Quant II

Lab 12: Machine Learning and Generalization

Junlong Aaron Zhou

May 07, 2021

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.
- How could machine learning be applied to social sciences?

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.
- How could machine learning be applied to social sciences?
- Variable creation: train a model and use it to code data

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.
- How could machine learning be applied to social sciences?
- Variable creation: train a model and use it to code data
- Predicting nuisance parameters

Machine learning in social sciences

- Remember that machine learning refers to the set of algorithms that minimize the prediction error of a model $y = f(x)$.
- Basic elements: sampling splitting, penalty, cross-validation, bias-variance tradeoff, etc.
- Basic algorithms: LASSO, Ridge, SVM, Tree, Forest, etc.
- How could machine learning be applied to social sciences?
- Variable creation: train a model and use it to code data
- Predicting nuisance parameters
- Estimating heterogeneous treatment effects and generate the optimal assignment

Variable creation

- The problem: transform texts, images, audio, or even video into variables

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)
- Example III: use images to detect protests (Zhang and Pan, 2019; Donghyeon et al., 2019)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)
- Example III: use images to detect protests (Zhang and Pan, 2019; Donghyeon et al., 2019)
- Example IV: audio and video processing (Knox and Lucas, 2018)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)
- Example III: use images to detect protests (Zhang and Pan, 2019; Donghyeon et al., 2019)
- Example IV: audio and video processing (Knox and Lucas, 2018)
- Example V: identify Russian bots on Twitter (Stukal et al., 2017)

Variable creation

- The problem: transform texts, images, audio, or even video into variables
- A common solution: 1. hire workers to code part of it, 2. train a model, 3. let the machine do the rest
- Example I: forest on Google maps (Burgess et al., 2012)
- Example II: topics of Chinese social media posts (King et al., 2013, 2014)
- Example III: use images to detect protests (Zhang and Pan, 2019; Donghyeon et al., 2019)
- Example IV: audio and video processing (Knox and Lucas, 2018)
- Example V: identify Russian bots on Twitter (Stukal et al., 2017)
- What if it is too expensive? Active learning (Miller et al., 2019)

Methods: predicting nuisance parameters

- Some relationships in causal inference can be non-causal.
- We just need to fit/predict it with a high accuracy.
  - Example I: Propensity score
  - Example II: First stage of IV
  - Example III: Response surface (what covariates to control for)
- These are "nuisance parameters" that have no causal interpretation.

Methods: predicting nuisance parameters

- Directly applying machine learning algorithms to estimate nuisance parameters leads to severe bias in finite sample.

Methods: predicting nuisance parameters

- Directly applying machine learning algorithms to estimate nuisance parameters leads to severe bias in finite sample.
- Cattaneo et al. (2018): two-stage estimation with too many covariates in the first stage is biased.
- Belloni et al. (2013) show that we need run "double selection" to obtain satisfying results.
- One model for the outcome, and the other for the treatment.

Methods: predicting nuisance parameters

- Directly applying machine learning algorithms to estimate nuisance parameters leads to severe bias in finite sample.
- Cattaneo et al. (2018): two-stage estimation with too many covariates in the first stage is biased.
- Belloni et al. (2013) show that we need run "double selection" to obtain satisfying results.
- One model for the outcome, and the other for the treatment.
- But why?

Double machine learning

- Let's consider the following DGP:

$$Y_i = \theta D_i + g_0(\mathbf{X}_i) + U_i$$

$$D_i = m_0(\mathbf{X}_i) + V_i$$

- We have $D_i \perp \{Y_i(1), Y_i(0)\}|\mathbf{X}_i$.

Double machine learning

- The classic model-based approach will find an estimate $\hat{g}$ for g_0.
- Then,

$$\hat{\theta} = \frac{\sum_{i=1}^{N} D_i(Y_i - \hat{g}(\mathbf{X}_i))}{N_1} - \frac{\sum_{i=1}^{N}(1 - D_i)(Y_i - \hat{g}(\mathbf{X}_i))}{N_0}$$

Double machine learning

- The classic model-based approach will find an estimate $\hat{g}$ for g_0.
- Then,

$$\hat{\theta} = \frac{\sum_{i=1}^{N} D_i(Y_i - \hat{g}(\mathbf{X}_i))}{N_1} - \frac{\sum_{i=1}^{N}(1 - D_i)(Y_i - \hat{g}(\mathbf{X}_i))}{N_0}$$

- This is "single selection."

