Heterogeneous Treatment Effects, Causal Inference with Text

Apoorva Lal

May 26, 2022

Stanford

Heterogeneous Treatment Effects

Heterogeneous Treatment Effects - Setup

- For i.i.d. observations $i \in \{1, .., N\}$, we observe $\{Y_i, X_i, T_i\}_i^N$ where:
  - Y_i is the **outcome**
  - $X_i \in \mathbb{R}^k$ is the **feature vector**
  - W_i is the **treatment assignment**
- We posit the existence of **potential outcomes** $Y_i^{(1)}$ and $Y_i^{(0)}$
- Under *Causal Consistency*, *Unconfoundedness*, and *Overlap*, we can estimate treatment effects
- We are interested in the **Conditional Average Treatment Effect (CATE)**:
  - $\text{CATE}_X = \tau(X) = E[Y^{(1)} - Y^{(0)} | X]$

Heterogeneous Treatment Effects - Setup

- For i.i.d. observations $i \in \{1, .., N\}$, we observe $\{Y_i, X_i, T_i\}_i^N$ where:
  - Y_i is the **outcome**
  - $X_i \in \mathbb{R}^k$ is the **feature vector**
  - W_i is the **treatment assignment**
- We posit the existence of **potential outcomes** $Y_i^{(1)}$ and $Y_i^{(0)}$
- Under *Causal Consistency*, *Unconfoundedness*, and *Overlap*, we can estimate treatment effects
- We are interested in the **Conditional Average Treatment Effect (CATE)**:
  - $\text{CATE}_X = \tau(X) = E[Y^{(1)} - Y^{(0)}|X]$
- Plug-in principle: fit the two conditional expectations using flexible learners

Heterogeneous Treatment Effects - Setup

- For i.i.d. observations $i \in \{1, .., N\}$, we observe $\{Y_i, X_i, T_i\}_i^N$ where:
  - Y_i is the **outcome**
  - $X_i \in \mathbb{R}^k$ is the **feature vector**
  - W_i is the **treatment assignment**
- We posit the existence of **potential outcomes** $Y_i^{(1)}$ and $Y_i^{(0)}$
- Under *Causal Consistency*, *Unconfoundedness*, and *Overlap*, we can estimate treatment effects
- We are interested in the **Conditional Average Treatment Effect (CATE)**:
  - $\text{CATE}_X = \tau(X) = E[Y^{(1)} - Y^{(0)}|X]$
- Plug-in principle: fit the two conditional expectations using flexible learners
  - Problems?

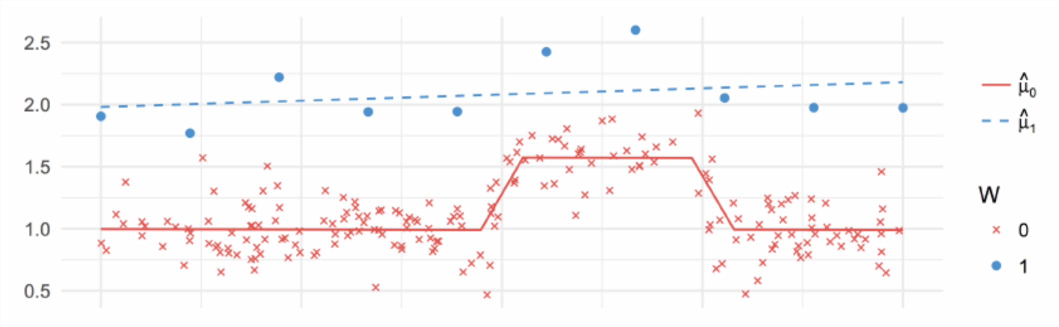
Early Ideas

T-Learner

- fits separate models on the treated and controls.
- Learn $\hat{\mu}_{(0)}(x)$ by predicting Y_i from X_i on the subset of observations with $T_i = 0$.
- Learn $\hat{\mu}_{(1)}(x)$ by predicting Y_i from X_i on the subset of observations with $T_i = 1$.
- Report $\hat{\tau}(x) = \hat{\mu}_{(1)}(x) - \hat{\mu}_{(0)}(x)$.

