldots, A_{k-1}$, and must choose A_k, for a given setting $l_{<(k+1)}, a_{<k}$.
- Reduces to conditional ignorable case.
- Optimize:

$$\arg\max_{f_{A_k}} E[L_k(A_k = f_A(l_{<k}, a_{<k})) \mid l_{<(k+1)}, a_{<k}] =$$

$$\arg\max_{f_{A_k}} E[L_k \mid A_k = f_{A_k}(l_{<k}, a_{<k}), l_{<(k+1)}, a_{<k}].$$

- Simply pick the best (in expectation) a_k for each history $l_{<(k+1)}, a_{<k}$.

Learning Multi-Stage Policies (Recursive Stage)

- Assume we know $f_{A_{m+1}}, \ldots, f_{A_k}$, $1 < m+1 < k$, and already chose $A_0, \ldots, A_{m-1}$.
- Must choose A_m, for a given setting $l_{<(m+1)}, a_{<m}$.
- Optimize:

$$\arg\max_{f_{A_m}} E[L_k(A_m = f_{A_m}(l_{<(m+1)}, a_{<m})) \mid l_{<(m+1)}, a_{<m}] =$$

$$\arg\max_{f_{A_m}} E[L_k \mid A_m = f_{A_m}(l_{<(m+1)}, a_{<m}), l_{<(m+1)}, a_{<m}].$$

- Take expectation with respect to future (after m).
- Already know all optimal future policies.
- This is called backwards induction or dynamic programming.
- Related to Q-learning in reinforcement learning.
- How to estimate?

Q-Learning Example

- Expectations we are optimizing at each stage are called "Q-functions" in reinforcement learning.
- If we have two treatment A_0, A_1, then:

$$Q_2^{opt}(l_1, a_1, l_0, a_0) = E[L_2 \mid A_1 = a_1, l_0, a_0, l_1]$$
$$Q_1^{opt}(l_0, a_0) = E[\max_{a_1} Q_2^{opt}(L_1, l_0, a_0, a_1) \mid l_0, a_0]$$

- Can use any expectation model, simple one is linear regression.

Q-Learning Example (2SLS)

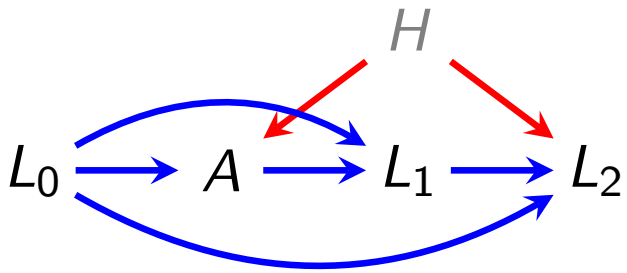
- Fit $Q_2^{opt}(l_1, a_1, l_0, a_0; \alpha) = E[L_2 \mid A_1 = a_1, l_0, a_0, l_1; \alpha]$ by MLE, yielding $\hat{\alpha}$.
- This is a "stage 2" regression.
- For data row j, replace L_1^j by $\hat{L}_1^j \equiv E[\max_{a_1} Q_2^{opt}(L_1, l_0^j, a_0^j, a_1; \hat{\alpha}) \mid l_0^j, a_0^j]$.
- Fit $E[\hat{L}_1 \mid A_0, L_0; \beta]$ by MLE, yielding $\hat{\beta}$.
- This is a "stage 1" regression.
- Pick A_0 to optimize.
- Easy to generalize to any number of stages, models (see reading).

Relationship to Reinforcement Learning

- f_{A_i} are sometimes called dynamic treatment regimes, or policies.
- Lots of overlap with the reinforcement learning literature.
- Reinforcement learning:
  - Emphasis on online learning (agents acting in the world).
  - Can generate lots of own data (games, robotics).
  - No bias issues, large sample sizes.
  - Complex models (deep learning, etc.)
  - Very impressive real world results!
- Causal inference:
  - Typically offline learning (patients data under suboptimal policy).
  - Cannot generate new data.
  - Confounding bias, small sample sizes.
  - Parametric models, counterfactual reasoning.

Policies Without Conditional Ignorability

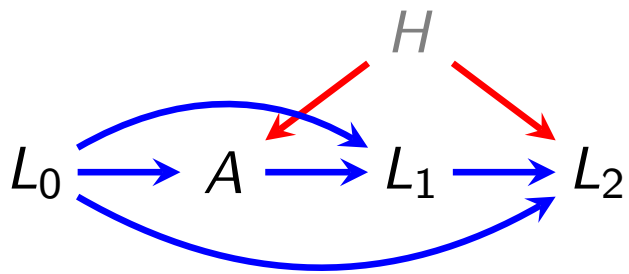
- Recall: front-door (this time with baseline L_0):



$$p(L_2(a)) = \sum_{l_1, l_0} \left(\sum_{a'} p(L_2 \mid l_1, a', l_0) p(a' \mid l_0) \right) p(L_1 = l_1 \mid a, l_0) p(l_0)$$

Policies Without Conditional Ignorability

- Recall: front-door (this time with baseline L_0):



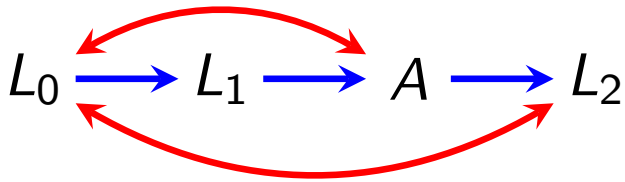
$$p(L_2(a)) = \sum_{l_1, l_0} \left(\sum_{a'} p(L_2 \mid l_1, a', l_0) p(a' \mid l_0) \right) p(L_1 = l_1 \mid a, l_0) p(l_0)$$

- Can pick $f_A(L_0)$ to maximize $E[L_2(A = f_A(L_0))]$,

$$= \sum_{l_1, l_0} \left(\sum_{a'} E[L_2 \mid l_1, a', l_0] p(a' \mid l_0) \right) p(L_1 = l_1 \mid A = f_A(l_0), l_0) p(l_0).$$

Identification Issues With Policy Learning

- Sometimes $p(L_2(a))$ is identified, but $p(L_2(a) \mid L_0)$ is not:



- By **ID** algorithm,

$$p(L_2(a)) = \frac{\sum_{L_0} p(L_2, a \mid L_1, L_0) p(L_0)}{\sum_{L_0} p(a \mid L_1, L_0) p(L_0)}$$

- But $p(L_2(a) \mid L_0)$ is not (verify!)
- So: can compute ACE, but cannot optimize $f_A(L_1, L_0)$ here without more assumptions.

Next time: Learning Causal Structure From Data.