Quant II

Lab 1: DAGs, Potential Outcomes

Dias Akhmetbekov

2024-02-02

Hi!

- Dias Akhmetbekov, 3^{th} year PhD.

Hi!

- Dias Akhmetbekov, 3^{th} year PhD.
- Fields: Comparative Politics

Hi!

- Dias Akhmetbekov, 3^{th} year PhD.
- Fields: Comparative Politics
- Email: da2669@nyu.edu

Hi!

- Dias Akhmetbekov, 3^{th} year PhD.
- Fields: Comparative Politics
- Email: da2669@nyu.edu
- Office: 302

Logistics (1): labs

- Lab: Fridays, 2pm - 4pm EST, Room 435

Logistics (1): labs

- Lab: Fridays, 2pm - 4pm EST, Room 435
- Lab materials will be posted on the lab's GitHub repo: https://github.com/AkhmetbekovDias/quant2-labs-spring2024

Logistics (1): labs

- Lab: Fridays, 2pm - 4pm EST, Room 435
- Lab materials will be posted on the lab's GitHub repo: https://github.com/AkhmetbekovDias/quant2-labs-spring2024
- Office hours: by appointment

Logistics (2): homeworks

- Homework due via email to Cyrus and me by the indicated deadline

Logistics (2): homeworks

- Homework due via email to Cyrus and me by the indicated deadline
- Deadline is **strict**

Logistics (2): homeworks

- Homework due via email to Cyrus and me by the indicated deadline
- Deadline is **strict**
- Submit **PDF document** + **Code used**

Logistics (2): homeworks

- Homework due via email to Cyrus and me by the indicated deadline
- Deadline is **strict**
- Submit **PDF document** + **Code used**
  - RMarkdown recommended

Logistics (2): homeworks

- Homework due via email to Cyrus and me by the indicated deadline
- Deadline is **strict**
- Submit **PDF document** + **Code used**
  - RMarkdown recommended
- Code should be well commented

Logistics (2): homeworks

- Homework due via email to Cyrus and me by the indicated deadline
- Deadline is **strict**
- Submit **PDF document** + **Code used**
  - RMarkdown recommended
- Code should be well commented
- Tables and plots format should be of high quality

Logistics (2): homeworks

- Homework due via email to Cyrus and me by the indicated deadline
- Deadline is **strict**
- Submit **PDF document** + **Code used**
  - RMarkdown recommended
- Code should be well commented
- Tables and plots format should be of high quality
- Substantive answers should be presented in the written paragraphs

Logistics (2): homeworks

- Homework due via email to Cyrus and me by the indicated deadline
- Deadline is **strict**
- Submit **PDF document** + **Code used**
  - RMarkdown recommended
- Code should be well commented
- Tables and plots format should be of high quality
- Substantive answers should be presented in the written paragraphs
- Ultimately, the goal is to learn how to **do** and **communicate** empirical research

DAG

- **DAG** is Directed Acyclic Graph

DAG

- **DAG** is Directed Acyclic Graph
  - *Directed*: No reverse causality or simultaneity;

DAG

- **DAG** is Directed Acyclic Graph
  - *Directed*: No reverse causality or simultaneity;
  - *Acyclic*: No cycles

DAG

- **DAG** is Directed Acyclic Graph
  - *Directed*: No reverse causality or simultaneity;
  - *Acyclic*: No cycles
- DAGs is a parsimonious representation of the qualitative aspects of the data generating process

DAG

- **DAG** is Directed Acyclic Graph
  - *Directed*: No reverse causality or simultaneity;
  - *Acyclic*: No cycles
- DAGs is a parsimonious representation of the qualitative aspects of the data generating process
  - Letters (X,Z,Y etc.) are random variables;

