Quant II

Lab 2: Regression

Junlong Aaron Zhou

February 12, 2021

Today's plan

- Regression
- Effective samples
- Causal inference from a machine learning perspective

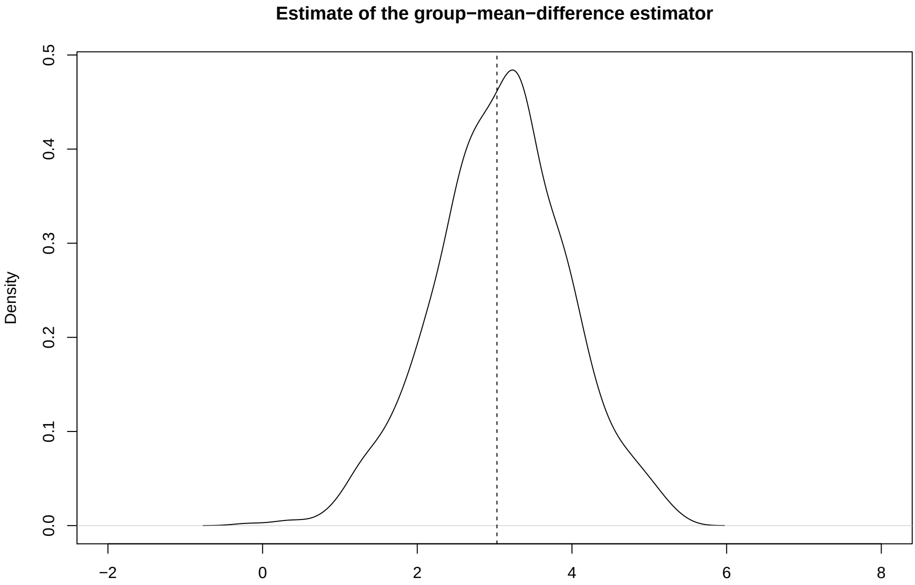
Covariate Adjustment in sampling

- Imagine that we are biologists who are interested in leaf size.
- Finding the size of leaves is hard, but weighting leaves is easy.
- We can use auxilliary information to be smarter:
  - Sample from leaves on a tree.
  - Measure their size and weight.
  - Let $\bar{y}_s$ be the average size in the sample.
  - Let $\bar{x}_s$ be the average weight in the sample.
  - We know that $\bar{y}_s$ unbiased and consistent for $\bar{y}$
  - But we have extra information!
  - We also have $\bar{x}$ (all the weights)
  - This motivates the regression estimator:
    $\hat{\bar{y}} = \bar{y}_s + \beta(\bar{x} - \bar{x}_s)$
  - We get β by a regression of leaf area on weight in the sample.

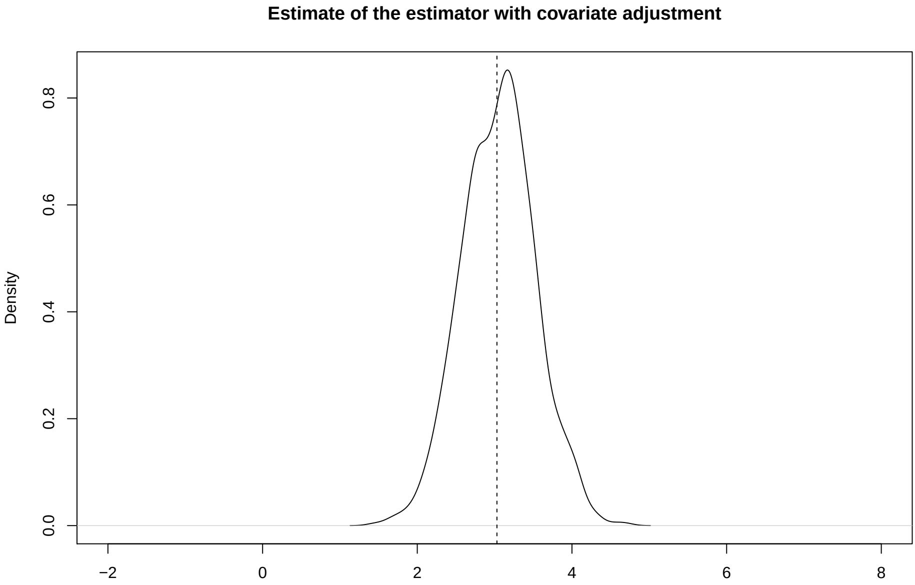
Efficiency from using covariates

```
X1 <- rnorm(N_pop, 3, 1)
X1_demeaned <- X1 - mean(X1)
Y0 <- abs(rnorm(N_pop, 5, 2)) + 3*X1 #+ 0.4*X1^2
Y1 <- Y0 + rnorm(N_pop, 3, 1)
TE <- Y1 - Y0
ATE <- mean(TE)
D <- rbinom(N_pop, 1, 0.3)
Y <- D*Y1 + (1-D)*Y0
reg_formula1 <- paste0(Y.name, "~", D.name)
reg_formula2 <- paste0(Y.name, "~", D.name, "+", X1.name)
reg_formula3 <- paste0(Y.name, "~", D.name, "*", X2.name)
```

