



# Section 1: Hospital Pricing and Selection on Observables

---

Ian McCarthy | Emory University  
Econ 470 & HLTH 470

# Table of contents

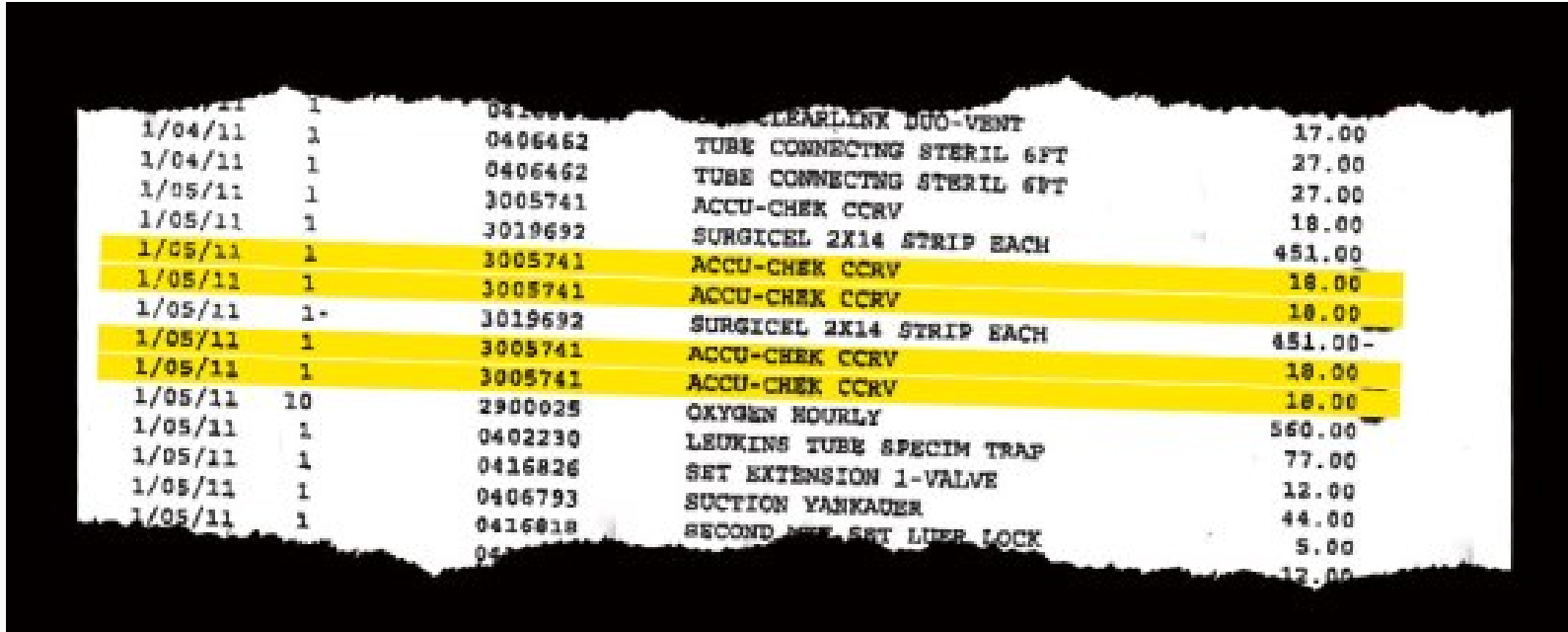
1. Hospital Pricing
2. HCRIS Data
3. Potential Outcomes Framework
4. Selection on Observables
5. Methods
6. Pricing and Profit Status

# Background on Hospital Pricing

---

# What is a hospital price?

Defining characteristic of hospital services: *it's complicated!*



1/04/11	1	0410000	CLEARLINK DUO-VENT	17.00
1/04/11	1	0406462	TUBE CONNECTING STERIL 6FT	27.00
1/04/11	1	0406462	TUBE CONNECTING STERIL 6FT	27.00
1/05/11	1	3005741	ACCU-CHEK CCRV	18.00
1/05/11	1	3019692	SURGICEL 2X14 STRIP EACH	451.00
1/05/11	1	3005741	ACCU-CHEK CCRV	18.00
1/05/11	1	3005741	ACCU-CHEK CCRV	18.00
1/05/11	1-	3019692	SURGICEL 2X14 STRIP EACH	451.00-
1/05/11	1	3005741	ACCU-CHEK CCRV	18.00
1/05/11	1	3005741	ACCU-CHEK CCRV	18.00
1/05/11	10	29000025	OXYGEN HOURLY	560.00
1/05/11	1	0402230	LEUKINS TUBE SPECIM TRAP	77.00
1/05/11	1	0416826	SET EXTENSION 1-VALVE	12.00
1/05/11	1	0406793	SUCTION YANKAER	44.00
1/05/11	1	0416818	SECOND SET LUER LOCK	5.00
		0416818		12.00

Brill, Steven. 2013. "Bitter Pill: Why Medical Bills are Killing Us." \*Time Magazine\*.

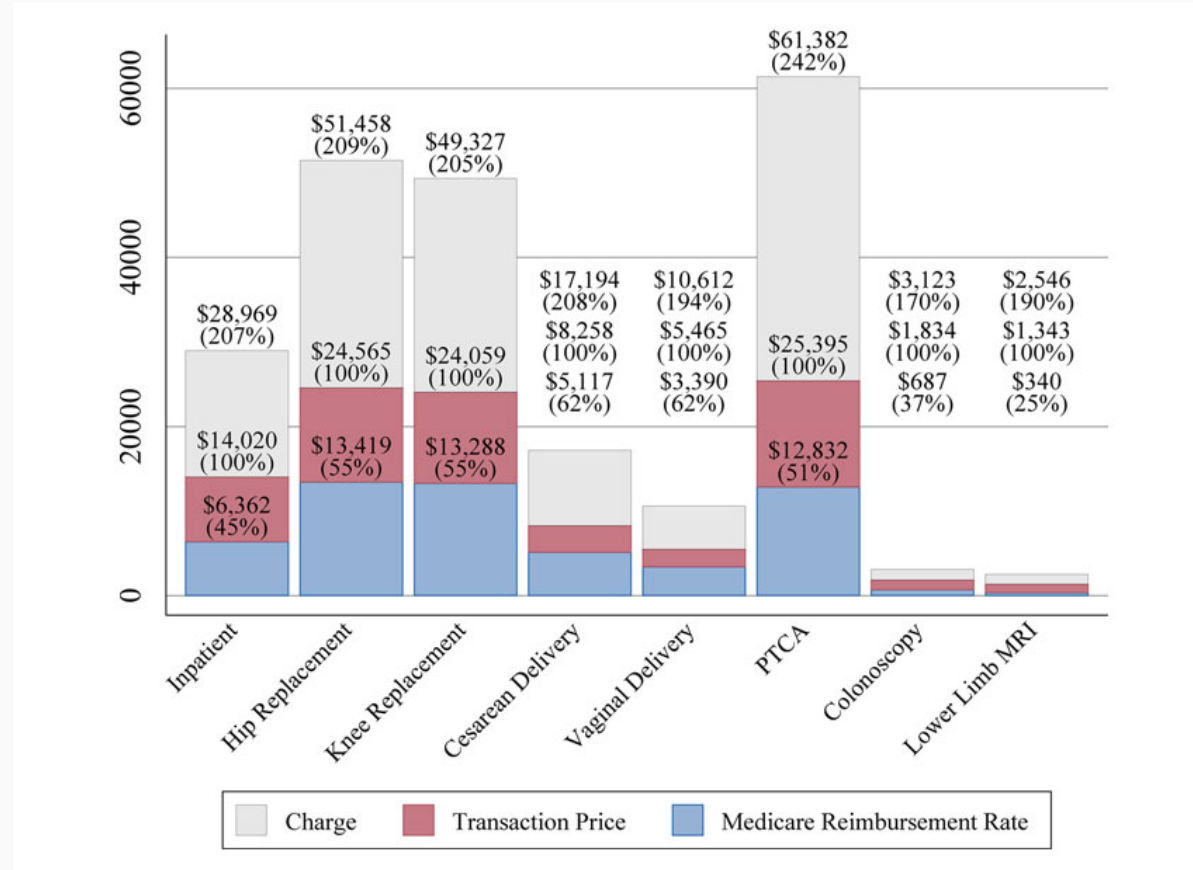
# What is a hospital price?

Lots of different payers paying lots of different prices:

- Medicare fee-for-service prices
- Medicaid payments
- Private insurance negotiations (including Medicare Advantage)
- But what about the price to patients?

Price  $\neq$  charge  $\neq$  cost  $\neq$  patient out-of-pocket spending

# What is a hospital price?



Source: [Health Care Pricing Project](#)

# What is a hospital price?

Not clear what exactly is negotiated...

## Fee-for-service

- price per procedure
- percentage of charges
- markup over Medicare rates

## Capitation

- payment per patient
- pay-for-performance
- shared savings

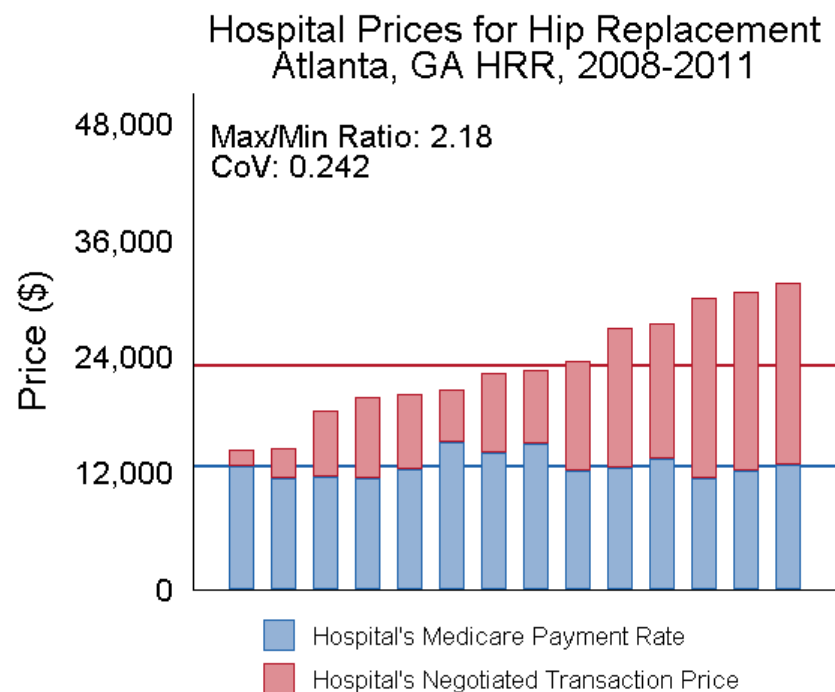
# Hospital prices in real life

A few empirical facts:

1. Hospital services are expensive
2. Prices vary dramatically across different areas
3. Lack of competition is a major reason for high prices

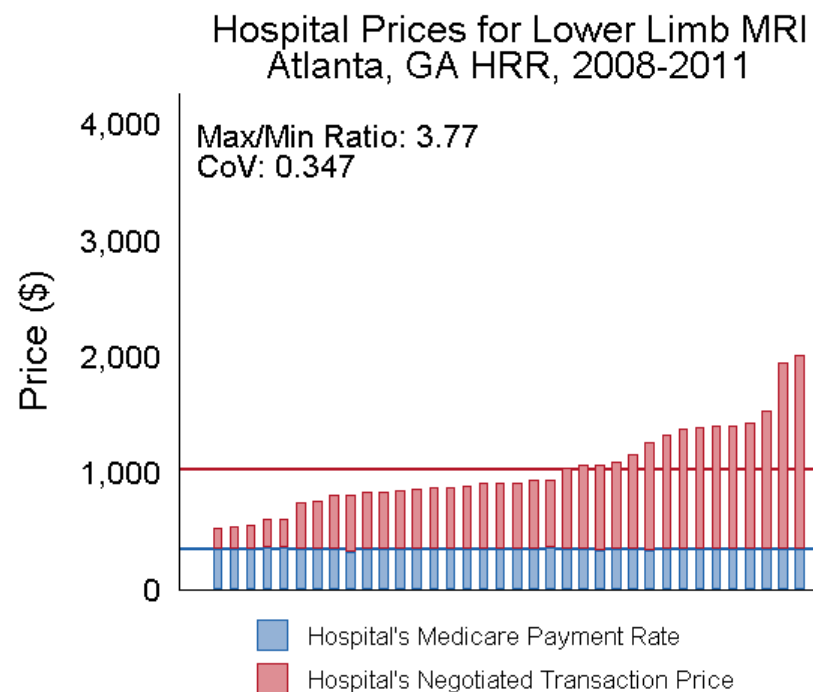


# Hospital prices in real life



**Note:** Each column captures a hospital's negotiated transaction price and Medicare reimbursement. Prices are averaged from 2008-2011 and presented in 2011 dollars. CoV captures the coefficient of variation of hospital negotiated transaction prices within the HRR. Max/Min captures the max/min ratio of hospital's negotiated transaction prices within the HRR. Horizontal lines indicate average rates and prices within the region.

© Health Care Pricing Project



**Note:** Each column captures a hospital's negotiated transaction price and Medicare reimbursement. Prices are averaged from 2008-2011 and presented in 2011 dollars. CoV captures the coefficient of variation of hospital negotiated transaction prices within the HRR. Max/Min captures the max/min ratio of hospital's negotiated transaction prices within the HRR. Horizontal lines indicate average rates and prices within the region.

© Health Care Pricing Project

# Understanding HCRIS Data

---

# What is HCRIS?

Healthcare Cost Report Information System ('cost reports')

- Nursing Homes (SNFs)
- Hospice
- Home Health Agencies
- Hospitals

# Hospital Cost Reports

10-12

FORM CMS-2552-10

4090 (Cont.)