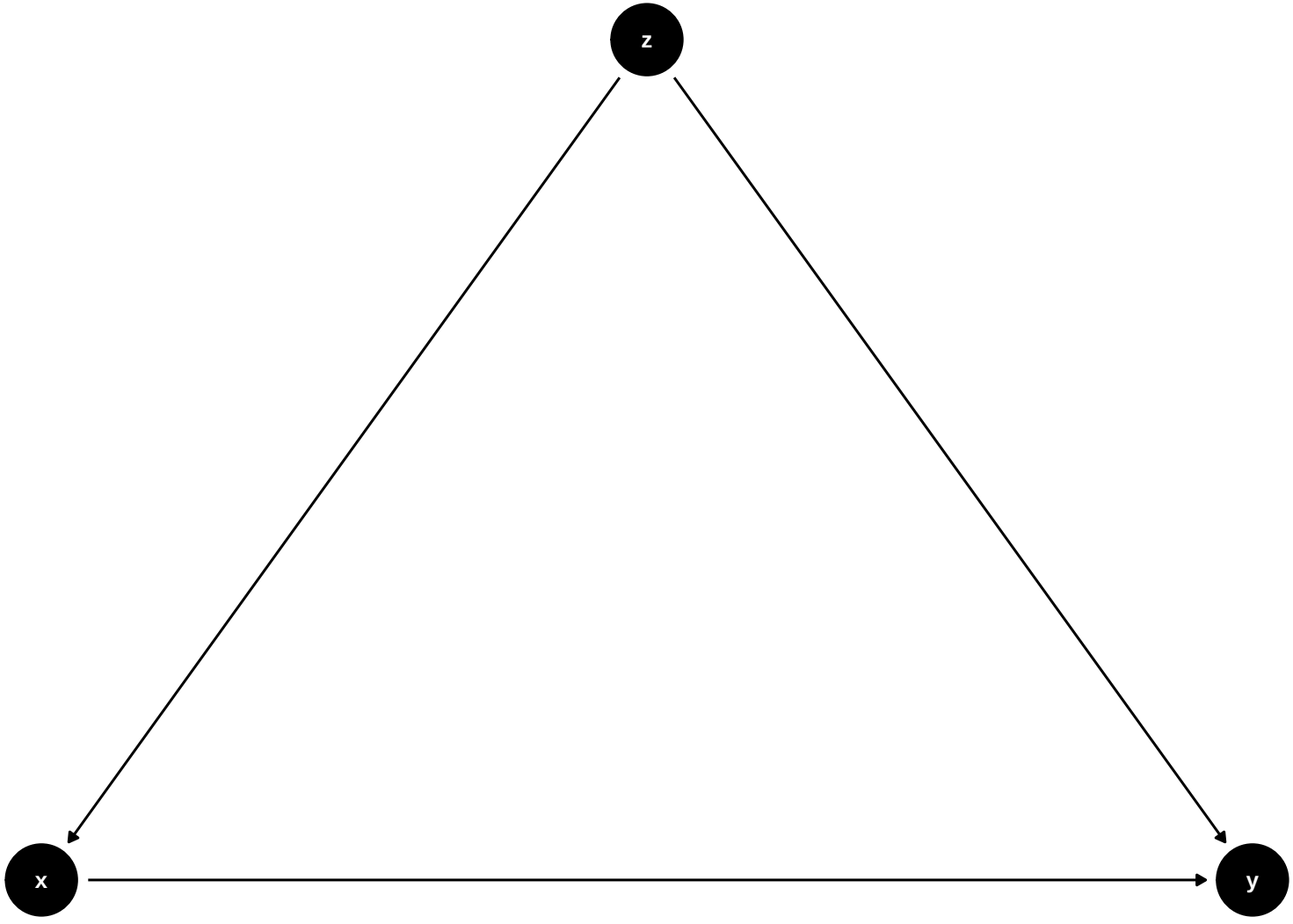
DAG

- **DAG** is Directed Acyclic Graph
  - *Directed*: No reverse causality or simultaneity;
  - *Acyclic*: No cycles
- DAGs is a parsimonious representation of the qualitative aspects of the data generating process
  - Letters (X,Z,Y etc.) are random variables;
  - Arrows ($X \rightarrow Y$) denote a (possible) direct causal effect of D on Y;

DAG

- **DAG** is Directed Acyclic Graph
  - *Directed*: No reverse causality or simultaneity;
  - *Acyclic*: No cycles
- DAGs is a parsimonious representation of the qualitative aspects of the data generating process
  - Letters (X,Z,Y etc.) are random variables;
  - Arrows ($X \rightarrow Y$) denote a (possible) direct causal effect of D on Y;
  - no assumptions about the functional form or distribution.