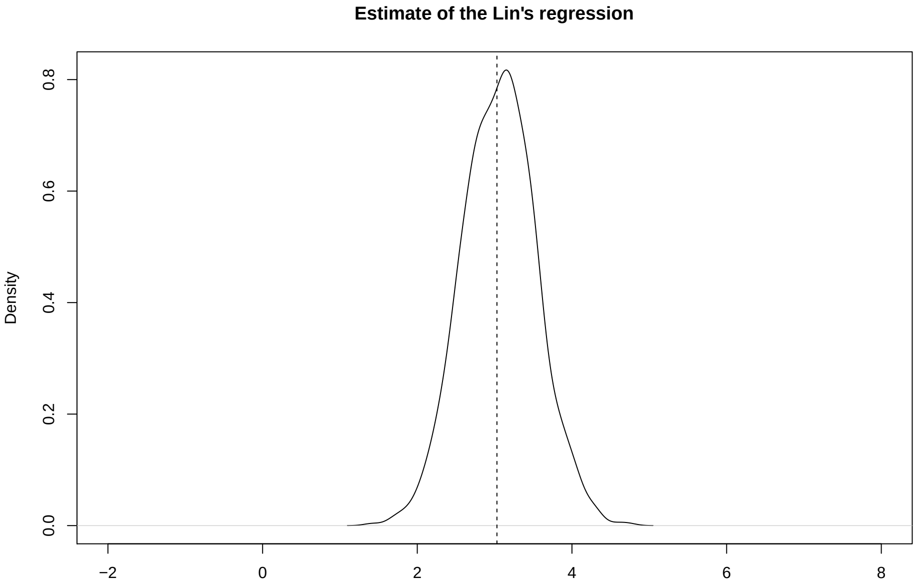
Efficiency from using covariates



Efficiency from using covariates



Efficiency from using covariates



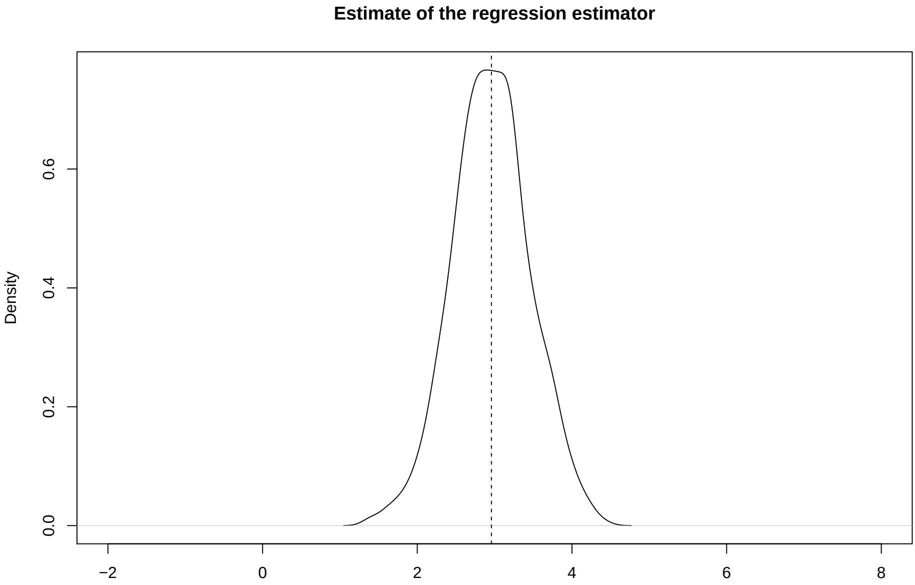
Efficiency from using covariates

```
## The true ATE is 3.029974
## The average of estimates is 3.074189
## The average SE of ATE estimates is 0.9028215
## The average of reg estimates (no cov) is 3.074189
## The average SE of reg estimates (no cov) is 0.9028215
## The average of reg estimates (cov) is 3.052842
## The average SE of reg estimates (no cov) is 0.4798321
## The average of reg estimates (Lin) is 3.059875
## The average SE of reg estimates (Lin) is 0.4825305
```

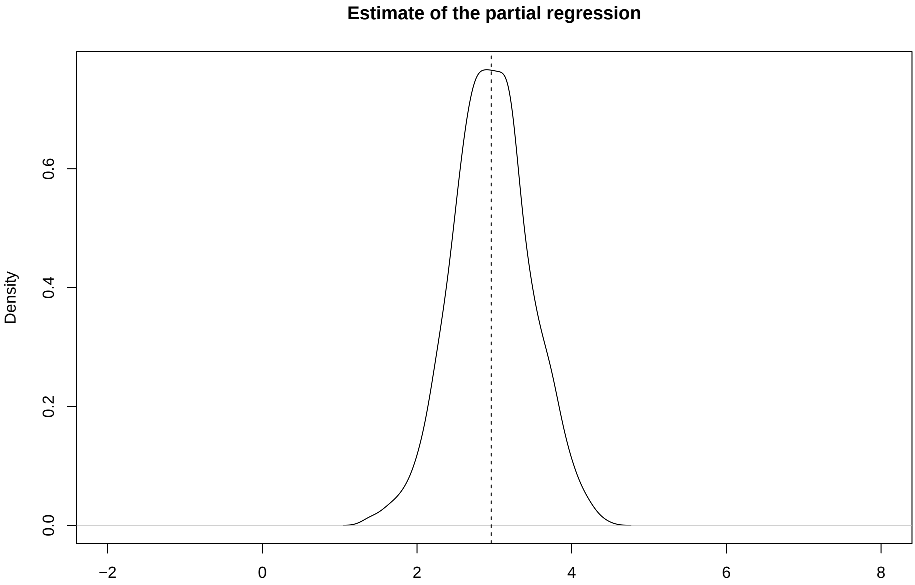
Partial regression

```
reg_formula1 <- paste0(Y.name, "~", D.name, "+", X1.name)
reg_formula2 <- paste0(D.name, "~", X1.name)
reg_formula3 <- paste0(Y.name, "~", X1.name)

reg1 <- lm(as.formula(reg_formula1), data = data.pop)
reg2 <- lm(as.formula(reg_formula2), data = data.pop)
reg3 <- lm(as.formula(reg_formula3), data = data.pop)

lm_est[i] <- coefficients(reg1)[2]
residual_Y <- residuals(reg3)
residual_D <- residuals(reg2)
lm_est_par[i] <- coefficients(lm(residual_Y~residual_D))[2]
```