STATEMENT OF PATIENT REVENUES  
AND OPERATING EXPENSES

PROVIDER CCN:

PERIOD:

FROM \_\_\_\_\_  
TO \_\_\_\_\_

WORKSHEET G-2,  
PARTS I & II

## PART I - PATIENT REVENUES

REVENUE CENTER		INPATIENT	OUTPATIENT	TOTAL	
		1	2	3	
<b>GENERAL INPATIENT ROUTINE CARE SERVICES</b>					
1	Hospital				1
2	Subprovider IPF				2
3	Subprovider IRF				3
4	Subprovider (Other)				4
5	Swing bed - SNF				5
6	Swing bed - NF				6
7	Skilled nursing facility				7
8	Nursing facility				8
9	Other long term care				9
10	Total general inpatient care services (sum of lines 1-9)				10
<b>INTENSIVE CARE TYPE INPATIENT HOSPITAL SERVICES</b>					
11	Intensive care unit				11
12	Coronary care unit				12
13	Burn intensive care unit				13
14	Surgical intensive care unit				14
15	Other special care (specify)				15
16	Total intensive care type inpatient hospital services (sum of lines 11-15)				16
17	Total inpatient routine care services (sum of lines 10 and 16)				17
18	Ancillary services				18
19	Outpatient services				19
20	Rural Health Clinic (RHC)				20
21	Federally Qualified Health Center (FQHC)				21
22	Home health agency				22

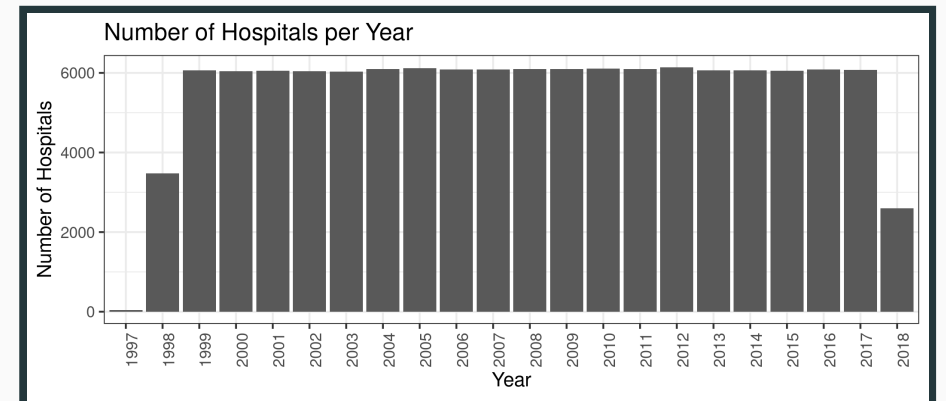
# The Data

Let's work with the [HCRIS GitHub repository](#). But forming the dataset is up to you this time.

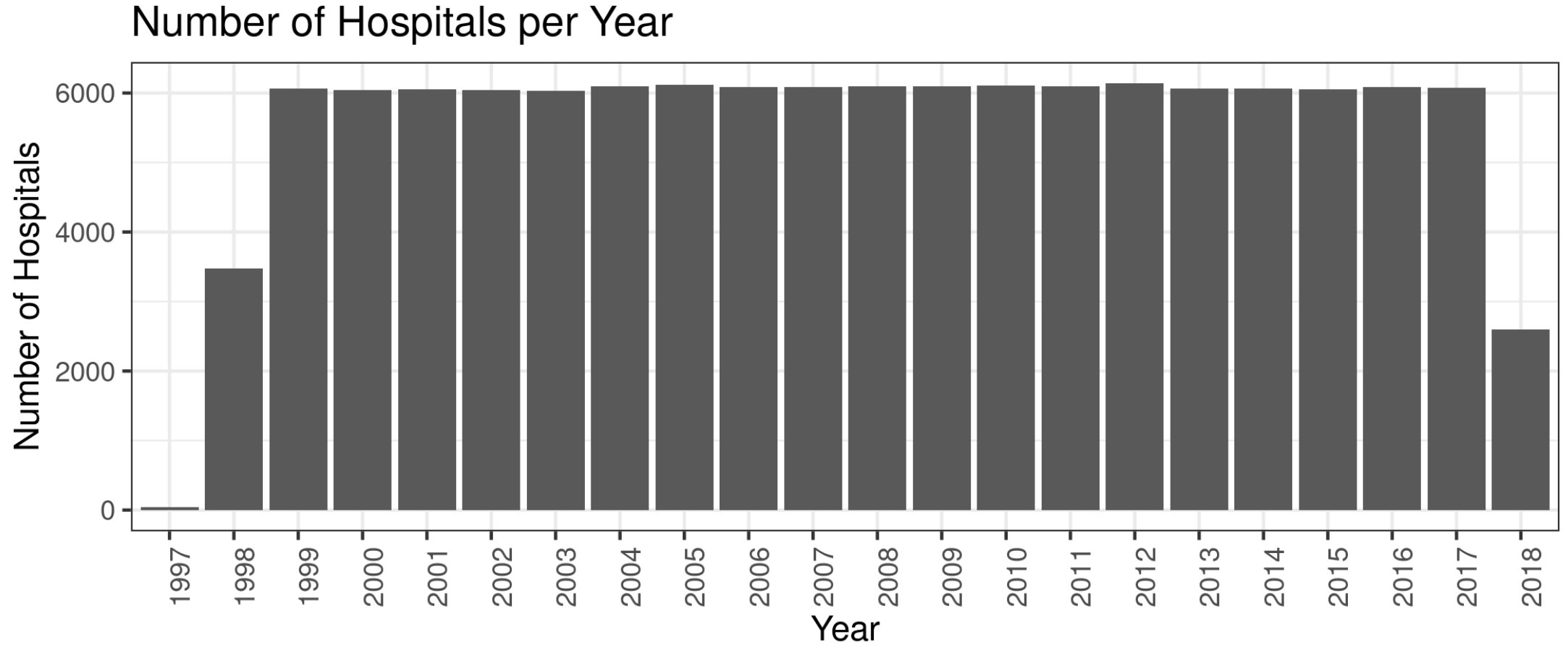


# The Data

```
hcris.data %>%  
  ggplot(aes(x=as.factor(year))) +  
  geom_bar() +  
  labs(  
    x="Year",  
    y="Number of Hospitals",  
    title="Number of Hospitals per Year"  
  ) + theme_bw() +  
  theme(axis.text.x = element_text(angle = 90, hjust=1))
```



# Number of hospitals



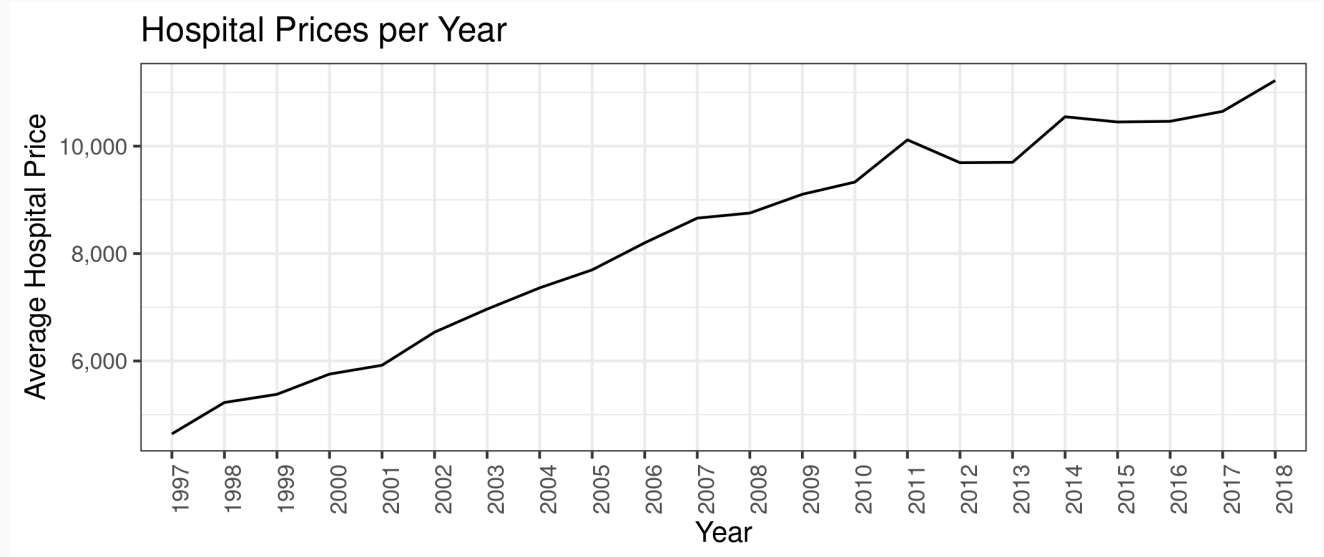
# Estimating hospital prices

```
hcris.data <- hcris.data %>%  
  mutate( discount_factor = 1-tot_discounts/tot_charges,  
          price_num = (ip_charges + icu_charges + ancillary_charges)*discount_factor - tot_mcare_payment,  
          price_denom = tot_discharges - mcare_discharges,  
          price = price_num/price_denom)
```



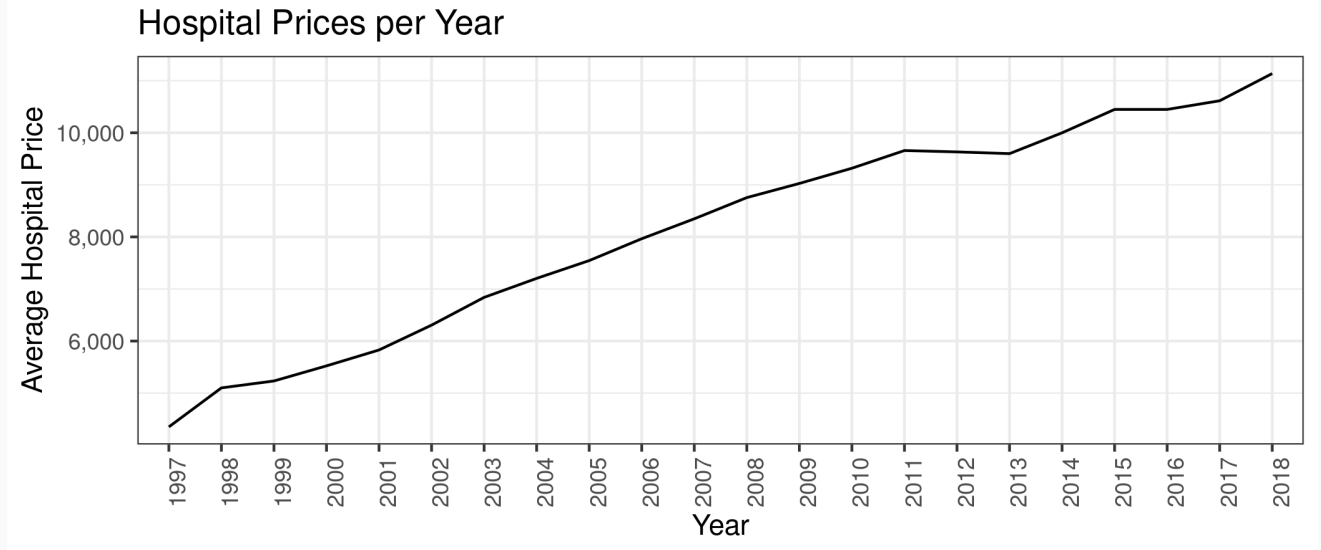
# Estimating hospital prices

```
hcris.data %>% group_by(year) %>%  
  filter(price_denom>10, !is.na(price_denom),  
         price_num>0, !is.na(price_num)) %>%  
  select(price, year) %>%  
  summarize(mean_price=mean(price, na.rm=TRUE))  
ggplot(aes(x=as.factor(year), y=mean_price))  
  geom_line(aes(group=1)) +  
  labs(  
    x="Year",  
    y="Average Hospital Price",  
    title="Hospital Prices per Year"  
  ) + scale_y_continuous(labels=comma) +  
  theme_bw() + theme(axis.text.x = element_te
```



# Estimating hospital prices

```
hcris.data %>% group_by(year) %>%  
  filter(price_denom>100, !is.na(price_denom),  
         price_num>0, !is.na(price_num),  
         price<100000) %>%  
  select(price, year) %>%  
  summarize(mean_price=mean(price, na.rm=TRUE))  
ggplot(aes(x=as.factor(year), y=mean_price))  
  geom_line(aes(group=1)) +  
  labs(  
    x="Year",  
    y="Average Hospital Price",  
    title="Hospital Prices per Year"  
  ) + scale_y_continuous(labels=comma) +  
  theme_bw() + theme(axis.text.x = element_te
```



# Potential Outcomes Framework

---

# What is a "Potential Outcome"



$$Y(1) = \$75,000$$



$$Y(0) = \$60,000$$

---

$$\text{Treatment effect} = Y(1) - Y(0) = \$15,000$$

# What is a "Potential Outcome"



$$Y(1) = \$75,000$$



$$Y(0) = ?$$

---

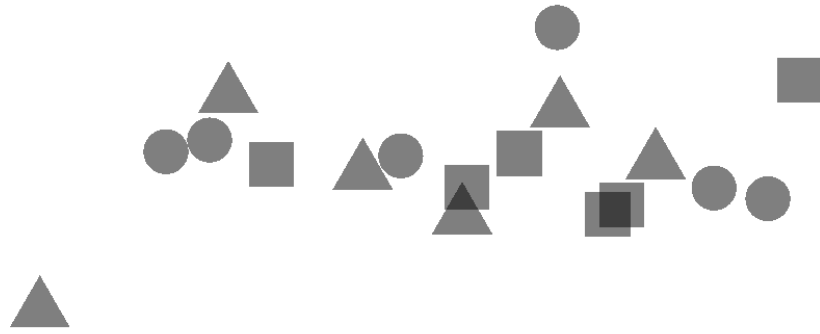
$$\text{Treatment effect} = Y(1) - Y(0) = ?$$

# Do we ever observe the potential outcomes?

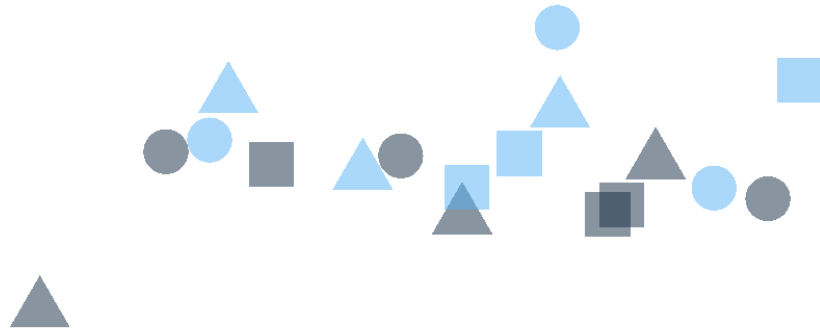


Without a time machine...hard to get *individual* treatment effects.