S-Learner

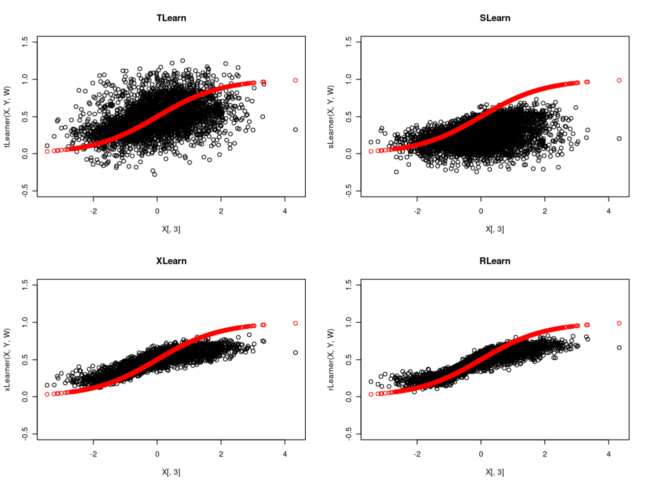
- fits a single model to all the data.
- Learn $\hat{\mu}(z)$ by predicting Y_i from $Z_i := (X_i, T_i)$ on all the data.
- Report $\hat{\tau}(x) = \hat{\mu}((x, 1)) - \hat{\mu}((x, 0))$.



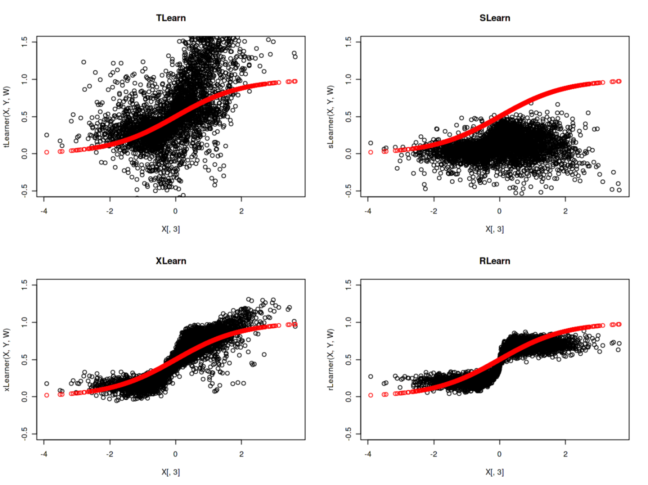
2.5
2.0
1.5
1.0
0.5
$\hat{\mu}_0$
$\hat{\mu}_1$
W
0
1

In action: RCT

- Simulation + Implementation



In action: Confounding



American Political Science Review (2019) 113, 4, 1078–1084
doi:10.1017/S0003055419000443

Letter

Concentrated Burdens: How Self-Interest and Partisanship Shape Opinion on Opioid Treatment Policy

JUSTIN DE BENEDICTIS-KESSNER *Boston University*
MICHAEL HANKINSON *Baruch College*

Figure 1: Paper for Today

Data for Today

- **Research Question**: Do people support an opioid addiction treatment clinic
- being established when it is near them?
- **Design:**: Survey experiment asking:
    - "Do you support the establishment of an opioid addiction treatment clinic [**near/far from**] you?"

Data

- $N = 2008$, but im going to split the data into 10 random samples of roughly
- 200 observations

```
foldMake = function (d, nf = 10) {
    n = nrow(d);
    foldid = rep.int(1:nf, times = ceiling(n/nf))[sample.int(n)]
    split(1:n, foldid)
}
foldAssignments = foldMake(df)
```

450B solution: Estimate OLS with interactions

- $Y_i = \beta_0 + \beta_1 T_i + \beta_2 X_i + \beta_3 T_i \times X_i + \epsilon_i$
- $\widehat{\text{CATE}}_X = \hat{\beta}_1 + \hat{\beta}_3 X_i$
- Why do we need machine learning / regularization to do this?

450B solution: Estimate OLS with interactions

- $Y_i = \beta_0 + \beta_1 T_i + \beta_2 X_i + \beta_3 T_i \times X_i + \epsilon_i$
- $\widehat{\text{CATE}}_X = \hat{\beta}_1 + \hat{\beta}_3 X_i$
- Why do we need machine learning / regularization to do this?
- **Overfitting**: We know that in general, when $k \approx N$, traditional OLS methods will badly overfit
- **Unknown Functional Form**: The analyst does not know what the underlying heterogeneity looks like
- **fishing**: Many methods provide a way to report HTE of varying functional form in an automated way (to avoid fishing) but also avoiding a pre-analysis plan

ATE using OLS

- Lets estimate OLS on the first dataset

```
mod <- lm(support~near, data = df[foldAssignments[[1]], ])
summary(mod)
```

```
##
## Call:
## lm(formula = support ~ near, data = df[foldAssignments[[1]],
##     ])
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -0.564 -0.411 -0.411  0.436  0.589
##
## Coefficients:
##             Estimate Std. Error t value            Pr(>|t|)
## (Intercept)   0.5636     0.0474   11.90 <0.0000000000000002 ***
## near         -0.1525     0.0706   -2.16               0.032 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.497 on 198 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.023,  Adjusted R-squared:  0.0181
## F-statistic: 4.67 on 1 and 198 DF,  p-value: 0.0319
```