Partial regression



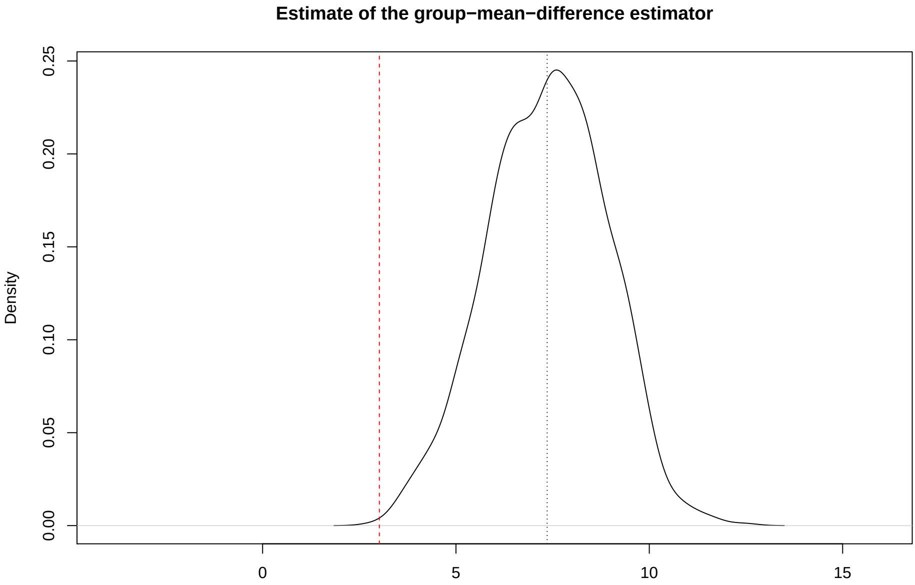
Partial regression



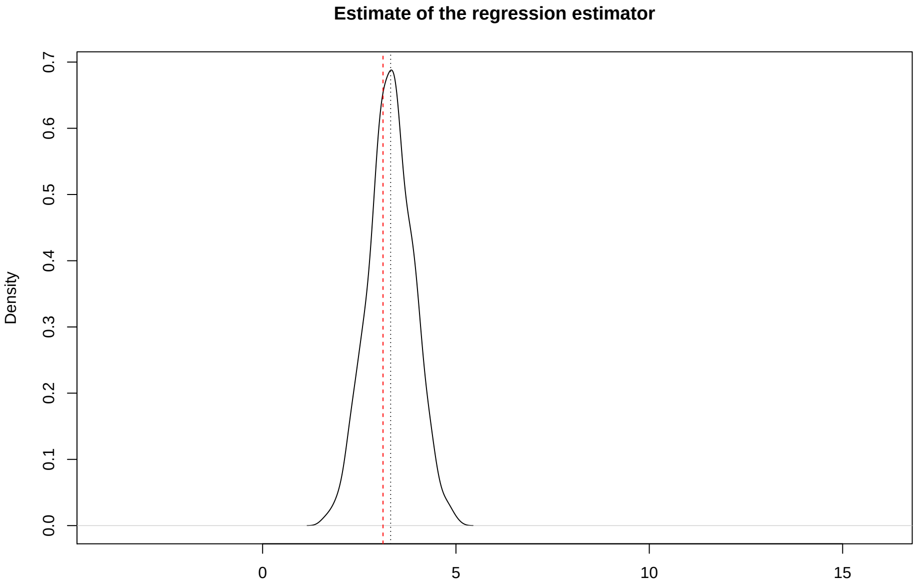
Bias due to confounders

```
N_pop <- 100
X1 <- rnorm(N_pop, 3, 1)
Y0 <- abs(rnorm(N_pop, 5, 2)) + 3*X1 + 0.6*X1^2
Y1 <- Y0 + rnorm(N_pop, 3, 1)
TE <- Y1 - Y0
ATE <- mean(TE)
pscore <- exp(-3 + 0.2*X1 + 0.1*X1^2)/(exp(-3 + 0.2*X1 + 0.1*
D <- rbinom(N_pop, 1, pscore)
```