# Average Treatment Effect



# Average Treatment Effect





# Average Treatment Effect



# Average Treatment Effect

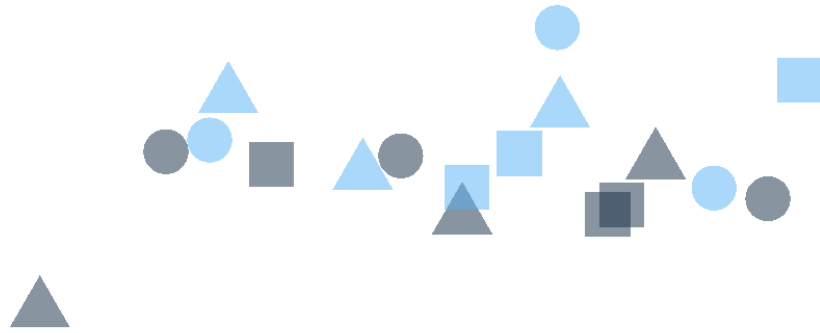


$$ATE = E[Y(1) - Y(0)]$$

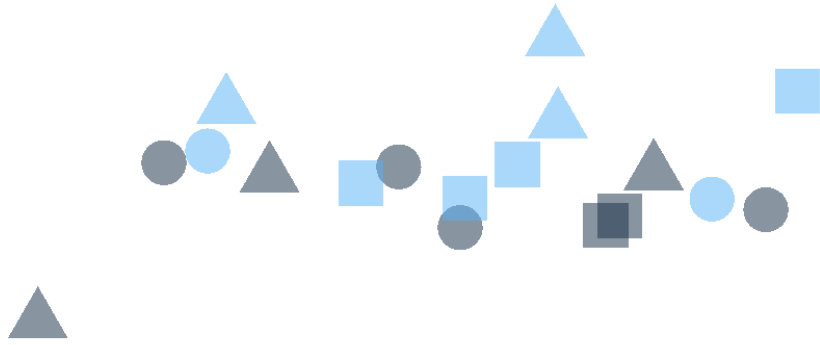
# Selection on Observables

---

# Average Treatment Effect



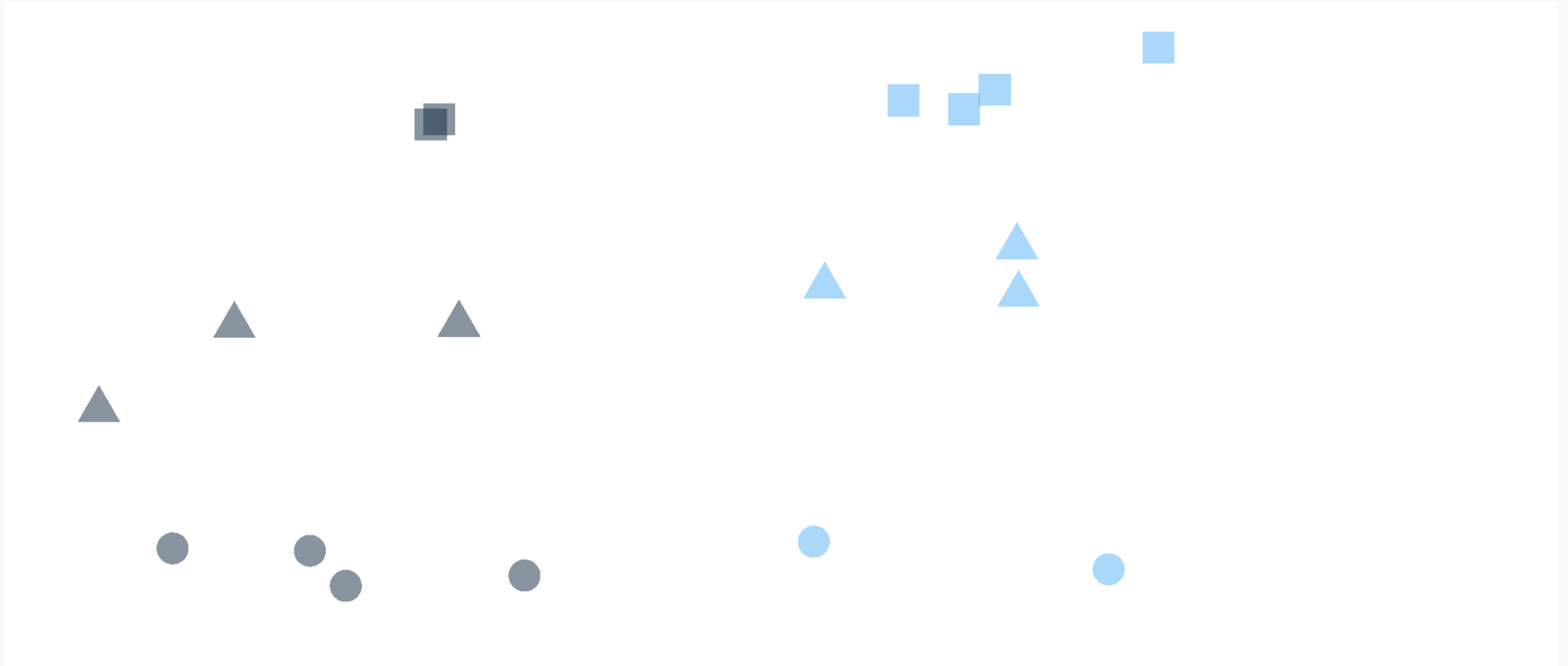
# ...with Selection



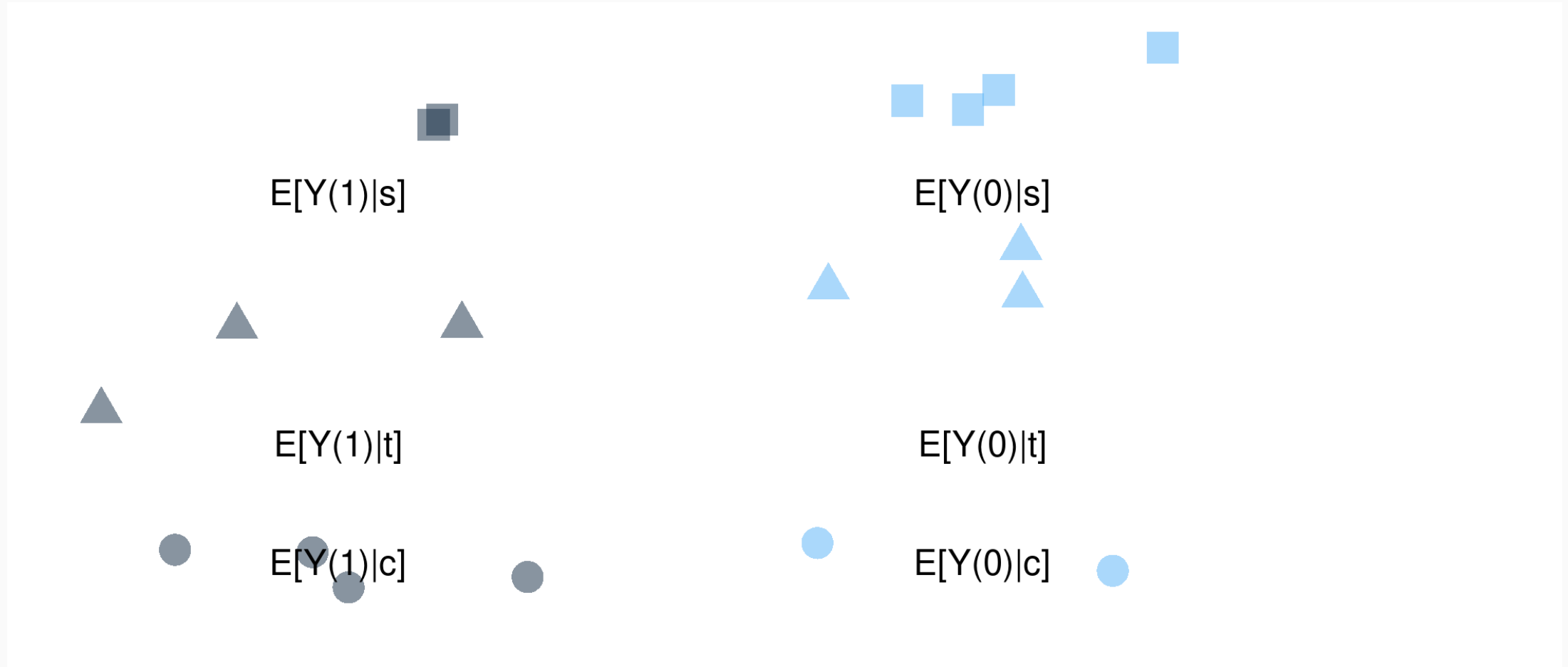
# ...with Selection



# ...with Selection

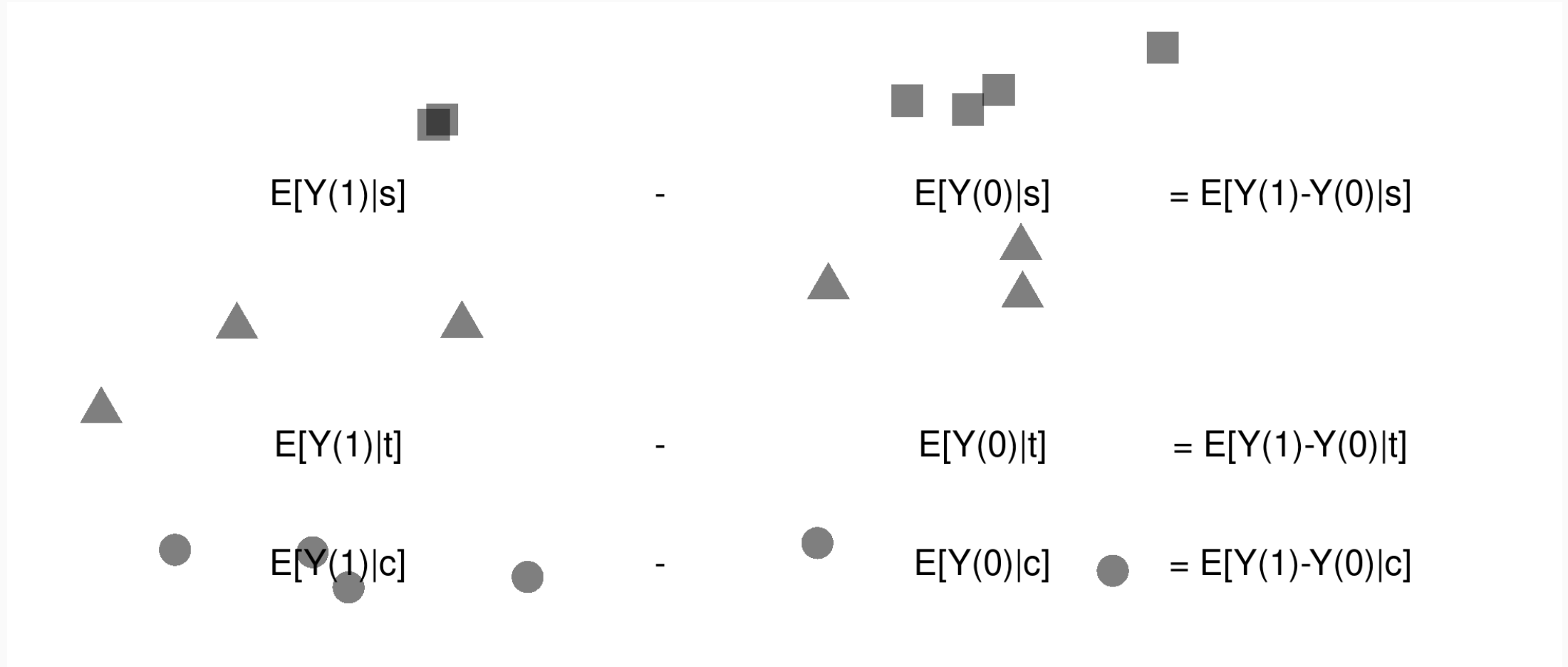


# ...with Selection

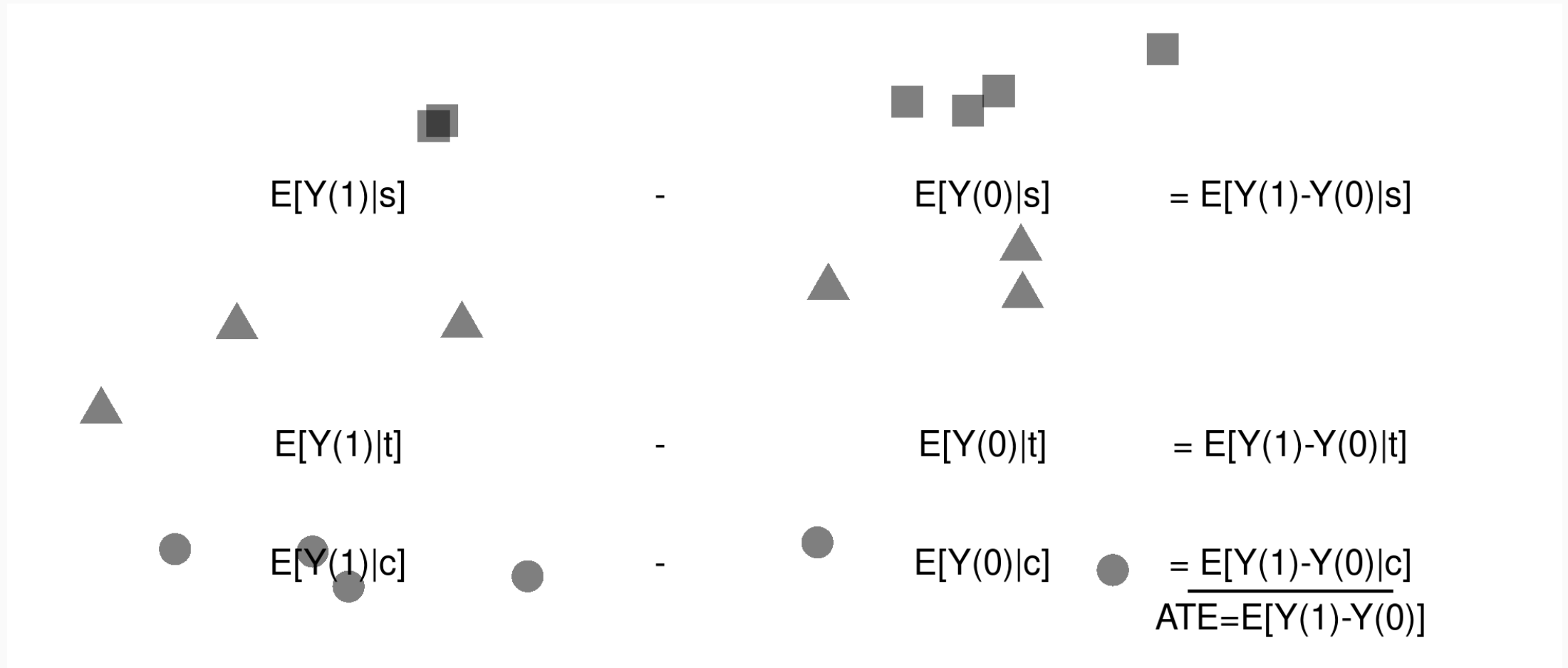




# Assumption 1: Selection on Observables



# Assumption 1: Selection on Observables



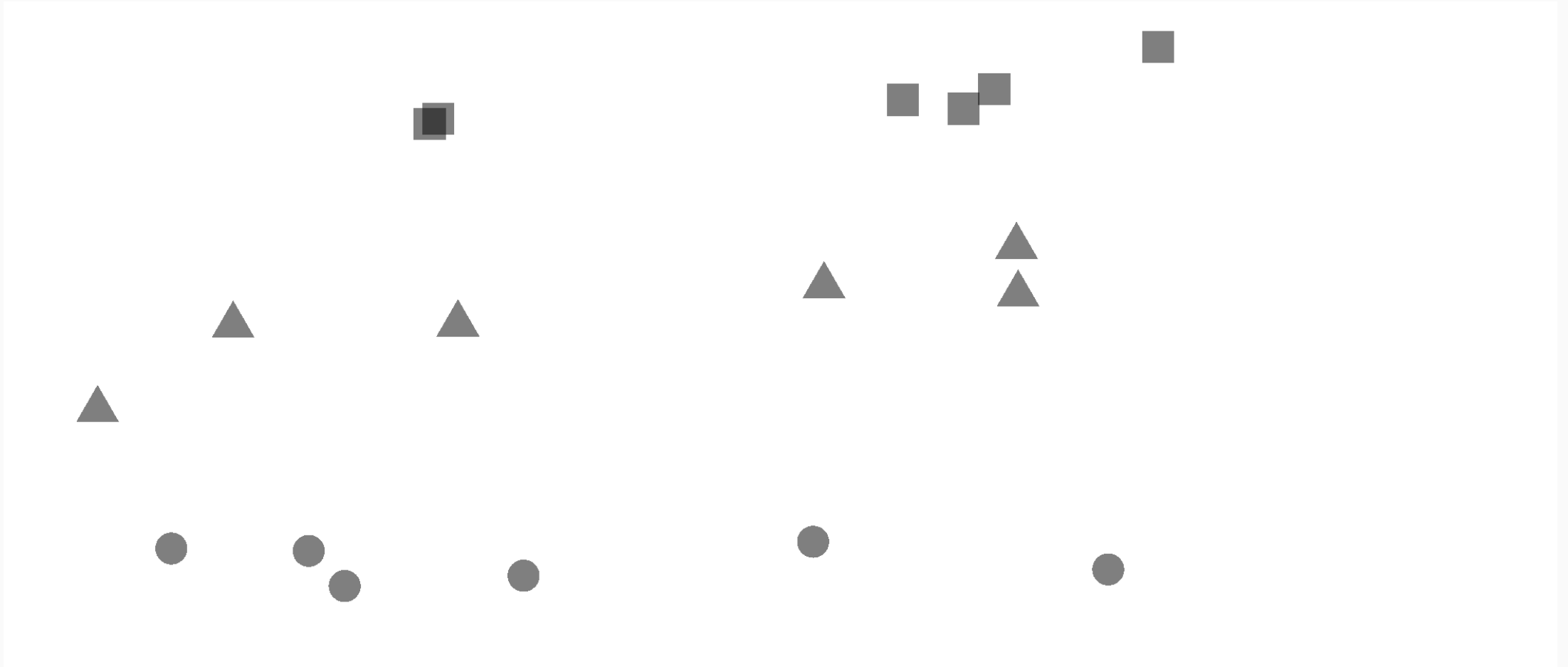
# Assumption 1: Selection on Observables

More formally:

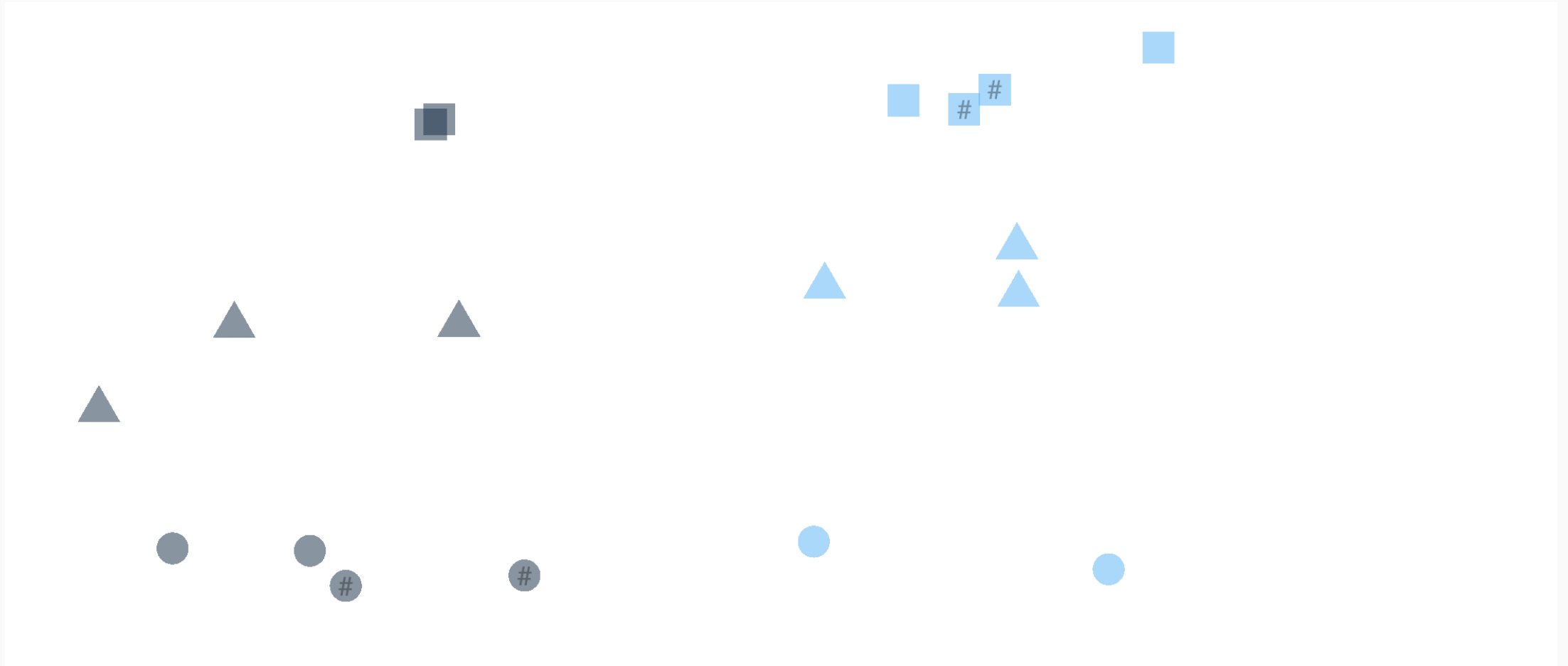
- $E[Y(1)|W, shape] = E[Y(1)|shape]$
- $Y(1), Y(0) \perp\!\!\!\perp W | shape$

In words...nothing unobserved that determines treatment selection and affects your outcome of interest.

# Violation of Selection on Observables



# Violation of Selection on Observables

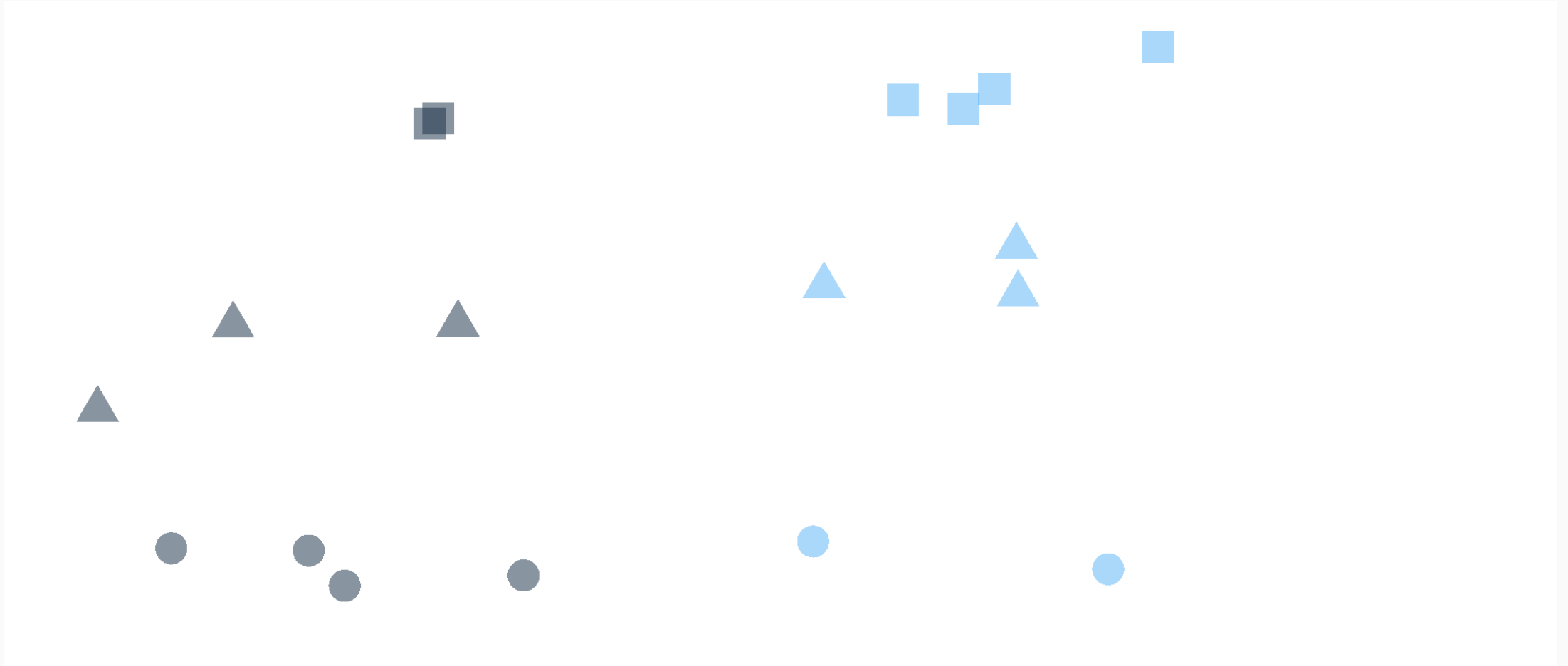


# Assumption 2: Common Support

Someone of each type must be in both the treated and untreated groups

$$0 < \Pr(W = 1|X) < 1$$

# Assumption 2: Common Support



# Violation of Common Support





# Causal Inference with Observational Data

---

# Fundamental Problem of Causal Inference

Potential outcomes:

- $Y_i(1)$  for  $W_i = 1$
- $Y_i(0)$  for  $W_i = 0$