Simple DAG



Simulation

```r
# set  seed
set.seed(1000)

# simulate data
n <- 1000
z <- rnorm(n)
x <- z + rnorm(n)
y <- x + z + rnorm(n)

# unconditional
lm1 <- lm(y~x)
# conditional
lm2 <- lm(y~x+z)
```

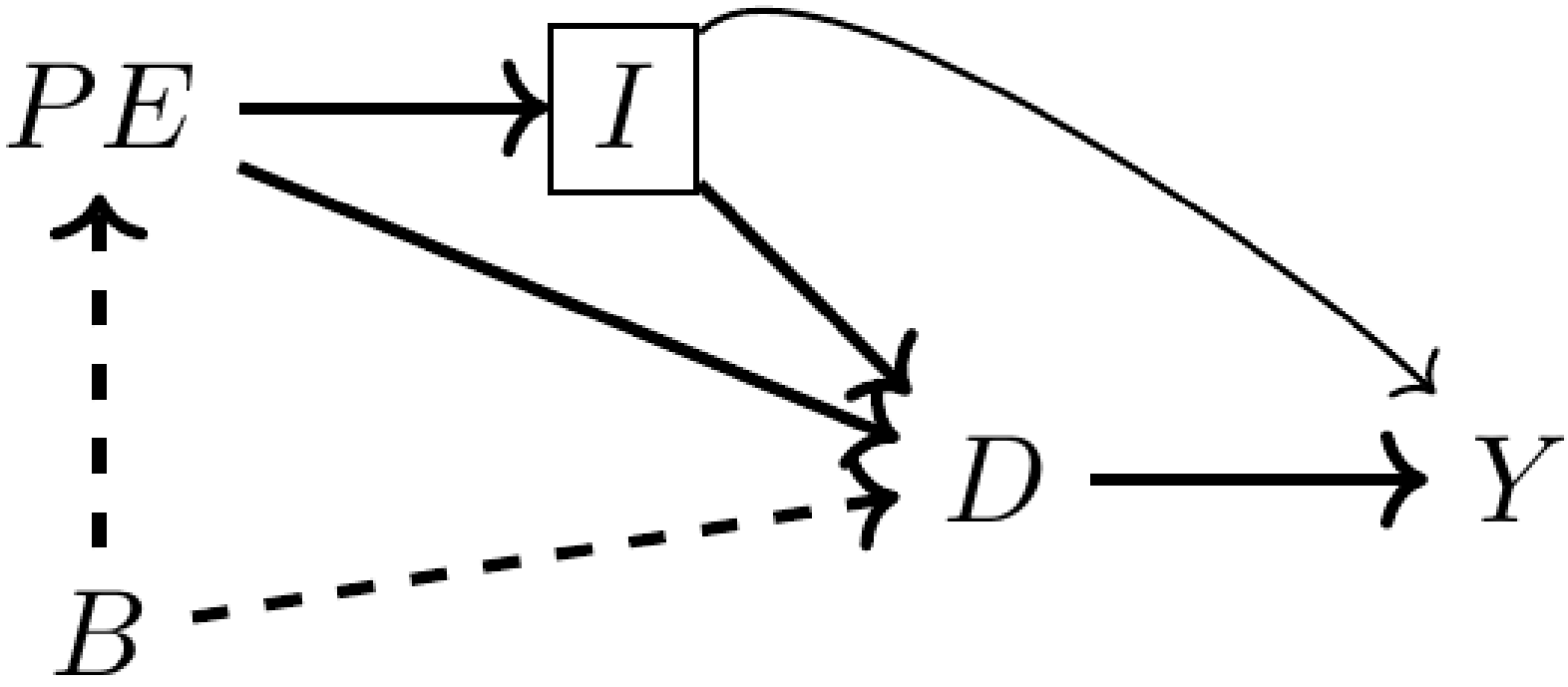
Simulation

```
stargazer(lm1,lm2, type = "text",
          column.labels = c("Biased unconditional",
                            "Unbiased Conditional"))
```