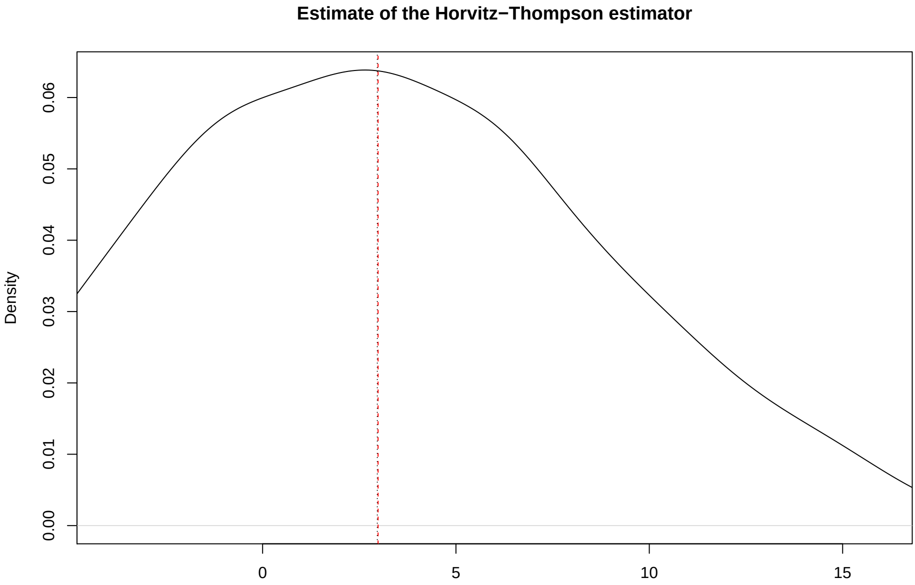
Bias due to confounders



Regression adjustment



Weighting adjustment



Effective samples

- The key result that we are going to use:

$$\hat{\beta} \xrightarrow{p} \frac{E[w_i \tau_i]}{E[w_i]}, \text{ where } w_i = (D_i - E[D_i|X_i])^2 = var(D_i|X_i)$$

- How did we get here?
- Remember that multiple regression estimates are equivalent to weighted averages of unit-specific contributions.
- These weights are driven by the conditional variance of the treatment of interest.
- The bias does not disappear even in the limit.

Effective samples

- We estimate these weights with:
  $\hat{w}_i = \hat{e}^2_{D,i}$ where $e^2_{D,i}$ is the ith squared residual.
- What does this imply? Which units will have a higher w_i? Why is this important?
- Basically the units whose treatment values are not well explained by the covariates.
- If the covariates perfectly predict your assignment to treatment, then you contribute no information to the estimate of β.

Effective samples

- We will use these weights to get a sense for what the effective sample is by examining the weight allocated to particular strata.
- We will be looking at Egan and Mullin (2012).
- The paper looks at how people translate their personal experiences into political attitudes.
- To solve the identification problem, the authors exploit the effect of local weather variations on beliefs in global warming.
- But what is the effective sample?
- In other words, where is weather (conditional on covariates) most variable?
- That's what we'll explore.

Egan and Mullin

```r
require(foreign)
```

```
## Loading required package: foreign
```

```r
d <- read.dta("gwdataset.dta")
zips <- read.dta("zipcodetostate.dta")
zips <- unique(zips[, c("statenum","statefromzipfile")])
pops <- read.csv("population_ests_2013.csv")
pops$state <- tolower(pops$NAME)
d$getwarmord <- as.double(d$getwarmord)
```

Base Model

```r
summary(reg_out)$coefficients[1:10,]
```

```
##                    Estimate  Std. Error   t value   Pr(>|t|)
## (Intercept)     1.945740062 0.771478843 2.5220913 0.01169077
## ddt_week        0.004857915 0.002475887 1.9620908 0.04979656
## wbnid_num3103 0.843451519 0.922666490 0.9141456 0.36067588
## wbnid_num3154 1.575071541 0.973391215 1.6181280 0.10568587
## wbnid_num3159 1.903629413 1.021302199 1.8639237 0.06237963
## wbnid_num3804 1.406498119 0.794035963 1.7713280 0.07655528
## wbnid_num3810 1.330878449 0.806312016 1.6505750 0.09887602
## wbnid_num3811 1.082204367 0.798796489 1.3547936 0.17553267
## wbnid_num3812 1.219327925 0.803974284 1.5166255 0.12941222
## wbnid_num3813 0.986084952 0.829563706 1.1886790 0.23461152
```

Estimate the weights

- We can simply square the residuals of a partial regression to get $\hat{e}^2_{D,i}$:

```
D_formula <- paste0(D, "~", paste0(X, collapse = "+"))

outD <- lm(as.formula(D_formula),d)
eD2 <- residuals(outD)^2
```

Effective sample statistics

- We can use these estimated weights for examining the sample.

```
compare_samples<- d[, c("wave", "ddt_week", "ddt_twoweeks",
  "ddt_threeweeks", "party_rep", "attend_1", "ideo_conservative",
  "age_1824", "educ_hsless")]
compare_samples <- apply(compare_samples,2,function(x)
  c(mean(x),sd(x),weighted.mean(x,eD2),
    sqrt(weighted.mean((x-weighted.mean(x,eD2))^2,eD2))))
compare_samples <- t(compare_samples)
colnames(compare_samples) <- c("Nominal Mean", "Nominal SD",
      "Effective Mean", "Effective SD")
```