Average treatment effect is  $E[Y_i(1) - Y_i(0)]$ , but:

- $E[Y_i | W_i = 1] \neq E[Y_i(1)]$
- $E[Y_i | W_i = 0] \neq E[Y_i(0)]$

# Fundamental Problem of Causal Inference

- We don't observe the counterfactual outcome...what would have happened if a treated unit was actually untreated.
- *ALL* attempts at causal inference represent some attempt at estimating the counterfactual outcome. We need an estimate for  $E[Y_i(0)]$  among those that were treated, and vice versa for  $E[Y_i(1)]$ .

# Causal inference with observational data

Solution for now: find covariates  $\mathbf{X}_i$  such that the following assumptions are plausible:

1. Selection on observables:

$$Y_i(1), Y_i(0) \perp\!\!\!\perp W_i | \mathbf{X}_i$$

2. Common support:

$$0 < \Pr(W_i = 1 | \mathbf{X}_i) < 1$$

# Causal inference with observational data

With selection on observables and common support:

1. Matching estimators:

$$E[Y_i(1) - Y_i(0)] = E[E[Y_i|W_i = 1, X_i] - E[Y_i|W_i = 0, X_i]]$$

2. Regression estimators:

$$E[Y_i(1) - Y_i(0)] = E[E[Y_i|W_i = 1, X_i]] - E[E[Y_i|W_i = 0, X_i]]$$

What's the difference?

# Matching

$E[Y(1)|s]$

-

$E[Y(0)|s]$

$= E[Y(1)-Y(0)|s]$

$E[Y(1)|t]$

-

$E[Y(0)|t]$

$= E[Y(1)-Y(0)|t]$

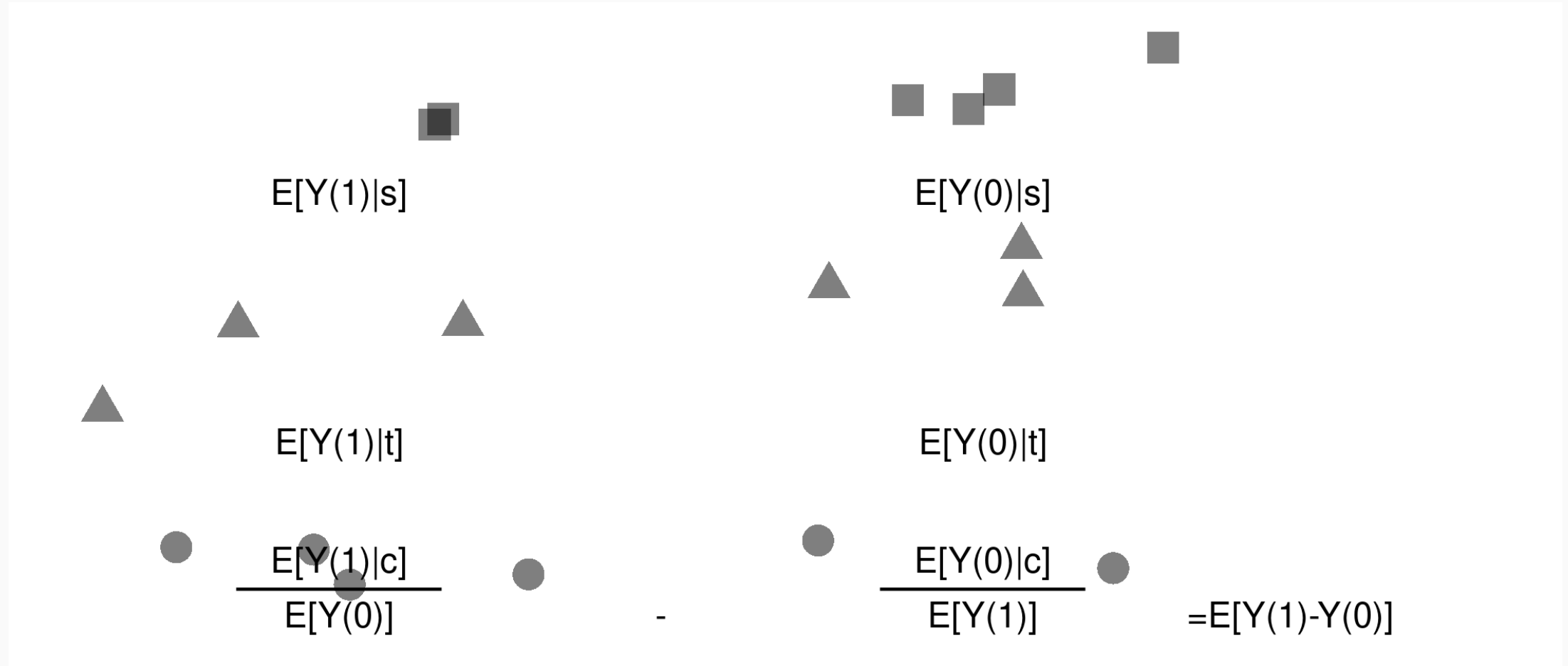
$E[Y(1)|c]$

-

$E[Y(0)|c]$

$= \frac{E[Y(1)-Y(0)|c]}{ATE=E[Y(1)-Y(0)]}$

# Regression



# Estimation options

- Matching
- Weighting
- Regression
- Doubly-robust weighting + regression (won't cover)