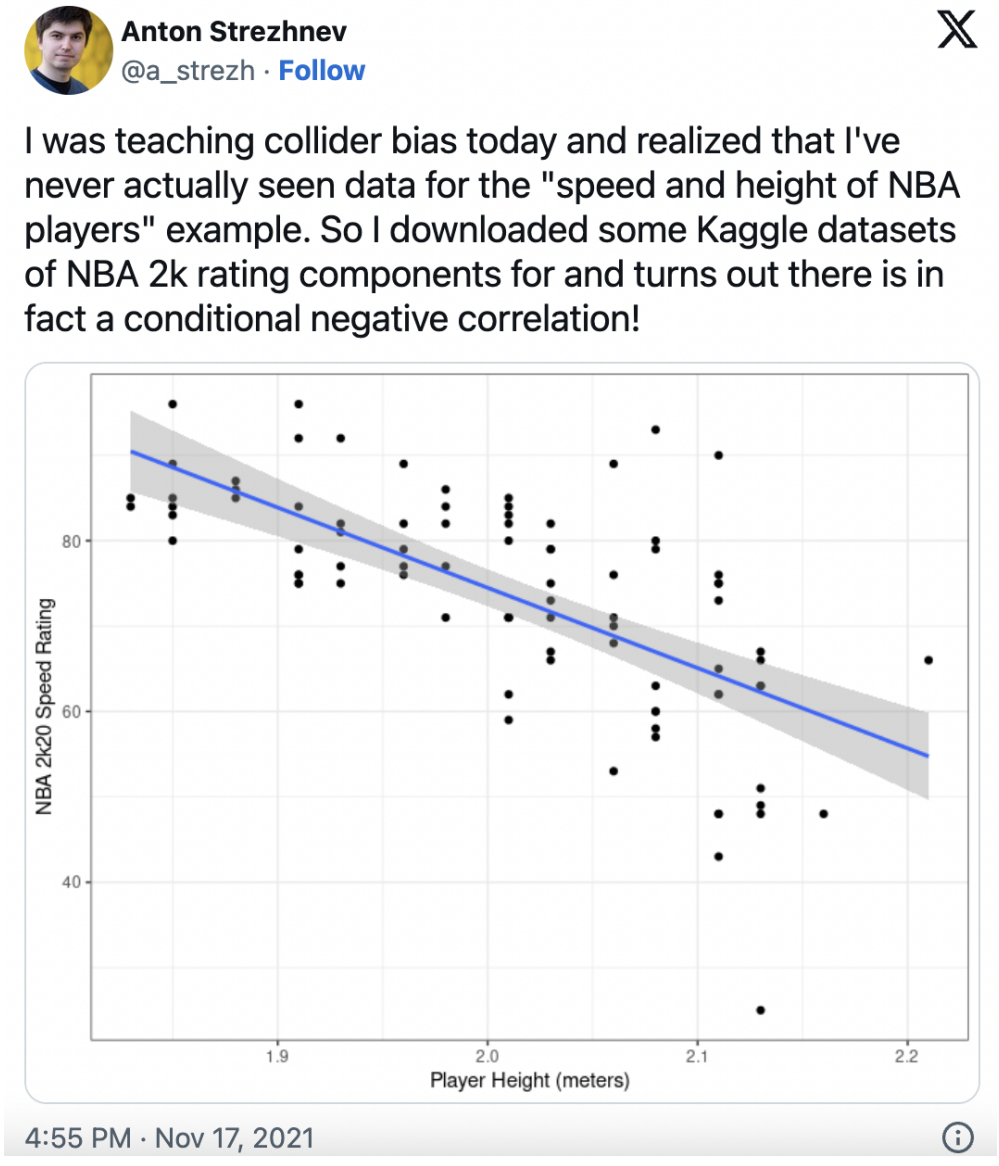
```
##
## ===============================================================================
##                                        Dependent variable:
##                    ---------------------------------------------------------
##                                                y
##                           Biased unconditional       Unbiased Conditional
##                                   (1)                         (2)
## -------------------------------------------------------------------------------
## x                                1.511***                    1.040***
##                                  (0.026)                     (0.032)
##
## z                                                            0.954***
##                                                              (0.047)
##
## Constant                         -0.008                      -0.002
##                                  (0.038)                     (0.032)
##
## -------------------------------------------------------------------------------
## Observations                      1,000                       1,000
## R2                                0.769                       0.837
## Adjusted R2                       0.768                       0.837
## Residual Std. Error         1.193 (df = 998)            1.002 (df = 997)
## F Statistic           3,317.308*** (df = 1; 998) 2,557.355*** (df = 2; 997)
## ===============================================================================
## Note:                                            *p<0.1; **p<0.05; ***p<0.01
```