Effective Sample Statistics

```
compare_samples
```

```
##                   Nominal Mean Nominal SD Effective Mean Effective SD
## wave                3.09693726  1.4252527     3.20788200    1.5609143
## ddt_week            3.83548593  5.9047249     5.11579140   10.8980228
## ddt_twoweeks        3.85505617  5.4572382     5.00137435    9.2262827
## ddt_threeweeks      3.96719696  4.7689594     5.10859485    8.4348180
## party_rep           0.29527208  0.4561989     0.28978321    0.4536617
## attend_1            0.11433244  0.3182383     0.12343459    0.3289354
## ideo_conservative   0.31132917  0.4630715     0.29325249    0.4552532
## age_1824            0.07195956  0.2584402     0.06881146    0.2531333
## educ_hsless         0.34151056  0.4742516     0.31219962    0.4633908
```

Effective sample maps

- But one of the most interesting things is to see this visually.
- Where in the US does the effective sample emphasize?
- To get at this, we'll use some tools in R that make this incredibly easy.
- In particular, we'll do this in ggplot2.

Effective sample maps

```r
# Effective sample by state
wt.by.state <- tapply(eD2,d$statenum,sum)
wt.by.state <- wt.by.state/sum(wt.by.state)*100
wt.by.state <- cbind(eD2=wt.by.state,statenum=names(wt.by.state))
data_for_map <- merge(wt.by.state,zips,by="statenum")
# Nominal Sample by state
wt.by.state <- tapply(rep(1,6726),d$statenum,sum)
wt.by.state <- wt.by.state/sum(wt.by.state)*100
wt.by.state <- cbind(Nom=wt.by.state,statenum=names(wt.by.state))
data_for_map <- merge(data_for_map,wt.by.state,by="statenum")
```

Effective sample maps

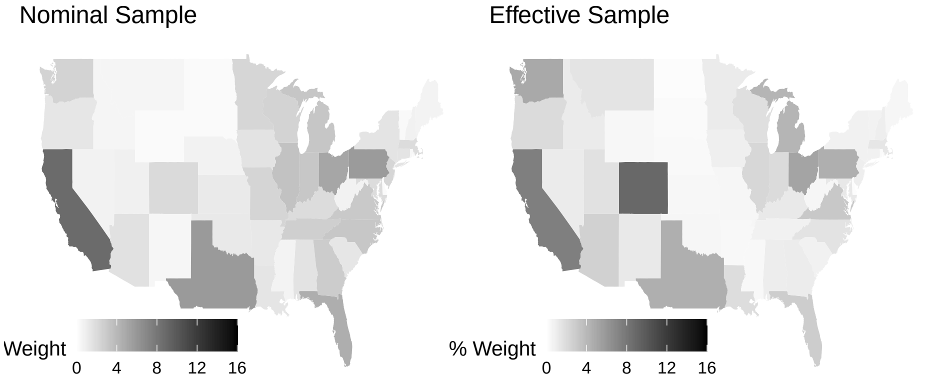
```r
# Get correct state names
require(maps,quietly=TRUE)
data(state.fips)
data_for_map <- merge(state.fips,data_for_map,by.x="abb",
                      by.y="statefromzipfile")
data_for_map$eD2 <- as.double(as.character(data_for_map$eD2))
data_for_map$Nom <- as.double(as.character(data_for_map$Nom))
data_for_map$state <- sapply(as.character(data_for_map$polyname),
                             function(x)strsplit(x,":")[[1]][1])
data_for_map$Diff <- data_for_map$eD2 - data_for_map$Nom
data_for_map <- merge(data_for_map,pops,by="state")
data_for_map$PopPct <- data_for_map$POPESTIMATE2013/sum(
  data_for_map$POPESTIMATE2013)*100
data_for_map$PopDiffEff <- data_for_map$eD2 -
  data_for_map$PopPct
data_for_map$PopDiffNom <- data_for_map$Nom - data_for_map$PopPct
data_for_map$PopDiff <- data_for_map$PopDiffEff - data_for_map$PopDiffNom
require(ggplot2,quietly=TRUE)
state_map <- map_data("state")
```

More setup

```
plotEff <- ggplot(data_for_map,aes(map_id=state))
plotEff <- plotEff + geom_map(aes(fill=eD2), map = state_map)
plotEff <- plotEff + expand_limits(x = state_map$long, y =
                                     state_map$lat)
plotEff <- plotEff + scale_fill_continuous("% Weight",
                                           limits=c(0,16),low="white", high
plotEff <- plotEff + labs(title = "Effective Sample")
plotEff <- plotEff + theme(
        legend.position=c(.2,.1),legend.direction = "horizontal",
        axis.line = element_blank(), axis.text =
          element_blank(),
        axis.ticks = element_blank(), axis.title = element_blank(),
        panel.background = element_blank(), plot.background = element_blank
        panel.border = element_blank(), panel.grid = element_blank()
)

plotNom <- ggplot(data_for_map,aes(map_id=state))
plotNom <- plotNom + geom_map(aes(fill=Nom), map = state_map)
plotNom <- plotNom + expand_limits(x = state_map$long, y = state_map$lat)
plotNom <- plotNom + scale_fill_continuous("% Weight",
```