# Matching: The process

1. For each observation  $i$ , find the  $m$  "nearest" neighbors,  $J_m(i)$ .
2. Impute  $\hat{Y}_i(0)$  and  $\hat{Y}_i(1)$  for each observation:

$$\hat{Y}_i(0) = \begin{cases} Y_i & \text{if } W_i = 0 \\ \frac{1}{m} \sum_{j \in J_m(i)} Y_j & \text{if } W_i = 1 \end{cases}$$

$$\hat{Y}_i(1) = \begin{cases} Y_i & \text{if } W_i = 1 \\ \frac{1}{m} \sum_{j \in J_m(i)} Y_j & \text{if } W_i = 0 \end{cases}$$

3. Form "matched" ATE:

$$\hat{\delta}^{\text{match}} = \frac{1}{N} \sum_{i=1}^N \left( \hat{Y}_i(1) - \hat{Y}_i(0) \right)$$

# Matching: Defining "nearest"

1. Euclidean distance:

$$\sum_{k=1}^K (X_{ik} - X_{jk})^2$$

2. Scaled Euclidean distance:

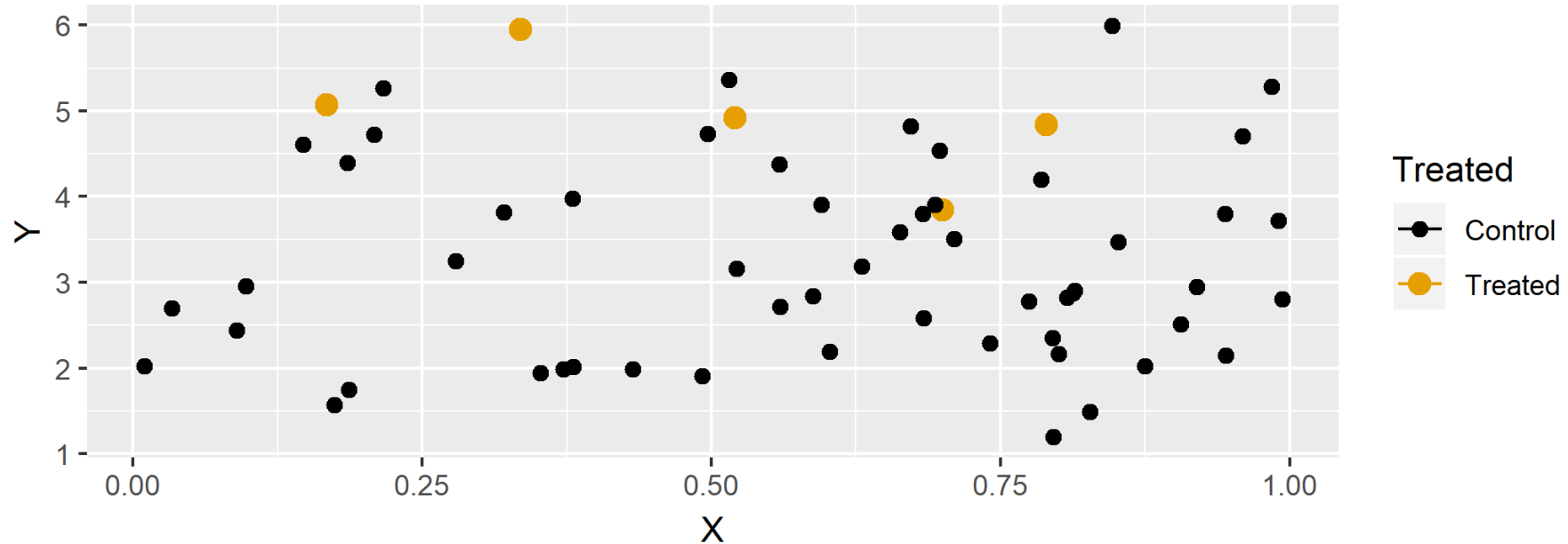
$$\sum_{k=1}^K \frac{1}{\sigma_{X_k}^2} (X_{ik} - X_{jk})^2$$

3. Mahalanobis distance:

$$(X_i - X_j)' \Sigma_X^{-1} (X_i - X_j)$$

# Animation for matching

The Effect of Treatment on Y while Matching on X (with a caliper)  
1. Start with raw data.



# Weighting

1. Estimate propensity score `ps ← glm(W~X, family=binomial, data)`, denoted  $\hat{\pi}(X_i)$
2. Weight by inverse of propensity score

$$\hat{\mu}_1 = \frac{\sum_{i=1}^N \frac{Y_i W_i}{\hat{\pi}(X_i)}}{\sum_{i=1}^N \frac{W_i}{\hat{\pi}(X_i)}} \text{ and } \hat{\mu}_0 = \frac{\sum_{i=1}^N \frac{Y_i (1-W_i)}{1-\hat{\pi}(X_i)}}{\sum_{i=1}^N \frac{1-W_i}{1-\hat{\pi}(X_i)}}$$

3. Form "inverse-propensity weighted" ATE:

$$\hat{\delta}^{IPW} = \hat{\mu}_1 - \hat{\mu}_0$$

# Regression

1. Regress  $Y_i$  on  $X_i$  among  $W_i = 1$  to form  $\hat{\mu}_1(X_i)$
2. Regress  $Y_i$  on  $X_i$  among  $W_i = 0$  to form  $\hat{\mu}_0(X_i)$
3. Form difference in predictions:

$$\hat{\delta}^{reg} = \frac{1}{N} \sum_{i=1}^N (\hat{\mu}_1(X_i) - \hat{\mu}_0(X_i))$$

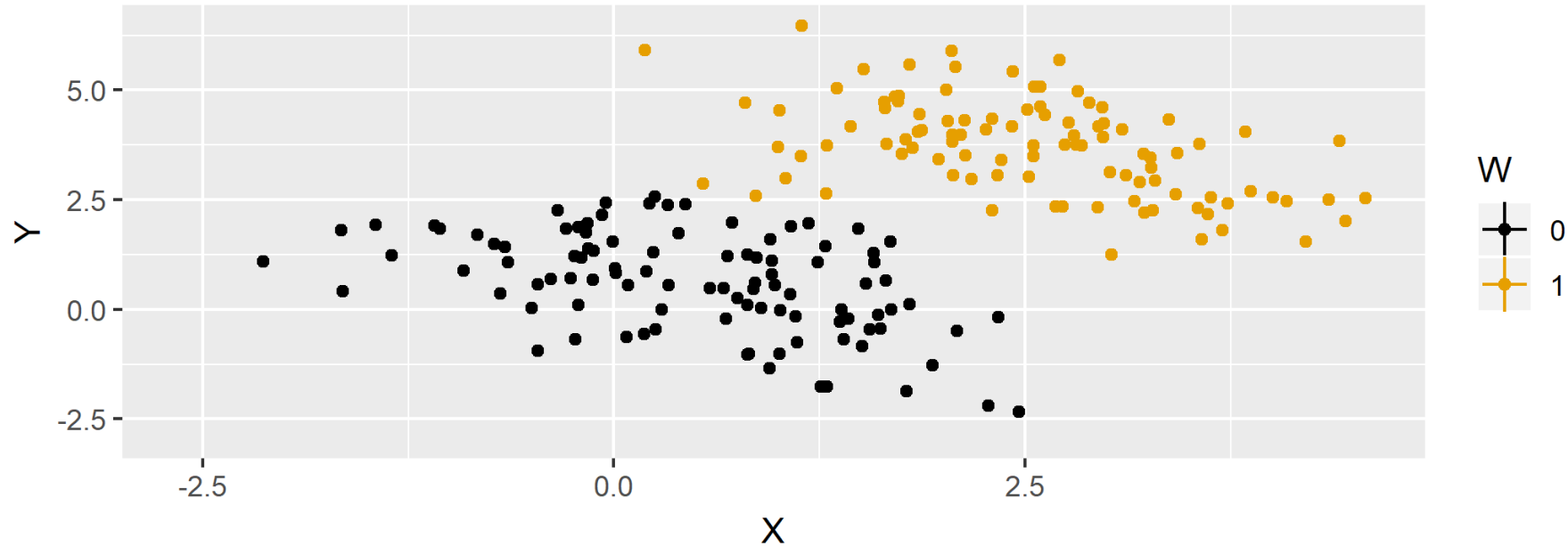
# Regression

Or estimate in one step,

$$Y_i = \delta W_i + \beta X_i + W_i \times (X_i - \bar{X}) \gamma + \varepsilon_i$$

# Animation for regression

The Relationship between Y and X, Controlling for a Binary Variable W  
1. Start with raw data. Correlation between X and Y: 0.403



# Simulated data

Now let's do some matching, re-weighting, and regression with simulated data:

```
n ← 5000
select.dat ← tibble(
  x = runif(n, 0, 1),
  z = rnorm(n, 0, 1),
  w = (x > 0.65),
  y = -2.5 + 4*w + 1.5*x + rnorm(n, 0, 1),
  w_alt = (x + z > 0.35),
  y_alt = -2.5 + 4*w_alt + 1.5*x + 2.25*z + rnorm(n, 0, 1)
)
```



# Simulation: nearest neighbor matching

```
nn.est1 ← Matching::Match(Y=select.dat$y,  
                           Tr=select.dat$w,  
                           X=select.dat$x,  
                           M=1,  
                           Weight=1,  
                           estimand="ATE")
```

```
summary(nn.est1)
```

```
##  
## Estimate ...    4.0175  
## AI SE.....   0.52954  
## T-stat.....   7.5869  
## p.val.....    3.2863e-14  
##  
## Original number of observations..... 5000  
## Original number of treated obs..... 1732  
## Matched number of observations..... 5000  
## Matched number of observations (unweighted). 5016
```

# Simulation: nearest neighbor matching

```
nn.est2 ← Matching::Match(Y=select.dat$y,  
                           Tr=select.dat$w,  
                           X=select.dat$x,  
                           M=1,  
                           Weight=2,  
                           estimand="ATE")
```

```
summary(nn.est2)
```

```
##  
## Estimate ...    4.0175  
## AI SE.....   0.52954  
## T-stat.....   7.5869  
## p.val.....    3.2863e-14  
##  
## Original number of observations..... 5000  
## Original number of treated obs..... 1732  
## Matched number of observations..... 5000  
## Matched number of observations (unweighted). 5016
```

# Simulation: regression

```
reg1.dat ← select.dat %>% filter(w==1)
reg1 ← lm(y ~ x, data=reg1.dat)

reg0.dat ← select.dat %>% filter(w==0)
reg0 ← lm(y ~ x, data=reg0.dat)
pred1 ← predict(reg1,new=select.dat)
pred0 ← predict(reg0,new=select.dat)
mean(pred1-pred0)
```

```
## [1] 4.076999
```

# Violation of selection on observables

## NN Matching

```
nn.est3 ← Matching::Match(Y=select.dat$y_alt,  
                           Tr=select.dat$w_alt,  
                           X=select.dat$x,  
                           M=1,  
                           Weight=2,  
                           estimand="ATE")  
  
summary(nn.est3)
```

```
##  
## Estimate... 7.6642  
## AI SE..... 0.052903  
## T-stat..... 144.87  
## p.val..... < 2.22e-16  
##  
## Original number of observations..... 5000  
## Original number of treated obs..... 2748  
## Matched number of observations..... 5000  
## Matched number of observations (unweighted). 23014
```

## Regression

```
reg1.dat ← select.dat %>% filter(w_alt=1)  
reg1 ← lm(y_alt ~ x, data=reg1.dat)  
  
reg0.dat ← select.dat %>% filter(w_alt=0)  
reg0 ← lm(y_alt ~ x, data=reg0.dat)  
pred1_alt ← predict(reg1,new=select.dat)  
pred0_alt ← predict(reg0,new=select.dat)  
mean(pred1_alt-pred0_alt)
```

```
## [1] 7.646532
```

# Pricing and Hospital Profit Status

---

# Penalized hospitals

```
final.hcris <- hcris.data %>% ungroup() %>%  
  filter(price_denom>100, !is.na(price_denom),  
         price_num>0, !is.na(price_num),  
         price<1000000,  
         beds>30, year=2012) %>%  
  mutate( hvbp_payment = ifelse(is.na(hvbp_payment),0,hvbp_payment),  
         hrrp_payment = ifelse(is.na(hrrp_payment),0,abs(hrrp_payment)),  
         penalty = (hvbp_payment-hrrp_payment<0))
```

# Summary stats

Always important to look at your data before doing any formal analysis. Ask yourself a few questions:

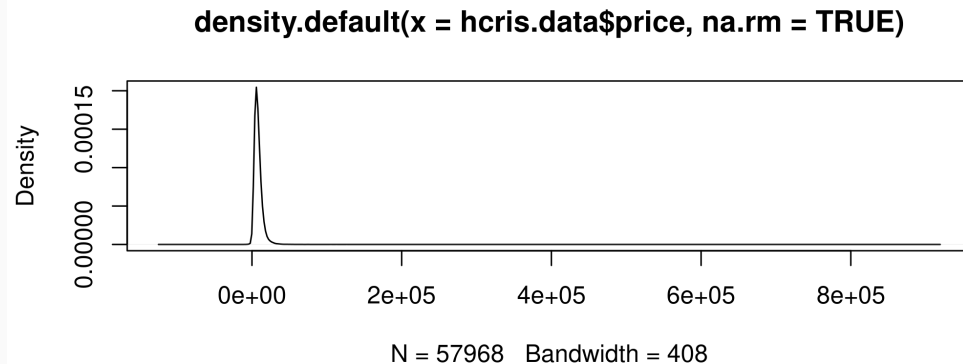
1. Are the magnitudes reasonable?
2. Are there lots of missing values?
3. Are there clear examples of misreporting?