More complex DAG

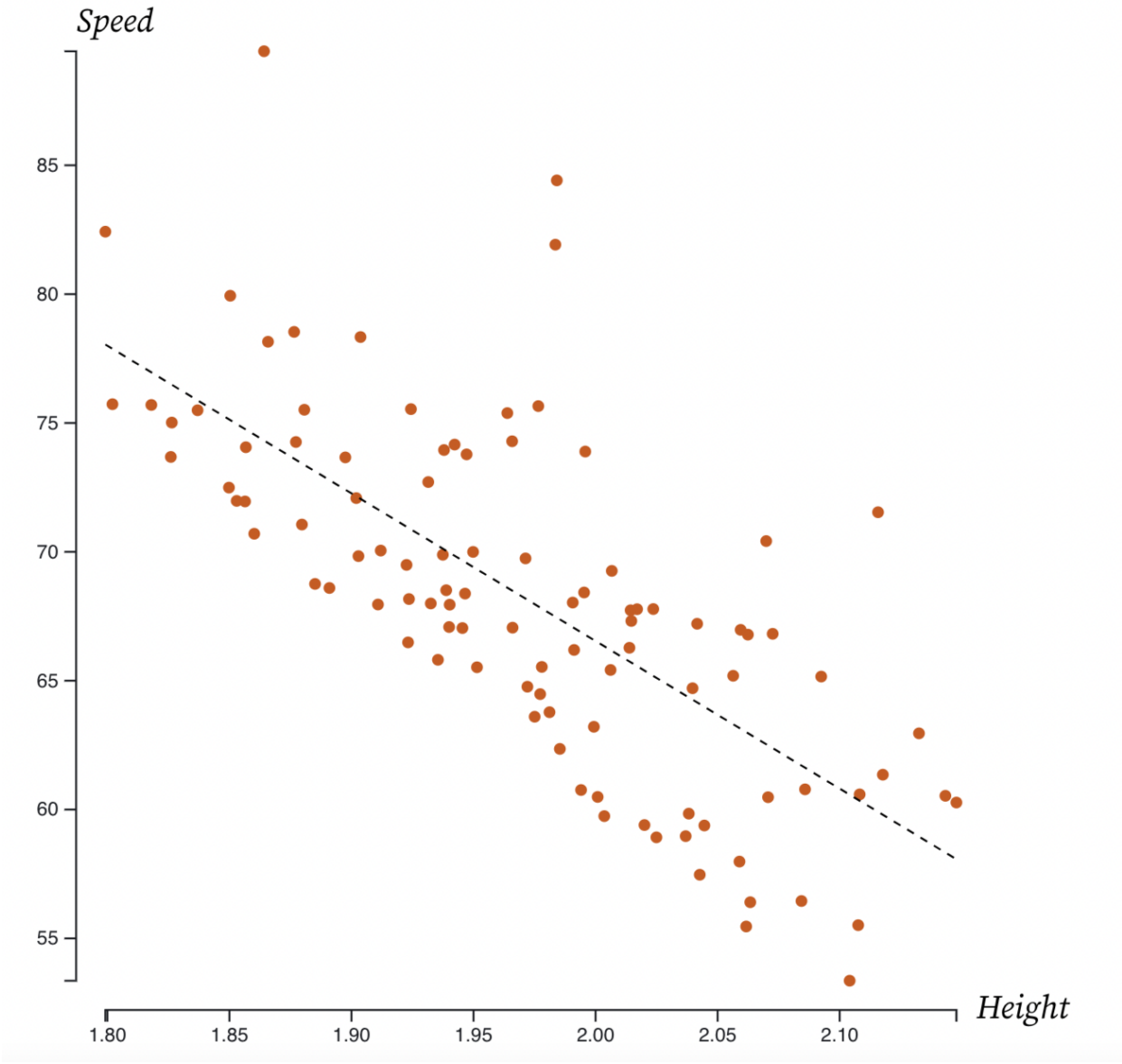
An example from the Mixtape inspired by Becker (1994):