And the maps

```
require(gridExtra,quietly=TRUE)
grid.arrange(plotNom,plotEff,ncol=2)
```

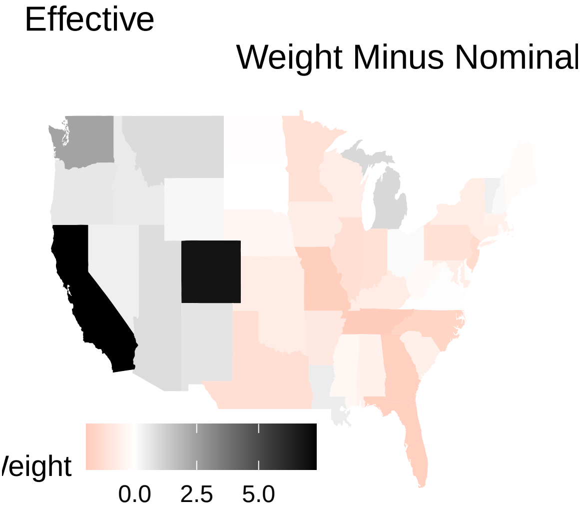


Setup comparison plot

```
plotDiff <- ggplot(data_for_map,aes(map_id=state))
plotDiff <- plotDiff + geom_map(aes(fill=Diff),
                                map = state_map)
plotDiff <- plotDiff + expand_limits(x = state_map$long,
                                     y =
                                       state_map$lat)
plotDiff <- plotDiff + scale_fill_gradient2("% Weight",
                                            low = "red",
                                            mid = "white",
                                            high = "black")
plotDiff <- plotDiff + labs(title = "Effective
                      Weight Minus Nominal Weight")
plotDiff <- plotDiff + theme(
        legend.position=c(.2,.1),legend.direction = "horizontal",
        axis.line = element_blank(), axis.text = element_blank(),
        axis.ticks = element_blank(), axis.title = element_blank(),
        panel.background = element_blank(), plot.background = element_blank
        panel.border = element_blank(), panel.grid = element_blank()
)
```

Difference in weights

```
plotDiff
```



Causal inference from a machine learning perpective

- Now we have been familiar with the Rubin model:

$$Y_i = \begin{cases} Y_i(1) \text{ if } D_i = 1 \\ Y_i(0) \text{ if } D_i = 0 \end{cases}$$

Causal inference from a machine learning perpective

- Now we have been familiar with the Rubin model:

$$Y_i = \begin{cases} Y_i(1) \text{ if } D_i = 1 \\ Y_i(0) \text{ if } D_i = 0 \end{cases}$$

- For each i, we observe either $Y_i(0)$ or $Y_i(1)$ ("Fundamental problem of causal inference").

Causal inference from a machine learning perpective

- Now we have been familiar with the Rubin model:

$$Y_i = \begin{cases} Y_i(1) \text{ if } D_i = 1 \\ Y_i(0) \text{ if } D_i = 0 \end{cases}$$

- For each i, we observe either $Y_i(0)$ or $Y_i(1)$ ("Fundamental problem of causal inference").
- Suppose we are interested in ATT, then we just need to know $Y_i(0)$ for each treated unit.

Causal inference from a machine learning perpective

- Now we have been familiar with the Rubin model:

$$Y_i = \begin{cases} Y_i(1) \text{ if } D_i = 1 \\ Y_i(0) \text{ if } D_i = 0 \end{cases}$$

- For each i, we observe either $Y_i(0)$ or $Y_i(1)$ ("Fundamental problem of causal inference").
- Suppose we are interested in ATT, then we just need to know $Y_i(0)$ for each treated unit.
- It is a prediction problem: $\hat{Y}_i(0) = f(\mathbf{X}, \mathbf{Y}_{(-\mathbf{i})})$.
- If we want to estimate ATE rather than ATT, just do another prediction for $\hat{Y}_i(1)$.

Causal inference from a machine learning perpective

- That's where machine learning enters!

Causal inference from a machine learning perpective

- That's where machine learning enters!
- The target of machine learning algorithms is to find a prediction function $\hat{f}$ that minimizes the expected squared prediction error (ESPE), $E\left[(f - \hat{f})^2\right]$ (in practice we use MSPE)

Causal inference from a machine learning perpective

- That's where machine learning enters!
- The target of machine learning algorithms is to find a prediction function $\hat{f}$ that minimizes the expected squared prediction error (ESPE), $E[(f-\hat{f})^2]$ (in practice we use MSPE)
- It is easy to see that

$$
\begin{aligned}
E[(f-\hat{f})^2] &= E[f^2 - 2*f*\hat{f} + \hat{f}^2] \\
&= f^2 - 2*f*E[\hat{f}] + E[\hat{f}^2] \\
&= f^2 - 2*f*E[\hat{f}] + E[\hat{f}]^2 - E[\hat{f}]^2 + E[\hat{f}^2] \\
&= (E[\hat{f}] - f)^2 + E[\hat{f}^2] - E[\hat{f}]^2 \\
&= (Bias(\hat{f}))^2 + Var(\hat{f})
\end{aligned}
$$