# Summary stats

```
summary(hcris.data$price)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
##	-123697	4783	7113	Inf	10230	Inf	63662

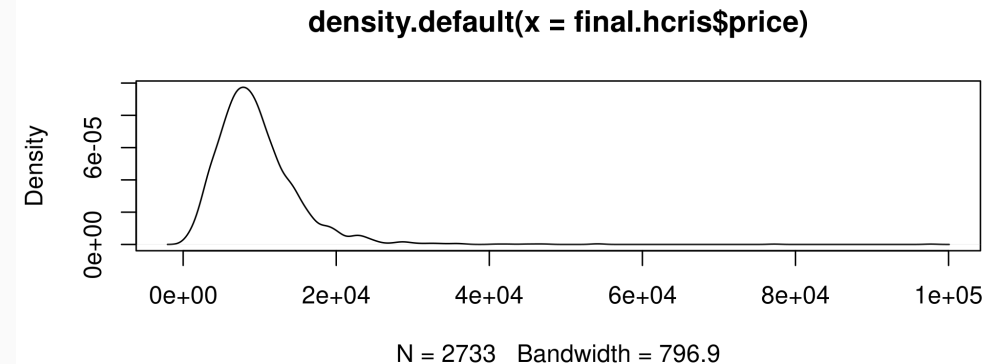
```
plot(density(hcris.data$price, na.rm=TRUE))
```



```
summary(final.hcris$price)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	340.8	6129.9	8705.4	9646.9	11905.4	97688.8

```
plot(density(final.hcris$price))
```





# Dealing with problems

We've adopted a very brute force way to deal with outlier prices. Other approaches include:

1. Investigate very closely the hospitals with extreme values
2. Winsorize at certain thresholds (replace extreme values with pre-determined thresholds)
3. Impute prices for extreme hospitals

# Differences among penalized hospitals

- Mean price among penalized hospitals: 9,896.31
- Mean price among non-penalized hospitals: 9,560.41
- Mean difference: 335.9

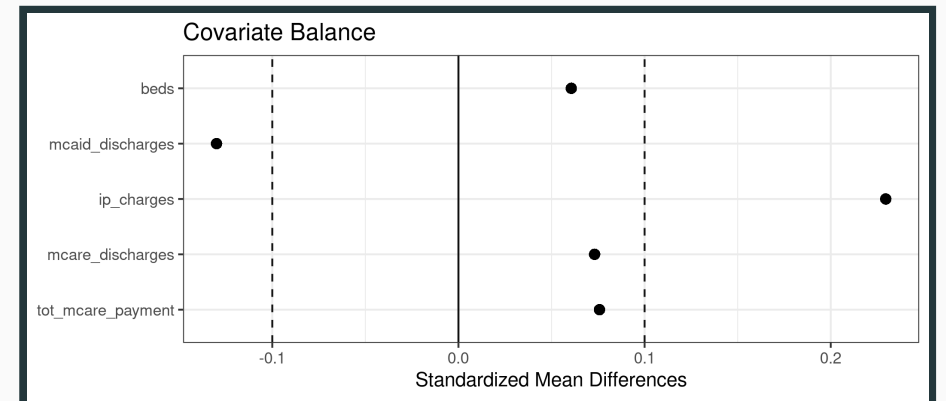
# Comparison of hospitals

Are penalized hospitals sufficiently similar to non-penalized hospitals?

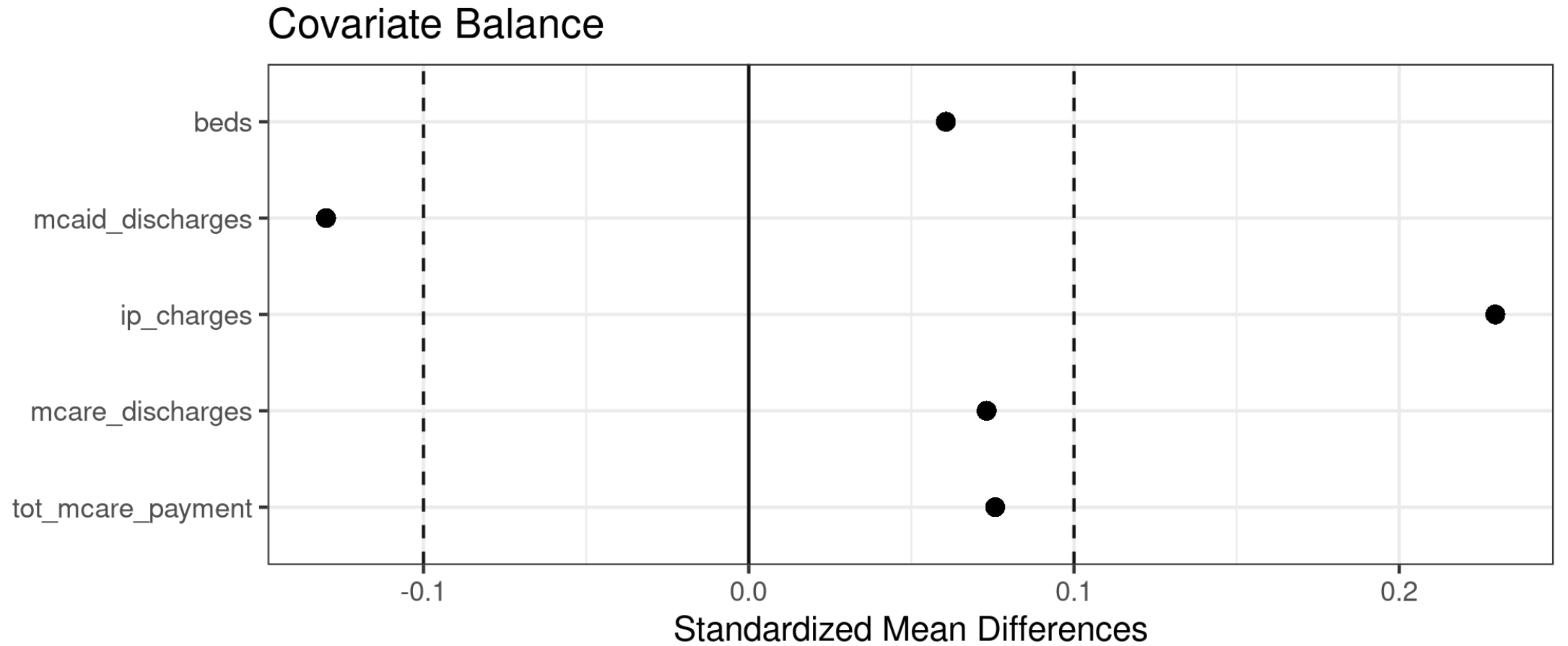
Let's look at covariate balance using a love plot, part of the `library(cobalt)` package.

# Love plots without adjustment

```
love.plot(bal.tab(lp.covs,treat=lp.vars$penalty), colors="black", shapes="circle", threshold=0.1) +  
  theme_bw() + theme(legend.position="none")
```



# Love plots without adjustment



# Using matching to improve balance

Some things to think about:

- exact versus nearest neighbor
- with or without ties (and how to break ties)
- measure of distance

# 1. Exact Matching

```
m.exact ← Matching::Match(Y=lp.vars$price,  
                          Tr=lp.vars$penalty,  
                          X=lp.covs,  
                          M=1,  
                          exact=TRUE)  
print(m.exact)
```

```
## [1] NA  
## attr(,"class")  
## [1] "Match"
```

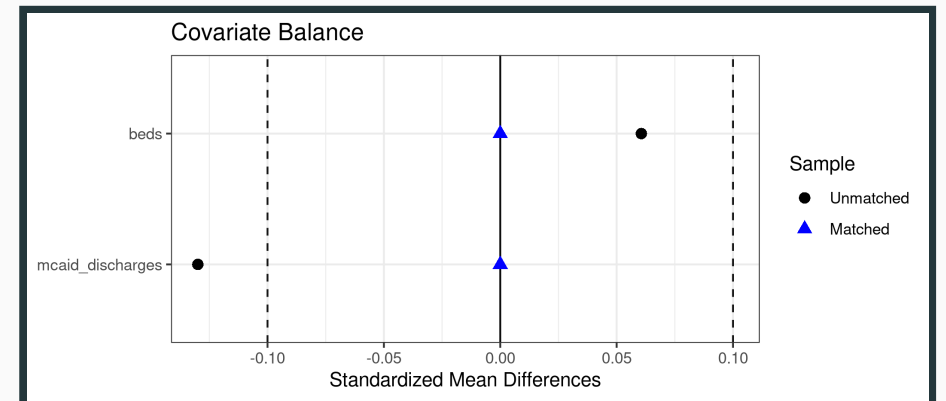
# 1. Exact Matching (on a subset)

```
lp.covs2 ← lp.covs %>% select(beds, mcaid_discharges)
m.exact ← Matching::Match(Y=lp.vars$price,
                          Tr=lp.vars$penalty,
                          X=lp.covs2,
                          M=1,
                          exact=TRUE,
                          estimand="ATE")
```

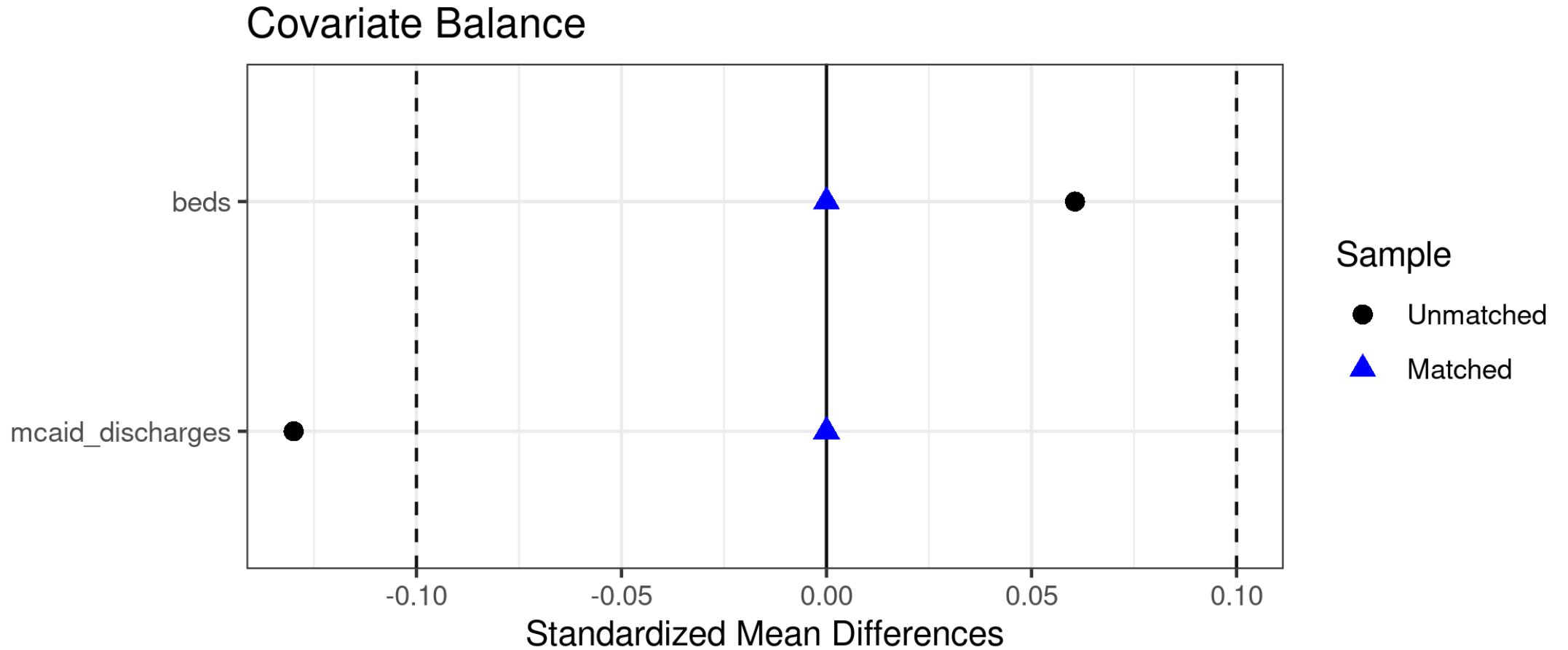


# 1. Exact Matching (on a subset)

```
love.plot(bal.tab(m.exact, covs = lp.covs2, treat = lp.vars$penalty),  
  threshold=0.1,  
  grid=FALSE, sample.names=c("Unmatched", "Matched"),  
  position="top", shapes=c("circle", "triangle"),  
  colors=c("black", "blue")) +  
  theme_bw()
```