Heterogeneous Treatment Effect (HTE) using OLS

- Suppose now we posit that the treatment will be the strongest for homeowners and non-college educated respondents

```
df = df %>% mutate(own2 = scale(own, scale = F), college2 = scale(college, scale = F))

mod <- lm(support ~ near * own2 * college2 , data = df[foldAssignments[[1]], ])
tidy(mod) %>% filter(str_detect(term, "near.*"))
```

term	estimate	std.error	statistic	p.value
near	-0.1597	0.0736	-2.1684	0.0314
near:own2	0.1028	0.1566	0.6566	0.5123
near:college2	0.1161	0.1473	0.7880	0.4317
near:own2:college2	-0.1084	0.3135	-0.3459	0.7298

Heterogeneous Treatment Effect (HTE) using OLS

- There is a temptation to stop here and report a heterogenous treatment effect
- “We find, perhaps surprisingly, that among college educated renters, a closer clinic is preferred to a far away one.”
- “We find suggestive evidence for what we term a *opioid clinic affinity* among college educated renters.[Footnote: The effect is statistically significant at the 20 percent level.]”
- “Although we lack the power to make a strong causal claim, the positive coefficient is consistent with a model of....”

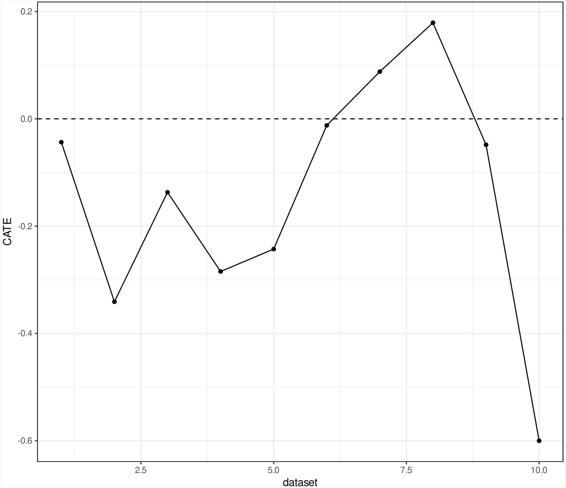
Heterogeneous Treatment Effect (HTE) using OLS

- Lets investigate how robust this is across the 10 datasets

```
cates <- c()
for (i in 1:10){
  coefs <- lm(support~near*own2*college2, data = df[ foldAssignments[[i]], ])$coef
  cates[i] <- coefs['near'] +coefs['near:college2']
}
plt<- ggplot(data = tibble(dataset = 1:10, CATE = cates),
      aes(x = dataset, y = CATE))+
geom_point()+geom_path(group = 1)
```

Heterogeneous Treatment Effect (HTE) using OLS

- Lets investigate how robust this is across the 10 datasets



Heterogeneous Treatment Effect (HTE) using OLS

- Why is this the CATE so variable?

```
##                            [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## non-college+homeowner      62   64   63   68   65   58   58   68   58   58
## non-college+non-homeowner 37   39   37   29   46   31   38   32   39   37
## college+non-homeowner      27   27   28   23   26   26   25   35   26   20
## college+homeowner          73   65   68   75   57   78   75   59   71   81
```

Heterogeneous Treatment Effect (HTE) using OLS

- Why is this the CATE so variable?
- Only 27 people in the {college + non-homeowner} bin!

Causal Forest

```
yn = 'support'; wn = 'near'; xn = c("own", "college")
df2 = df[, c(yn, wn, xn)] %>% na.omit()
y = df2[[yn]]; w = df2[[wn]]
X = df2[, xn] %>% as.matrix()

cf = causal_forest(X, y, w)
average_treatment_effect(cf)
```

```
## estimate  std.err
## -0.14760  0.02217
```

Heterogeneous effects

```
##
## Best linear fit using forest predictions (on held-out data)
## as well as the mean forest prediction as regressors, along
## with one-sided heteroskedasticity-robust (HC3) SEs:
##
##                                Estimate Std. Error t value          Pr(>
## mean.forest.prediction            0.984      0.147    6.67 0.000000000
## differential.forest.prediction   -0.650      0.702   -0.93            0.
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Linear Approximation of Heterogeneous Effects