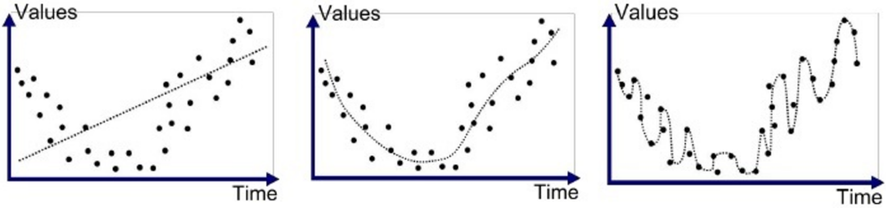
Causal inference from a machine learning perpective

- That's where machine learning enters!
- The target of machine learning algorithms is to find a prediction function $\hat{f}$ that minimizes the expected squared prediction error (ESPE), $E[(f-\hat{f})^2]$ (in practice we use MSPE)
- It is easy to see that

$$\begin{aligned} E[(f-\hat{f})^2] &= E[f^2 - 2*f*\hat{f} + \hat{f}^2] \\ &= f^2 - 2*f*E[\hat{f}] + E[\hat{f}^2] \\ &= f^2 - 2*f*E[\hat{f}] + E[\hat{f}]^2 - E[\hat{f}]^2 + E[\hat{f}^2] \\ &= (E[\hat{f}] - f)^2 + E[\hat{f}^2] - E[\hat{f}]^2 \\ &= (Bias(\hat{f}))^2 + Var(\hat{f}) \end{aligned}$$

- This is called bias-variance trade-off.
- A method with smaller bias usually has larger variance.

Bias and variance



Underfitted

Good Fit/Robust

Overfitted

Causal inference from a machine learning perpective

- In causal inference, we train a model based on the control group observations, then use the model to predict counterfactuals.
- The selection of models depends on the assumption you impose (based on substantive knowledge).

Causal inference from a machine learning perpective

- In causal inference, we train a model based on the control group observations, then use the model to predict counterfactuals.
- The selection of models depends on the assumption you impose (based on substantive knowledge).
- If $\hat{f} = \bar{Y}_{D_i=0}$, what do we have?

Causal inference from a machine learning perpective

- In causal inference, we train a model based on the control group observations, then use the model to predict counterfactuals.
- The selection of models depends on the assumption you impose (based on substantive knowledge).
- If $\hat{f} = \bar{Y}_{D_i=0}$, what do we have?
  Random experiment.

Causal inference from a machine learning perpective

- In causal inference, we train a model based on the control group observations, then use the model to predict counterfactuals.
- The selection of models depends on the assumption you impose (based on substantive knowledge).
- If $\hat{f} = \bar{Y}_{D_i=0}$, what do we have?
  Random experiment.
- If $\hat{f} = \bar{Y}_{D_i=0,\mathbf{X}=\mathbf{x}}$, what do we have?

Causal inference from a machine learning perpective

- In causal inference, we train a model based on the control group observations, then use the model to predict counterfactuals.
- The selection of models depends on the assumption you impose (based on substantive knowledge).
- If $\hat{f} = \bar{Y}_{D_i=0}$, what do we have?
  Random experiment.
- If $\hat{f} = \bar{Y}_{D_i=0,\mathbf{X}=\mathbf{x}}$, what do we have?
  Blocking experiment or matching.

Causal inference from a machine learning perpective

- In causal inference, we train a model based on the control group observations, then use the model to predict counterfactuals.
- The selection of models depends on the assumption you impose (based on substantive knowledge).
- If $\hat{f} = \bar{Y}_{D_i=0}$, what do we have?
  Random experiment.
- If $\hat{f} = \bar{Y}_{D_i=0,\mathbf{X}=\mathbf{x}}$, what do we have?
  Blocking experiment or matching.
- Now, what is the assumption behind regression?

Causal inference from a machine learning perpective

- In causal inference, we train a model based on the control group observations, then use the model to predict counterfactuals.
- The selection of models depends on the assumption you impose (based on substantive knowledge).
- If $\hat{f} = \bar{Y}_{D_i=0}$, what do we have?
  Random experiment.
- If $\hat{f} = \bar{Y}_{D_i=0,\mathbf{X}=\mathbf{x}}$, what do we have?
  Blocking experiment or matching.
- Now, what is the assumption behind regression?
  $\hat{f} = \mathbf{X}_{D_i=0}\beta$ (Linearity)
  $\gamma_i = \gamma$ for any i (Constant treatment effect)

Causal inference from a machine learning perpective

- In causal inference, we train a model based on the control group observations, then use the model to predict counterfactuals.
- The selection of models depends on the assumption you impose (based on substantive knowledge).
- If $\hat{f} = \bar{Y}_{D_i=0}$, what do we have?
  Random experiment.
- If $\hat{f} = \bar{Y}_{D_i=0,\mathbf{X}=\mathbf{x}}$, what do we have?
  Blocking experiment or matching.
- Now, what is the assumption behind regression?
  $\hat{f} = \mathbf{X}_{D_i=0}\beta$ (Linearity)
  $\gamma_i = \gamma$ for any i (Constant treatment effect)
- Matching: low bias and high variance; regression: high bias and low variance