Double machine learning

- Nevertheless, we can show that:

$$
\begin{aligned}
&\sqrt{N}(\hat{\theta}-\theta) \\
=&\sqrt{N}\left[\frac{\sum_{i=1}^{N} D_i U_i}{N_1} - \frac{\sum_{i=1}^{N}(1-D_i)U_i}{N_0}\right] \\
+&\sqrt{N}\left[\frac{\sum_{i=1}^{N} D_i(g_0(\mathbf{X}_i)-\hat{g}(\mathbf{X}_i))}{N_1} - \frac{\sum_{i=1}^{N}(1-D_i)(g_0(\mathbf{X}_i)-\hat{g}(\mathbf{X}_i))}{N_0}\right]
\end{aligned}
$$

- The first part is just the Hajek estimator, which converges to $N(0, I)$.
- But the second part may diverge to infinity as the convergence of $\hat{g}$ to g_0 is often slow.

Double machine learning

- Denote $E[Y_i|\mathbf{X}_i] = m_0(\mathbf{X}_i)\theta + g_0(\mathbf{X}_i)$ as $l_0(\mathbf{X}_i)$.
- We use machine learning to estimate $m_0(\mathbf{X}_i)$ and $l_0(\mathbf{X}_i)$.
- Then, we take the residual: $\hat{V}_i = D_i - \hat{m}(\mathbf{X}_i)$ and $\hat{W}_i = Y_i - \hat{l}(\mathbf{X}_i)$.
- Finally, $\hat{\theta}$ is estimated by regressing $\hat{W}_i$ on $\hat{V}_i$.

Double machine learning

- Denote $E[Y_i|\mathbf{X}_i] = m_0(\mathbf{X}_i)\theta + g_0(\mathbf{X}_i)$ as $l_0(\mathbf{X}_i)$.
- We use machine learning to estimate $m_0(\mathbf{X}_i)$ and $l_0(\mathbf{X}_i)$.
- Then, we take the residual: $\hat{V}_i = D_i - \hat{m}(\mathbf{X}_i)$ and $\hat{W}_i = Y_i - \hat{l}(\mathbf{X}_i)$.
- Finally, $\hat{\theta}$ is estimated by regressing $\hat{W}_i$ on $\hat{V}_i$.
- Intuitively, the second part of the bias is now decided by $(\hat{m}(\mathbf{X}_i) - m_0(\mathbf{X}_i))(\hat{l}(\mathbf{X}_i) - l_0(\mathbf{X}_i))$ plus $V_i(\hat{g}(\mathbf{X}_i) - g_0(\mathbf{X}_i))$.
- Even when each estimator converges to the true value slowly, their product may have a satisfying convergence rate.

Double machine learning

- This is called "Robinson's Transformation" (Robinson, 1988).
- The transformation allows us to achieve "Neyman orthogonality," meaning the bias from estimating nuisance parameters have negligible influence on the estimation of causal parameters.