```r
tau.hat = predict(cf)
d2 = data.frame(X, tauhat = tau.hat[, 1])
lm_robust(tauhat ~ own * college, d2) %>% tidy() %>%
  select(term, estimate, `std.error`)
```

term	estimate	std.error
(Intercept)	-0.1069	0.0002
own	-0.0728	0.0002
college	0.0032	0.0003
own:college	0.0130	0.0003

Causal Inference with Text

Text as Treatment (Fong and Grimmer (2016, 2021))

- Goal: discover treatments and estimate their effects
  - CS version: Fong and Grimmer 2016 - identify treatments and estimate their Average Marginal Component specific Effect (AMCE)
  - PS version: Fong and Grimmer 2021
- Text $\mathbf{T}_i$, potential outcome $Y_i(\mathbf{T}_i)$
- Measured treatment $g(\mathbf{T}_i) =: Z_i$
- Unmeasured treatment $h(\mathbf{T}_i) =: B_i$

Text as Treatment (Fong and Grimmer (2016, 2021))

- Goal: discover treatments and estimate their effects
  - CS version: Fong and Grimmer 2016 - identify treatments and estimate their Average Marginal Component specific Effect (AMCE)
  - PS version: Fong and Grimmer 2021
- Text $\mathbf{T}_i$, potential outcome $Y_i(\mathbf{T}_i)$
- Measured treatment $g(\mathbf{T}_i) =: Z_i$
- Unmeasured treatment $h(\mathbf{T}_i) =: B_i$

1. SUTVA
2. Random Assignment of Texts
3. Measured and Unmeasured representation
4. One of two

- Measured and unmeasured latent treatments independent
- Unmeasured treatments unrelated to outcome

Estimand and Estimator

$$\text{ATE} = \sum_{b \in B} \left(\mathbb{E}\left[Y_i(Z_i = 1, \mathbf{B}_i = b)\right] - \mathbb{E}\left[Y_i(Z_i = 0, \mathbf{B}_i = \mathbf{b})\right] \right) \mathbf{Pr}\left(B_i = b\right)$$

$$\widehat{\text{ATE}} = \mathbb{E}\left[Y_i(\mathbf{T}_i | g(\mathbf{T}_i = 1))\right] - \mathbb{E}\left[Y_i(\mathbf{T}_i | g(\mathbf{T}_i = 0))\right]$$

Trump tweets experiment (Section 5.2)

```
library(tidytext); library(texteffect); library(textdata)
dat <- read.csv("trumpdt.csv")
Y <- dat[,1]; G <- dat[,2:4]; X <- dat[,5:ncol(dat)]
rm(dat)

## Sample Splitting
set.seed(12082017)

training.tweets <- sample(1:(nrow(X)/3), nrow(X)/3*.5)
train.ind <- c()
for (i in 1:length(training.tweets)){
  train.ind <- c(train.ind, 3*(training.tweets[i]-1)+(1:3))
}
```

- Infer Treatments

```
## Fit sIBP with many different parameter figurations so the analyst can cl
## the most substantively interesting run
## Note: This will take a while to run (approx 20 minutes)

sibp.search <- sibp_param_search(X, Y, K = 5, alphas = c(2,3,4),
              sigmasq.ns = c(0.5, 0.75, 1), iters = 5,
      train.ind = train.ind, G = G, seed = s)
save(sibp.search, file = "sibp_search.rds")
```

Identified Latent Treatments

```
load("sibp_search.rds")

# evaluate coherence
# sibp_rank_runs(sibp.search, X, 10)
sibp.fit = sibp.search[["3"]][["1"]][[1]]

sibp_top_words(sibp.fit, colnames(X))
```

```
##       [,1]         [,2]      [,3]        [,4]      [,5]
##  [1,] "minister"   "nytimes" "obamacare" "stock"   "hunt"
##  [2,] "prime"      "failing" "repeal"    "cnn"     "witch"
##  [3,] "states"     "alabama" "replace"   "market"  "insurance"
##  [4,] "united"     "luther"  "pass"      "nbc"     "players"
##  [5,] "responders" "strange" "dead"      "abc"     "companies"
##  [6,] "behalf"     "korea"   "premiums"  "travel"  "total"
##  [7,] "korea"      "north"   "cuts"      "players" "nfl"
##  [8,] "pence"      "china"   "stock"     "ban"     "flag"
##  [9,] "flotus"     "wrong"   "insurance" "fake"    "anthem"
## [10,] "north"      "abc"     "tax"       "nfl"     "dems"
```

Effect estimates by group

	Model 1	Model 2	Model 3
(Intercept)	−82.943	−1.355	95.551
	(1.703)	(1.297)	(1.023)
Z1	26.931	16.575	5.363
	(7.714)	(5.876)	(4.634)
Z2	−29.423	−28.136	−16.620
	(8.098)	(6.168)	(4.865)
Z3	−19.581	−15.622	−0.192
	(6.413)	(4.885)	(3.853)
Z4	4.762	5.640	6.685
	(9.498)	(7.235)	(5.706)
Z5	−29.515	−15.210	2.028
	(11.556)	(8.803)	(6.942)
Num.Obs.	752	752	752
R2	0.054	0.055	0.017
R2 Adj.	0.047	[illegible]	[illegible]

Workflow

Workflow

- Learn to use the command line for large/long-running jobs
    - Farmshare / Sherlock access
- Spatial data