Causal inference from a machine learning perpective

- It is straightfoward to drop the constant treatment effect assumption $\hat{\gamma}_i = Y_i - \mathbf{X}_{D_i=0}\hat{\beta}$ (Regression with interaction)
- Replacing $\mathbf{X}_{D_i=0}\beta$ with $(\mathbf{X}_{D_i=0} - \bar{\mathbf{X}}_{D_i=0})\beta$, we get the more efficient option: Lin's regression

Causal inference from a machine learning perpective

- It is straightfoward to drop the constant treatment effect assumption $\hat{\gamma}_i = Y_i - \mathbf{X}_{D_i=0}\hat{\beta}$ (Regression with interaction)
- Replacing $\mathbf{X}_{D_i=0}\beta$ with $(\mathbf{X}_{D_i=0} - \bar{\mathbf{X}}_{D_i=0})\beta$, we get the more efficient option: Lin's regression
- Question: How to get rid of the linearity assumption?

Problems with naive regression

- It is biased and inconsistent under treatment effect heterogeneity.

Problems with naive regression

- It is biased and inconsistent under treatment effect heterogeneity.
- What is its expectation then?
  Abadie et al. (2020): a weighted sum of the true individualistic effects under linearity, and a weighted sum of something without linearity.

Problems with naive regression

- It is biased and inconsistent under treatment effect heterogeneity.
- What is its expectation then?
  Abadie et al. (2020): a weighted sum of the true individualistic effects under linearity, and a weighted sum of something without linearity.
- Should we add as many covariates as possible?
  No. Covariates may sometimes amplify the existing bias (Middleton et al., 2016)

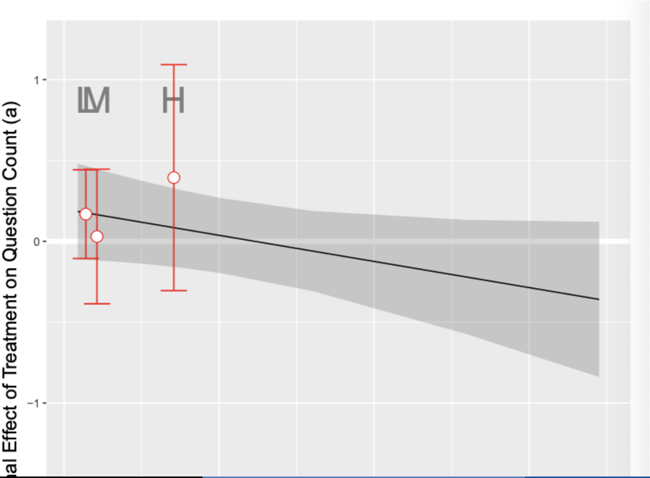
1. X may absorb the variation of D and reduces its explanatory power of Y.
2. If X is negatively correlated with Y and the unobservables are positively correlated with Y, leaving X outside the regression may offset the impact of the unobservables.

Problems with naive regression

- Don't forget the overlapping assumption!

Problems with naive regression

- Don't forget the overlapping assumption!
- Hainmueller, Mummolo, and Xu (2018): When overlapping does not hold, the estimation relies on extrapolation



More complicated models in causal inference

- Regression is often underfitted.

More complicated models in causal inference

- Regression is often underfitted.
- We can use more complicated models to further reduce the MSPE.

More complicated models in causal inference

- Regression is often underfitted.
- We can use more complicated models to further reduce the MSPE.
- With no extra assumptions on model specification: agnostic, or non-parametric estimation

More complicated models in causal inference

- Regression is often underfitted.
- We can use more complicated models to further reduce the MSPE.
- With no extra assumptions on model specification: agnostic, or non-parametric estimation
  Group-mean difference, Matching

More complicated models in causal inference

- Regression is often underfitted.
- We can use more complicated models to further reduce the MSPE.
- With no extra assumptions on model specification: agnostic, or non-parametric estimation
  Group-mean difference, Matching
- When a complete model is specified: Parametric estimation

More complicated models in causal inference

- Regression is often underfitted.
- We can use more complicated models to further reduce the MSPE.
- With no extra assumptions on model specification: agnostic, or non-parametric estimation
  Group-mean difference, Matching
- When a complete model is specified: Parametric estimation
  Regression, Probit, Logit, All Bayesian approaches, etc.

More complicated models in causal inference

- Regression is often underfitted.
- We can use more complicated models to further reduce the MSPE.
- With no extra assumptions on model specification: agnostic, or non-parametric estimation
  Group-mean difference, Matching
- When a complete model is specified: Parametric estimation
  Regression, Probit, Logit, All Bayesian approaches, etc.
- With some "structure" assumed for $\hat{f}$: Semi-parametric estimation

More complicated models in causal inference

- Regression is often underfitted.
- We can use more complicated models to further reduce the MSPE.
- With no extra assumptions on model specification: agnostic, or non-parametric estimation
  Group-mean difference, Matching
- When a complete model is specified: Parametric estimation
  Regression, Probit, Logit, All Bayesian approaches, etc.
- With some "structure" assumed for $\hat{f}$: Semi-parametric estimation
  Kernelized or serial estimation, factor models