Double machine learning

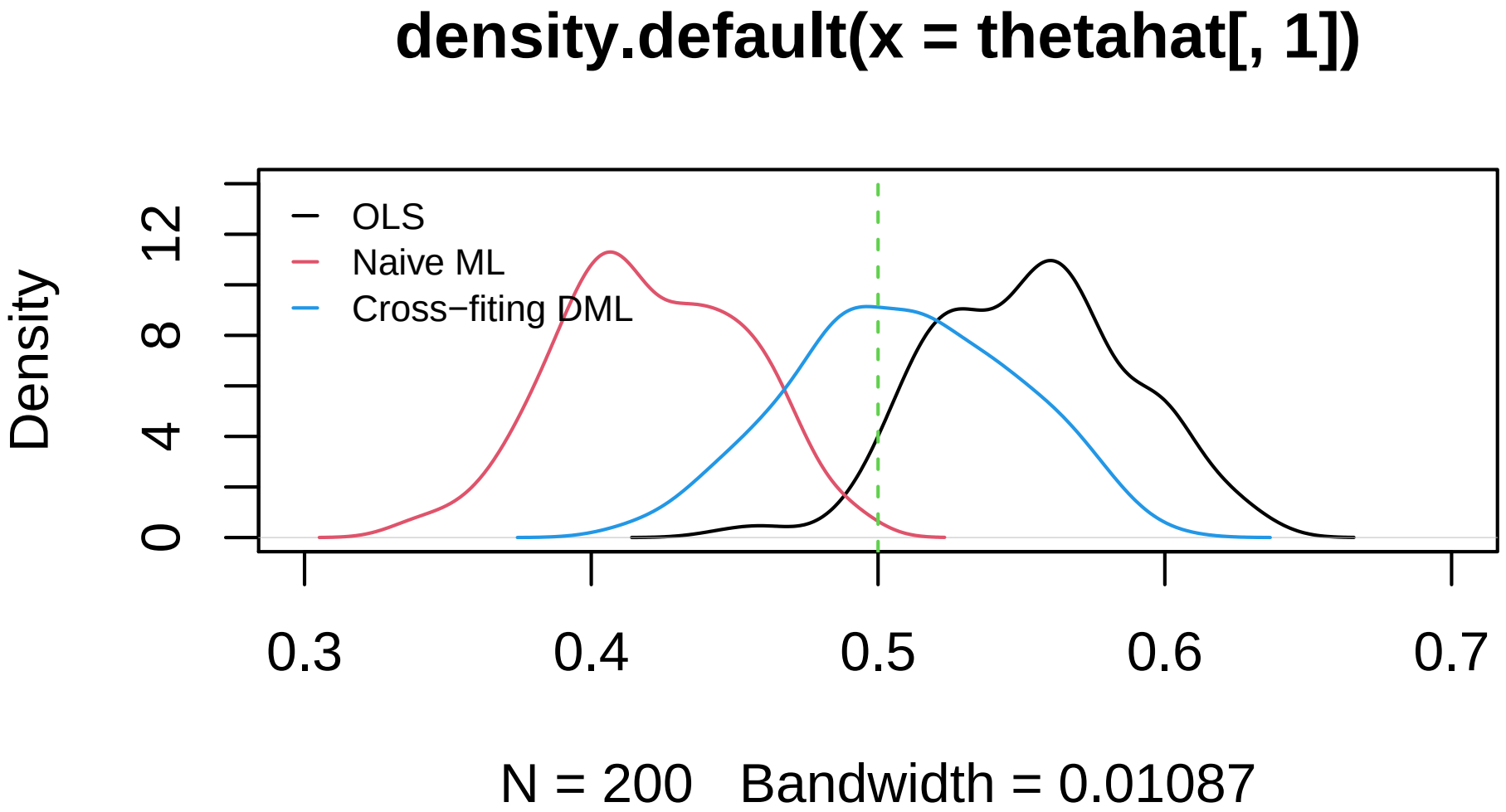
- This is called "Robinson's Transformation" (Robinson, 1988).
- The transformation allows us to achieve "Neyman orthogonality," meaning the bias from estimating nuisance parameters have negligible influence on the estimation of causal parameters.
- Double machine learning is built upon the same idea as the doubly robust estimator.
- You need to get either the response surface or the propensity score correct, and you have higher efficiency by getting both correct.

Double machine learning

- We still have a remainder: $V_i(\hat{g}(\mathbf{X}_i) - g_0(\mathbf{X}_i))$.
- This term only relies on the property of $\hat{g}$.
- If you use LASSO, the remainder converges to zero at a fast rate.
- For more general algorithms, we use sample splitting to eliminate it.
- As $\hat{g}$ is generated on an independent sample, it should be orthogonal to V_i.
- We can split the sample multiple times and take the average over the estimates.
- There is no efficiency loss.

Double machine learning

```
##          OLS      Naive ML Cross-fiting DML
##    0.5532314     0.4208227       0.5089906
```



Double machine learning

- Belloni et al. (2012): use LASSO/Post-LASSO to select instruments.
- Belloni et al. (2013): use LASSO/Post-LASSO to select covariates.
- Chernozhukov et al. (2016): use LASSO/Post-LASSO to select covariates in panel data.
- Belloni et al. (2016): use double machine learning to to estimate any functional.
- Chernozhukov et al. (2018): use double machine learning to estimate nuisance parameters.

Generalization

- We have data from some samples/population
- We care about the effect in other population

Generalization

- We have data from some samples/population
- We care about the effect in other population
- Example: trials in one hospital. One cannot force patients to take the trials.
- Therefore: volunteer population and general population are different.

Generalization framework

- Use the setup from Imai, et al 2008
- Sample n from a finite population of $N >> n$.
- Insample indicator $I_i \in \{0, 1\}$. In sample if $I_i = 1$.
- (I_i, T_i, Y_i) are random variables.