# 1. Exact Matching (on a subset)

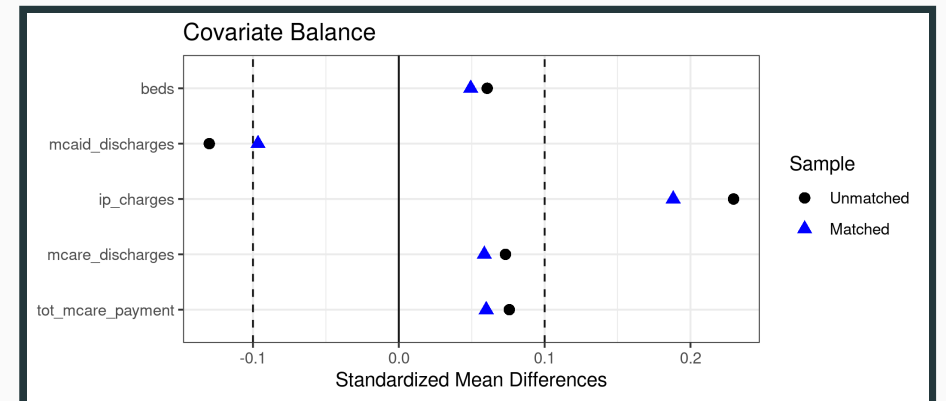


## 2. Nearest neighbor matching (inverse variance)

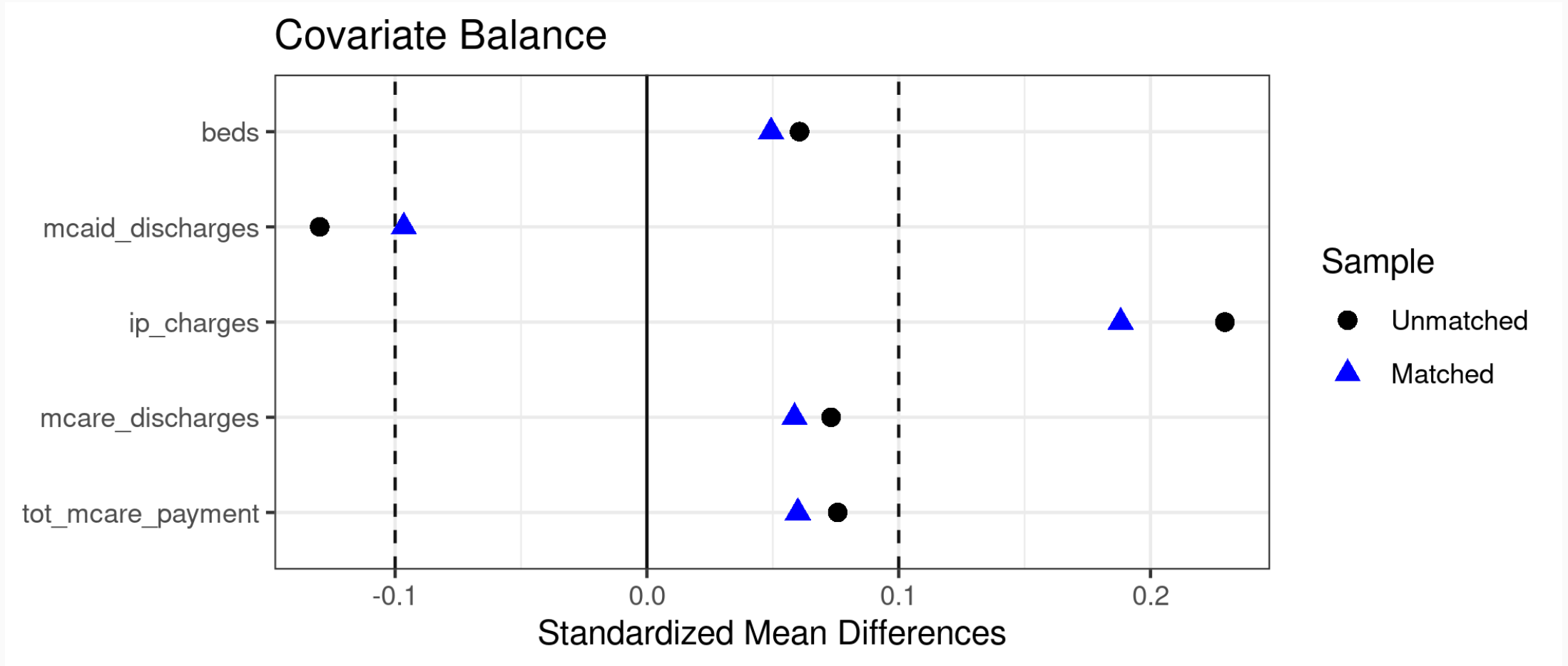
```
m.nn.var ← Matching::Match(Y=lp.vars$price,  
                           Tr=lp.vars$penalty,  
                           X=lp.covs,  
                           M=4,  
                           Weight=1,  
                           estimand="ATE")  
  
v.name=data.frame(new=c("Beds", "Medicaid Discharges", "Inpatient Charges",  
                        "Medicare Discharges", "Medicare Payments"))
```

## 2. Nearest neighbor matching (inverse variance)

```
love.plot(bal.tab(m.nn.var, covs = lp.covs, treat = lp.vars$penalty),  
          threshold=0.1,  
          var.names=v.name,  
          grid=FALSE, sample.names=c("Unmatched", "Matched"),  
          position="top", shapes=c("circle", "triangle"),  
          colors=c("black", "blue")) +  
theme_bw()
```



## 2. Nearest neighbor matching (inverse variance)

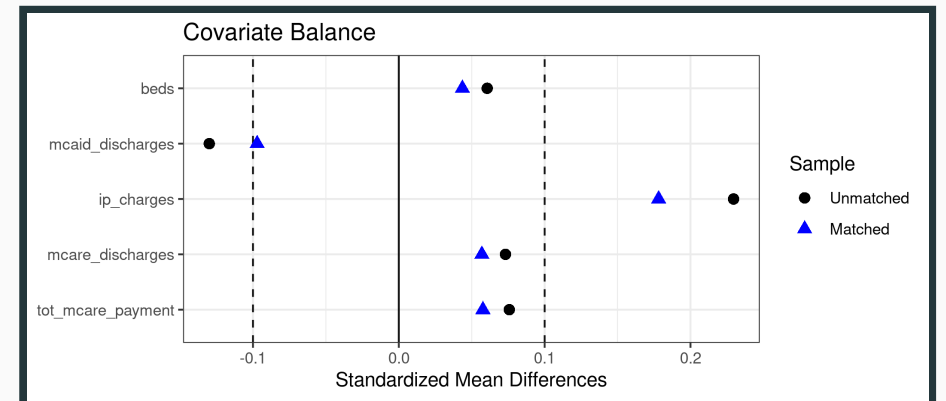


## 2. Nearest neighbor matching (inverse variance)

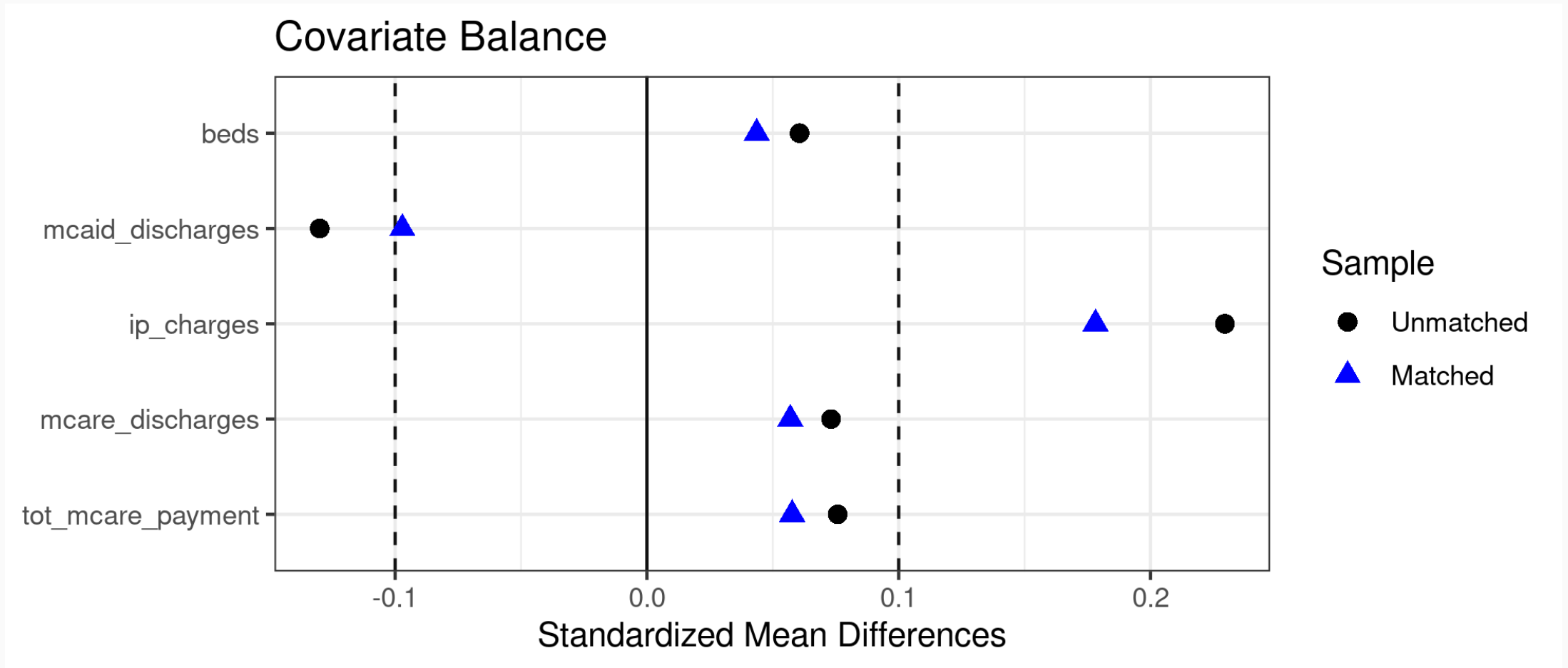
```
m.nn.var2 <- Matching::Match(Y=lp.vars$price,  
                             Tr=lp.vars$penalty,  
                             X=lp.covs,  
                             M=1,  
                             Weight=1,  
                             estimand="ATE")
```

## 2. Nearest neighbor matching (inverse variance)

```
love.plot(bal.tab(m.nn.var2, covs = lp.covs, treat = lp.vars$penalty),
  threshold=0.1,
  var.names=v.name,
  grid=FALSE, sample.names=c("Unmatched", "Matched"),
  position="top", shapes=c("circle","triangle"),
  colors=c("black","blue")) +
  theme_bw()
```



## 2. Nearest neighbor matching (inverse variance)



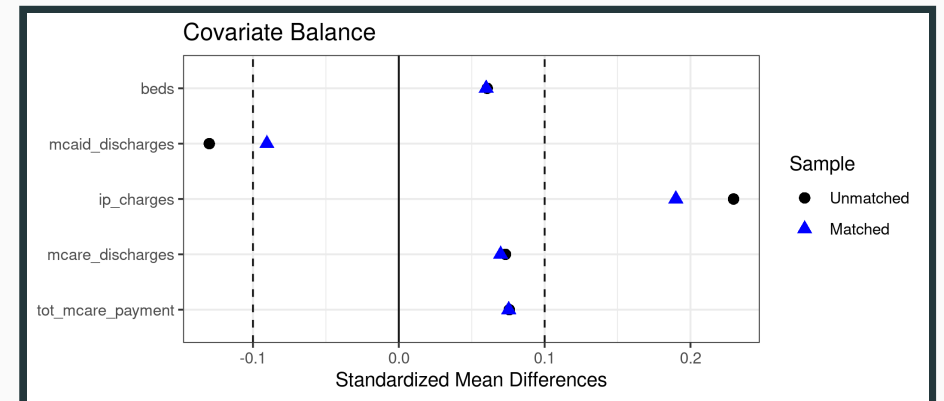


## 2. Nearest neighbor matching (Mahalanobis)

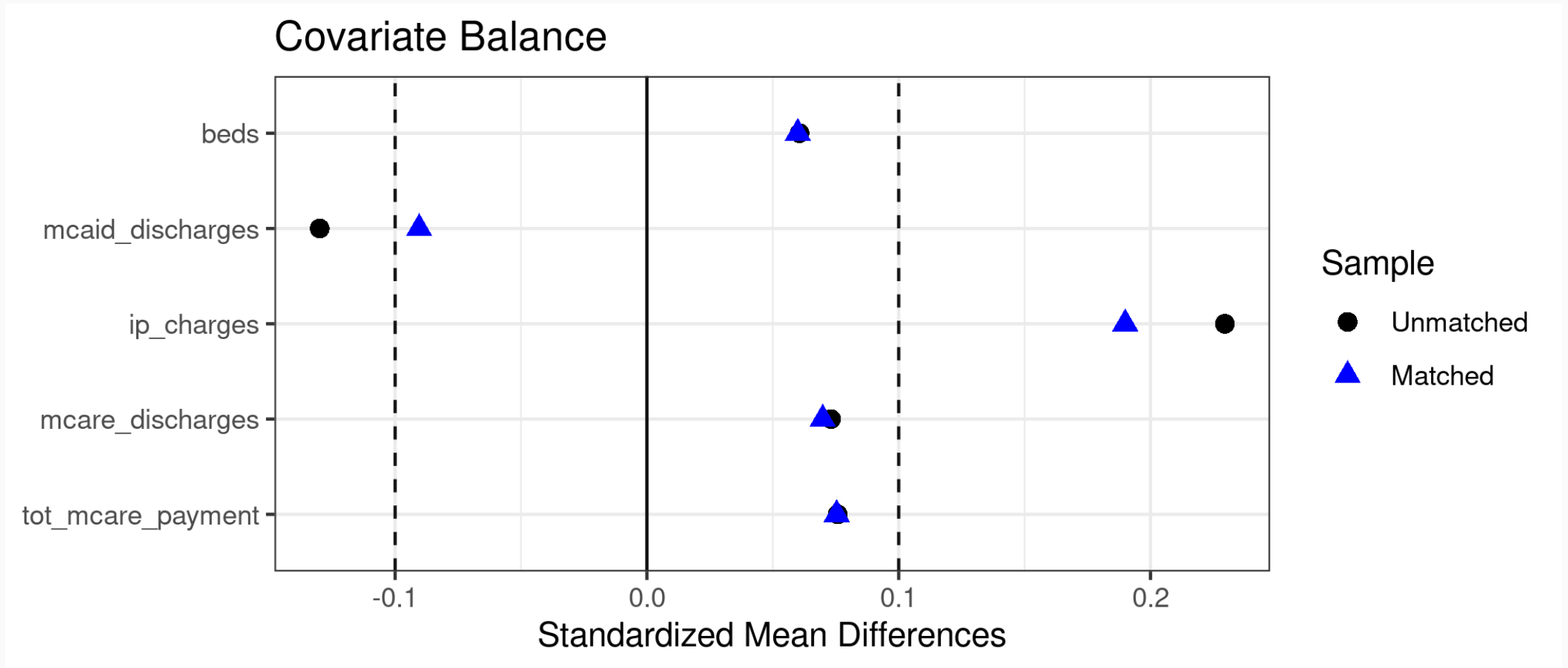
```
m.nn.md ← Matching::Match(Y=lp.vars$price,  
                          Tr=lp.vars$penalty,  
                          X=lp.covs,  
                          M=1,  
                          Weight=2,  
                          estimand="ATE")
```

## 2. Nearest neighbor matching (Mahalanobis)

```
love.plot(bal.tab(m.nn.md, covs = lp.covs, treat = lp.vars$penalty),  
          threshold=0.1,  
          var.names=v.name,  
          grid=FALSE, sample.names=c("Unmatched", "Matched"),  
          position="top", shapes=c("circle", "triangle"),  
          colors=c("black", "blue")) +  
theme_bw()
```