Generalization framework

- Use the setup from Imai, et al 2008
- Sample n from a finite population of $N >> n$.
- Insample indicator $I_i \in \{0, 1\}$. In sample if $I_i = 1$.
- (I_i, T_i, Y_i) are random variables.
- $TE_i = Y_i(1) - Y_i(0)$.
- $SATE = \frac{1}{n} \sum_{i \in \{I_i = 1\}} TE_i$
- $PATE = \frac{1}{N} \sum_1^N TE_i$

Estimator

- Assuming balance between treated group and control group
- Baseline estimator:

$$D = \frac{1}{n/2} \sum_{i \in \{I_i=1, T_i=1\}} Y_i - \frac{1}{n/2} \sum_{i \in \{I_i=1, T_i=0\}} Y_i$$

- Define estimation error: $\Delta = PATE - D$
- Consider observed covariates X_i, unobserved covariates U_i
- $Y_i(t) = g_t(X_i) + h_t(U_i)$

Estimator

$$\Delta = \Delta_S + \Delta_T = \Delta_{S_X} + \Delta_{S_U} + \Delta_{T_X} + \Delta_{T_U}$$

where

$$\underbrace{\Delta_S = PATE - SATE}_{\text{Sample selection}}$$

and

$$\underbrace{\Delta_T = SATE - D}_{\text{Treatment imbalance}}$$

Sample selection

- Δ_S vanish if we have census, or $SATE = PATE = NATE$.
- where $NATE = \frac{1}{N-n} \sum_{i \in \{I_i = 0\}} TE_i$

Sample selection

- Δ_S vanish if we have census, or $SATE = PATE = NATE$.
- where $NATE = \frac{1}{N-n} \sum_{i \in \{I_i = 0\}} TE_i$
- Otherwise:

$$\Delta_{S_X} = \frac{N-n}{N}\Big[\frac{1}{N-n} \sum_{i \in \{I_i = 0\}} (g_1(X_i) - g_0(X_i)) - \frac{1}{n} \sum_{i \in \{I_i = 1\}} (g_1(X_i) - g_0(X_i))\Big]$$

$$\Delta_{S_U} = \frac{N-n}{N}\Big[\frac{1}{N-n} \sum_{i \in \{I_i = 0\}} (h_1(U_i) - h_0(U_i)) - \frac{1}{n} \sum_{i \in \{I_i = 1\}} (h_1(U_i) - h_0(U_i))\Big]$$

Sample selection (cont.)

Alternatively

$$\Delta_{SX} = \frac{N-n}{N} \int \{g_1(X) - g_0(X)\} d\{\hat{F}(X|I=0) - \hat{F}(X|I=1)\}$$

$$\Delta_{SX} = \frac{N-n}{N} \int \{h_1(U) - h_0(U)\} d\{\hat{F}(U|I=0) - \hat{F}(U|I=1)\}$$

Treatment Imbalance

Similarly,

$$\Delta_{TX} = \int \frac{g_1(X) + g_0(X)}{2} d\{\hat{F}(X|T=0, I=1) - \hat{F}(X|T=1, I=1)\}$$

$$\Delta_{TU} = \int \frac{h_1(U) + h_0(U)}{2} d\{\hat{F}(U|T=0, I=1) - \hat{F}(U|T=1, I=1)\}$$

Generalization

- In general, if we have representative sample of target population, and random experiment, great.
- Usually, non-representative sample, or target population $\neq$ reference population

Generalization

- In general, if we have representative sample of target population, and random experiment, great.
- Usually, non-representative sample, or target population $\neq$ reference population
- Low-dimension X and RCT (without worrying U term)
  - Post-stratification:
  - Empirically get $\hat{F}$ and $g_t(X)$ for each cell of X for target population and reference population
  - Same case if sample is non-representative

Generalization via Propensity Score (Stuart et al 2011)

- Key assumption:
  1. $0 < P(I_i = 1|X_i) < 1$ for all X_i
  2. $I \perp \{Y(1), Y(0)\}|X$ or $E(\Delta_{S_U}) = 0$
  3. $T \perp \{I, Y(1), Y(0)\}|X$

Generalization via Propensity Score (Stuart et al 2011)

- Key assumption:
  1. $0 < P(I_i = 1|X_i) < 1$ for all X_i
  2. $I \perp \{Y(1), Y(0)\}|X$ or $E(\Delta_{S_U}) = 0$
  3. $T \perp \{I, Y(1), Y(0)\}|X$
- Propensity $p_i = Pr(I_i = 1|X_i)$
- Propensity difference $\Delta_p = \frac{1}{n} \sum_{i \in \{I_i = 1\}} \hat{p}_i - \frac{1}{N-n} \sum_{i \in \{I_i = 0\}} \hat{p}_i$
- If representative sample (random sample), $E(\Delta_p) = 0$

Generalization via Propensity Score (cont.)

- Otherwise:
  1. IPTW
  2. Full matching
  3. sub-classification

Generalization via Propensity Score (cont.)

- Otherwise:
  1. IPTW
  2. Full matching
  3. sub-classification
- Stuart et al 2011 finds they have similar results, and IPTW is slightly better.

Generalization via Propensity Score (cont.)

- For IPTW, we can see:

$$
\begin{aligned}
& E[\frac{I(1-T)Y}{w(X)(1-e(X))}] \\
&= E(E[\frac{1(I=1)(1-1(T=1))Y}{w(X)(1-e(X))}|Y,X]) \\
&= E(E[\frac{1(I=1)(1-1(T=1))Y(0)}{w(X)(1-e(X))}|Y(0),X]) \\
&= E(\frac{Y(0)}{w(X)(1-e(X))}E[1(I=1)(1-1(T=1))|Y(0),X]) \\
&= E(\frac{Y(0)}{w(X)(1-e(X))}P(I=1|X=x)(1-P(T=1|X=x))) \\
&= E(Y(0))
\end{aligned}
$$

where $w(X) = P(I=1|X=x)$, and $e(x) = P(T=1|X=x)$

Generalization via Propensity Score (cont.)

- For IPTW, we can see:

$$
\begin{aligned}
& E[\frac{I(1-T)Y}{w(X)(1-e(X))}] \\
&= E(E[\frac{1(I=1)(1-1(T=1))Y}{w(X)(1-e(X))}|Y,X]) \\
&= E(E[\frac{1(I=1)(1-1(T=1))Y(0)}{w(X)(1-e(X))}|Y(0),X]) \\
&= E(\frac{Y(0)}{w(X)(1-e(X))}E[1(I=1)(1-1(T=1))|Y(0),X]) \\
&= E(\frac{Y(0)}{w(X)(1-e(X))}P(I=1|X=x)(1-P(T=1|X=x))) \\
&= E(Y(0))
\end{aligned}
$$

  where $w(X) = P(I = 1|X = x)$, and $e(x) = P(T = 1|X = x)$
- It's possible to model them at different level, individual and context level
- It's accessible if population is well defined in dataset.

Generalization on PATT (Hartman et al. 2015)

- Hartman et al. 2015 further defines the sufficient conditions.

Generalization on PATT (Hartman et al. 2015)

- Hartman et al. 2015 further defines the sufficient conditions.
- Setup:
  - Y_{ist} represent potential outcome for a unit i assigned to study sample s and treatment t
  - $s = 1$ means reference context, and $s = 0$ means target population
  - $t = 1$ means setting the individual to treatment
  - S_i, sample indicator; T_i, treatment indicator
  - W_i^T, observed covariates related to the sample selection machenism for membership in reference group v.s. target population
  - $W_i^{C_T}$, observed covariates related to the sample assignment for inclusion of controls in reference group v.s. target population

Estimands:

- $\tau_{SATE} = E(Y_{11} - Y_{10}|S = 1)$
- $\tau_{SAT*} = E(Y_{11}|S = 1, T = t) - E(Y_{10}|S = 1, T = t)$

Estimands:

- $\tau_{SATE} = E(Y_{11} - Y_{10}|S = 1)$
- $\tau_{SAT*} = E(Y_{11}|S = 1, T = t) - E(Y_{10}|S = 1, T = t)$
- $\tau_{PATE} = E(Y_{01} - Y_{00}|S = 0)$
- $\tau_{PATC} = E(Y_{01} - Y_{00}|S = 0, T = 0)$
- $\tau_{PATT} = E(Y_{01} - Y_{00}|S = 0, T = 1)$

Assumption 1:

1. Consistency under parallel studies

$$Y_{i01} = Y_{i11}$$
$$Y_{i00} = Y_{i10}$$

Intuitively, potential outcome does not change by assigned contexts.