## 2. Nearest neighbor matching (Mahalanobis)

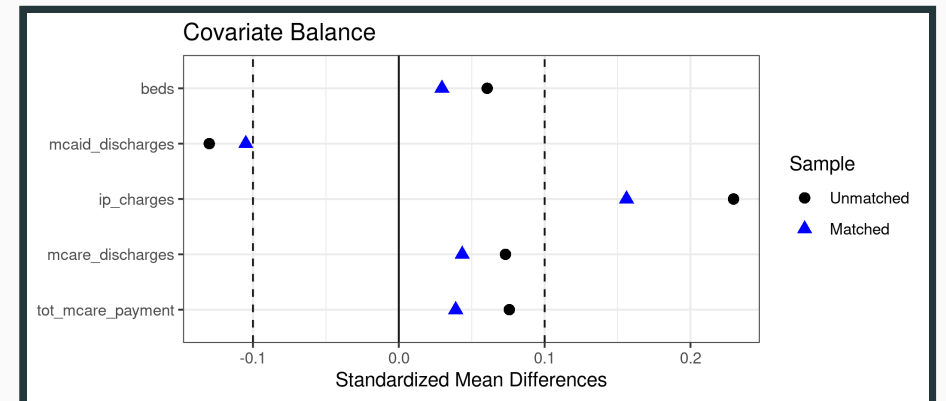


## 2. Nearest neighbor matching (propensity score)

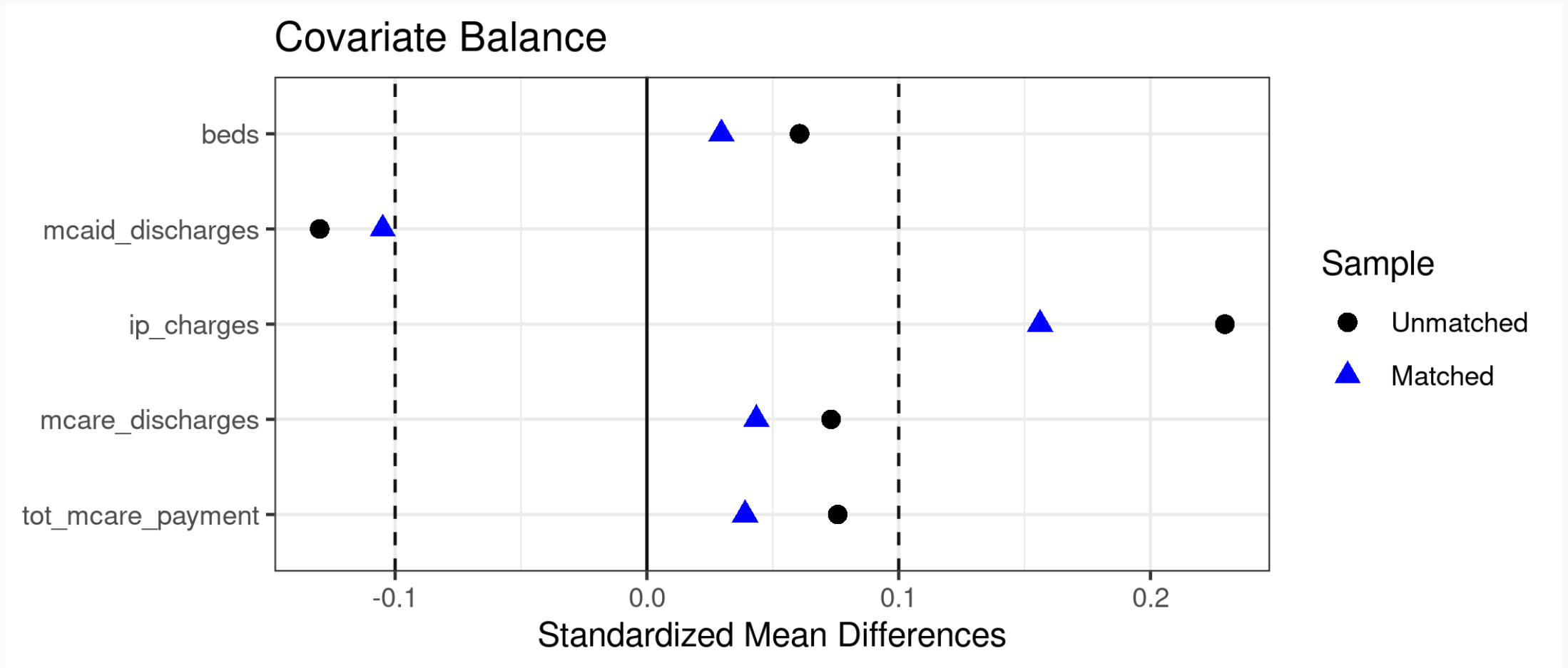
```
logit.model ← glm(penalty ~ beds + mcaid_discharges + ip_charges + mcare_discharges +  
                  tot_mcare_payment, family=binomial, data=lp.vars)  
ps ← fitted(logit.model)  
m.nn.ps ← Matching::Match(Y=lp.vars$price,  
                          Tr=lp.vars$penalty,  
                          X=ps,  
                          M=1,  
                          estimand="ATE")
```

## 2. Nearest neighbor matching (propensity score)

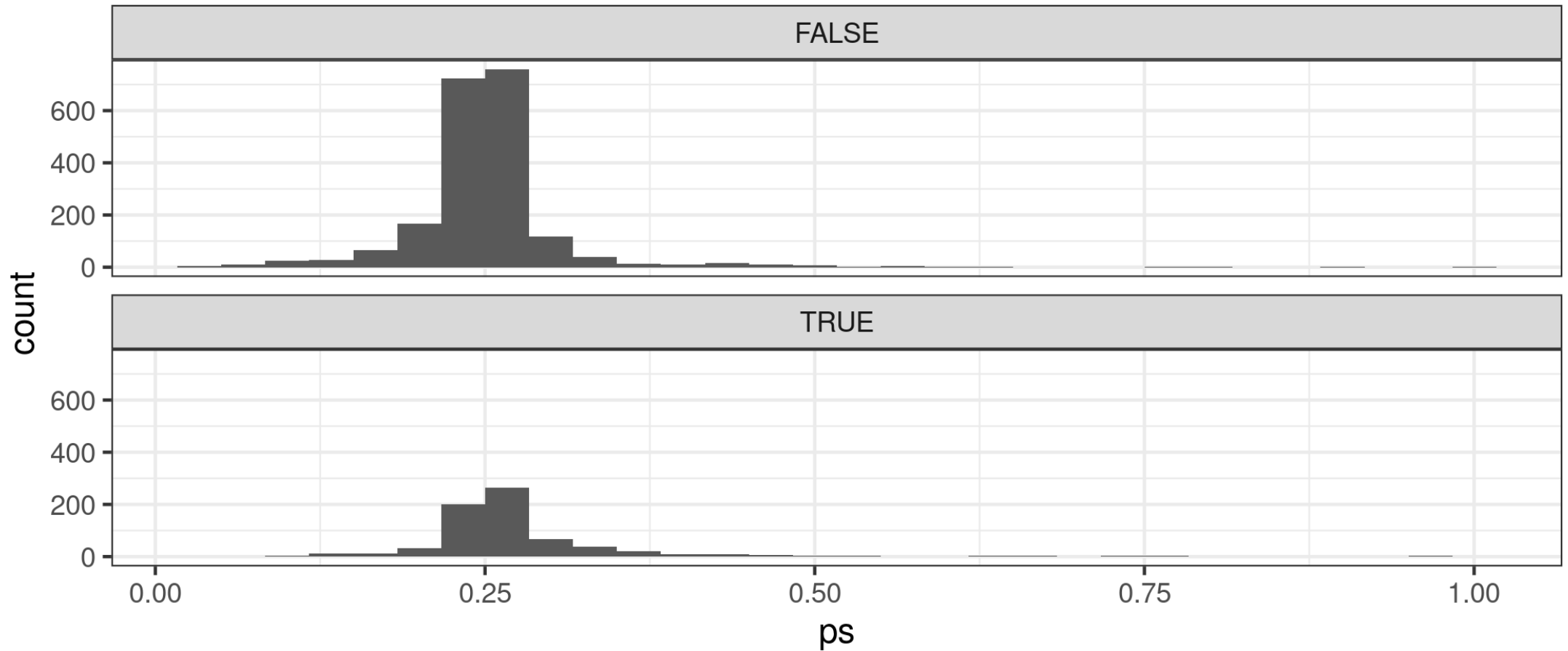
```
love.plot(bal.tab(m.nn.ps, covs = lp.covs, treat = lp.vars$penalty),
          threshold=0.1,
          var.names=v.name,
          grid=FALSE, sample.names=c("Unmatched", "Matched"),
          position="top", shapes=c("circle", "triangle"),
          colors=c("black", "blue")) +
theme_bw()
```



## 2. Nearest neighbor matching (propensity score)



### 3. Weighting



# Results: Exact matching

```
##
## Estimate ... 1777.6
## AI SE..... 34.725
## T-stat..... 51.191
## p.val..... < 2.22e-16
##
## Original number of observations..... 2707
## Original number of treated obs..... 698
## Matched number of observations..... 12
## Matched number of observations (unweighted). 12
##
## Number of obs dropped by 'exact' or 'caliper' 2695
```



# Results: Nearest neighbor

- Inverse variance

```
##  
## Estimate ... -526.95  
## AI SE..... 223.06  
## T-stat..... -2.3623  
## p.val..... 0.01816  
##  
## Original number of observations..... 2707  
## Original number of treated obs..... 698  
## Matched number of observations..... 2707  
## Matched number of observations (unweighted). 2711
```

# Results: Nearest neighbor

- Mahalanobis

```
##  
## Estimate ... -492.82  
## AI SE..... 223.55  
## T-stat..... -2.2046  
## p.val..... 0.027485  
##  
## Original number of observations..... 2707  
## Original number of treated obs..... 698  
## Matched number of observations..... 2707  
## Matched number of observations (unweighted). 2708
```

# Results: Nearest neighbor

- Propensity score

```
##
## Estimate ...    -201.03
## AI SE.....   275.76
## T-stat.....   -0.72898
## p.val.....    0.46601
##
## Original number of observations..... 2707
## Original number of treated obs..... 698
## Matched number of observations..... 2707
## Matched number of observations (unweighted). 14795
```

# Results: IPW weighting

```
lp.vars <- lp.vars %>%  
  mutate(ipw = case_when(  
    penalty=1 ~ 1/ps,  
    penalty=0 ~ 1/(1-ps),  
    TRUE ~ NA_real_  
  ))  
mean.t1 <- lp.vars %>% filter(penalty=1) %>%  
  select(price, ipw) %>% summarize(mean_p=weighted.mean(price,w=ipw))  
mean.t0 <- lp.vars %>% filter(penalty=0) %>%  
  select(price, ipw) %>% summarize(mean_p=weighted.mean(price,w=ipw))  
mean.t1$mean_p - mean.t0$mean_p
```

```
## [1] -196.8922
```

# Results: IPW weighting with regression

```
ipw.reg <- lm(price ~ penalty, data=lp.vars, weights=ipw)
summary(ipw.reg)

##
## Call:
## lm(formula = price ~ penalty, data = lp.vars, weights = ipw)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -18691  -4802  -1422    2651   94137
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9876.4      147.8   66.808  <2e-16 ***
## penaltyTRUE   -196.9      211.2   -0.932    0.351
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7829 on 2705 degrees of freedom
## Multiple R-squared:  0.0003211,    Adjusted R-squared:  -4.85e-05
## F-statistic: 0.8688 on 1 and 2705 DF,  p-value: 0.3514
```

# Results: Regression

```
reg1.dat <- lp.vars %>% filter(penalty=1, complete.cases(.))
reg1 <- lm(price ~ beds+ mcaid_discharges + ip_charges + mcare_discharges +
           tot_mcare_payment, data=reg1.dat)

reg0.dat <- lp.vars %>% filter(penalty=0, complete.cases(.))
reg0 <- lm(price ~ beds + mcaid_discharges + ip_charges + mcare_discharges +
           tot_mcare_payment, data=reg0.dat)
pred1 <- predict(reg1,new=lp.vars)
pred0 <- predict(reg0,new=lp.vars)
mean(pred1-pred0)

## [1] -5.845761
```

# Results: Regression in one step

```
reg.dat ← lp.vars %>% ungroup() %>% filter(complete.cases(.)) %>%  
  mutate(beds_diff = penalty*(beds - mean(beds)),  
         mcaid_diff = penalty*(mcaid_discharges - mean(mcaid_discharges)),  
         ip_diff = penalty*(ip_charges - mean(ip_charges)),  
         mcare_diff = penalty*(mcare_discharges - mean(mcare_discharges)),  
         mpay_diff = penalty*(tot_mcare_payment - mean(tot_mcare_payment)))  
reg ← lm(price ~ penalty + beds + mcaid_discharges + ip_charges + mcare_discharges + tot_mcare_payment +  
         beds_diff + mcaid_diff + ip_diff + mcare_diff + mpay_diff,  
         data=reg.dat)
```

# Results: Regression in one step

```
##
## Call:
## lm(formula = price ~ penalty + beds + mcaid_discharges + ip_charges +
##      mcare_discharges + tot_mcare_payment + beds_diff + mcaid_diff +
##      ip_diff + mcare_diff + mpay_diff, data = reg.dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -38175  -2900   -597    2105   67409
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    8.466e+03  1.711e+02  49.482 < 2e-16 ***
## penaltyTRUE    -5.846e+00  2.124e+02  -0.028  0.97804
## beds           1.107e+00  1.421e+00   0.779  0.43618
## mcaid_discharges -4.714e-01  7.296e-02  -6.462 1.23e-10 ***
## ip_charges       6.426e-06  1.285e-06   5.002 6.04e-07 ***
## mcare_discharges -8.122e-01  9.257e-02  -8.774 < 2e-16 ***
## tot_mcare_payment 9.502e-05  6.858e-06  13.857 < 2e-16 ***
## beds_diff       2.517e+00  2.986e+00   0.843  0.39931
## mcaid_diff       1.058e-01  1.570e-01   0.674  0.50050
## ip_diff         -4.534e-06  2.027e-06  -2.237  0.02539 *
## mcare_diff       4.806e-01  1.809e-01   2.657  0.00793 **
## mpay_diff       -5.452e-05  1.321e-05  -4.128 3.78e-05 ***
## ---
```



# Summary of ATEs

1. Exact matching: 1777.63
2. NN matching, inverse variance: -526.95
3. NN matching, mahalanobis: -492.82
4. NN matching, pscore: -201.03
5. Inverse pscore weighting: -196.89
6. IPW regression: -196.89
7. Regression: -5.85
8. Regression 1-step: -5.85

# So what have we learned?

---

# Key assumptions for causal inference

1. Selection on observables
2. Common support

These become more nuanced but the intuition is the same in almost all questions of causal inference.

# Causal effect assuming selection on observables

If we assume selection on observables holds, then we only need to condition on the relevant covariates to identify a causal effect. But we still need to ensure common support...

1. Matching
2. Reweighting
3. Regression