Assumption 2:

2. Strong Ignorability of Sample Assignment of Treated

$$(Y_{01}, Y_{11}) \perp S | (W^T, T = 1)$$
$$0 < Pr(S = 1 | W^T, T = 1) < 1$$
$$\rightarrow E(Y_{s1} | S = 0, T = 1) = E_{01}(E(Y_{s1} | W^T, S = 1, T = 1))$$
$$\text{for } s = 0, 1$$
$$E_{01}(E(\cdot | W^T, ...)) = E_{W^T | S=0, T=1}(E(\cdot | W^T, ...))$$

i.e. the characteristics of **treated** units in reference population (W^T for $s = 1, t = 1$) are adjusted to match the characteristics of **treated** units in target population ($s = 0, t = 1$)

Assumption 3:

3. Strong Ignorability of Sample Assignment for Control

$$(Y_{00}, Y_{10}) \perp S | (W^{C_T}, T = 1)$$
$$0 < Pr(S = 1 | W^{C_T}, T = 1) < 1$$
$$\rightarrow E(Y_{s0} | S = 0, T = 1) = E_{01}(E(Y_{s0} | W^{C_T}, S = 1, T = 0))$$
$$\text{for } s = 0, 1$$
$$E_{01}(E(\cdot | W^{C_T}, ...)) = E_{W^{C_T} | S=0, T=1}(E(\cdot | W^{C_T}, ...))$$

i.e. the characteristics of **control** units in reference population (W^{C_T} for $s = 1, t = 1$) are adjusted to match the characteristics of **treated** units in target population ($s = 0, t = 1$)

Assumption 4:

4. Stable Unit Treatment Value Assumption

$$Y_{ist}^{L_i} = Y_{ist}^{L_j}, \forall \text{ treatment vectors } L_i, L_j$$

Theorem 1 (Hartman et al. 2015)

Theorem 1. Assuming consistency and SUTVA hold, if

$$\begin{aligned} & E_{01}(E(Y_{s1}|W^T, S=0, T=1)) - E_{01}(E(Y_{s0}|W^{C_T}, S=0, T=1)) \\ = & E_{01}(E(Y_{s1}|W^T, S=1, T=1)) - E_{01}(E(Y_{s0}|W^{C_T}, S=1, T=1)) \end{aligned}$$

or sample assignment for treated units is strongly ignoreable given W^T, and sample assignment for controls is strongly ignoreable given W^{C_T}, then

$$\tau_{PATT} = E_{01}\{E(Y|W^T, S=1, T=1)\} - E_{01}\{E(Y|W^{C_T}, S=1, T=0)\}$$

Placebo Tests for Checking Assumptions

- Idea: Check whether adjusted outcome is the same as the true observed outcome in target population.
- Suppose one potential outcome is observed in target population. For example, Y_{00} is observed ($S = 0, T = 0$).

Placebo Tests for Checking Assumptions

- Idea: Check whether adjusted outcome is the same as the true observed outcome in target population.
- Suppose one potential outcome is observed in target population. For example, Y_{00} is observed $(S = 0, T = 0)$.
- If assumption 1,3,4 hold, we have
  $E(Y|S = 0, T = 0) = E_{01}\{E(Y|W^{C_T}, S = 1, T = 0)\}$
- Similar if one can observe Y_{01}.
- Implementation: equavlent test (Hartman and Hidalgo 2011)

Estimation

1. Create matched pairs or strata within reference population. (Genetic Mtaching, GenMatch)
2. Reweight the matched pairs according to the characteristics of the target population. (Maximum Entropy Match, MaxEnt)

Some Extensions

- Some consider controlling context level heterogeneity
  - General meta-analysis approach: hierachical model
  - Meager 2018: bayesian hierachical model on ATE of microcredit expansion in 7 settings
- Other post-stratification
  - MRP (Multilevel regression with poststratification, Starting from Gelman and Little 1997)
  - Even in experiment setting (Miratrix et al. 2019)
- Change estimation scheme of propensity
  - Kern et al, 2016, BART (Bayesian Additive Regression Tree)
  - Doubly robust

Reference

- Imai K, King G, Stuart E A. Misunderstandings between experimentalists and observationalists about causal inference[J]. Journal of the royal statistical society: series A (statistics in society), 2008, 171(2): 481-502.
- Stuart E A, Cole S R, Bradshaw C P, et al. The use of propensity scores to assess the generalizability of results from randomized trials[J]. Journal of the Royal Statistical Society: Series A (Statistics in Society), 2011, 174(2): 369-386.
- Hartman E, Grieve R, Ramsahai R, et al. From sample average treatment effect to population average treatment effect on the treated: combining experimental with observational studies to estimate population treatment effects[J]. Journal of the Royal Statistical Society: Series A (Statistics in Society), 2015, 178(3): 757-778.

The End

Good luck with your final!