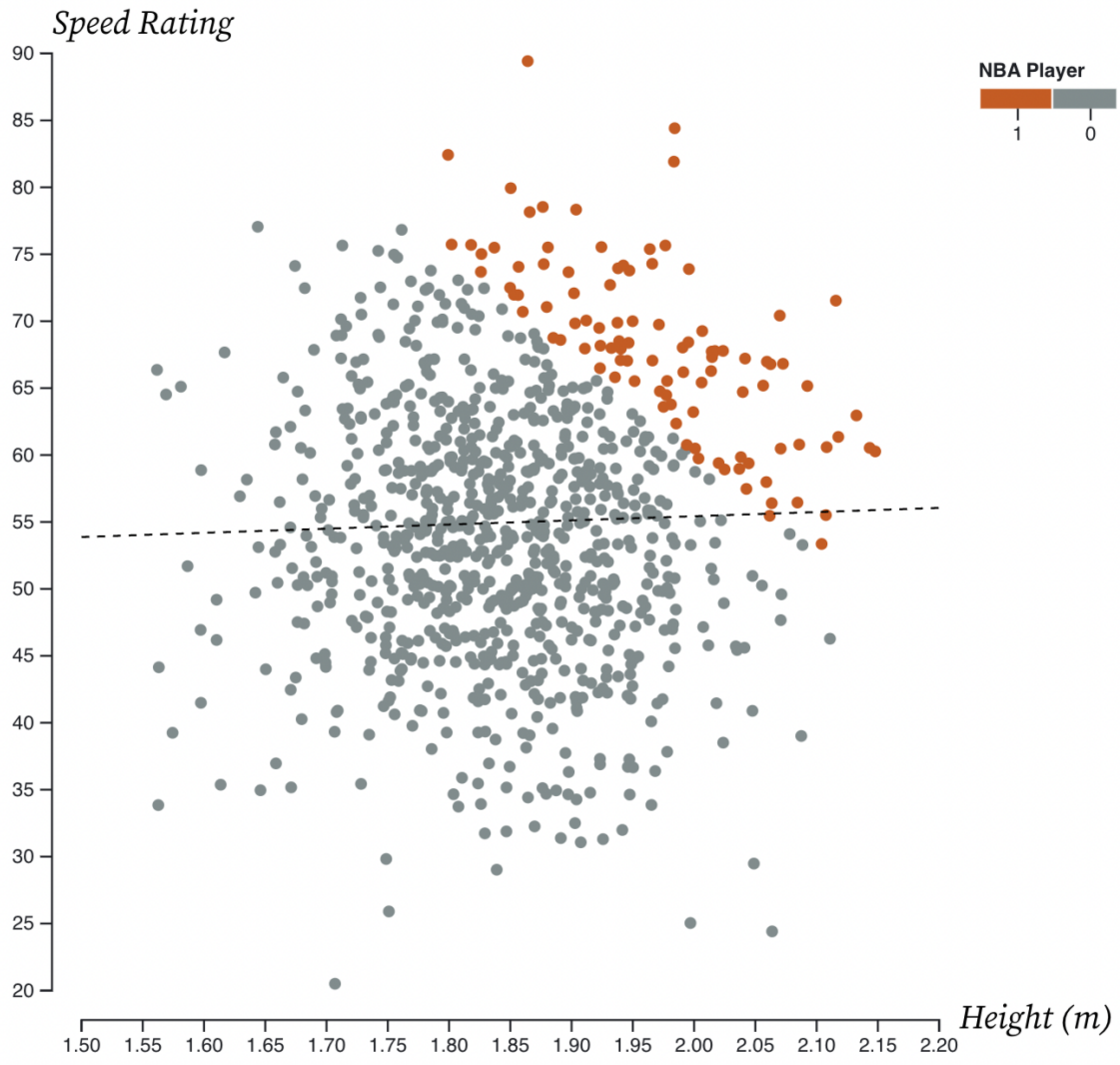
Collider Bias



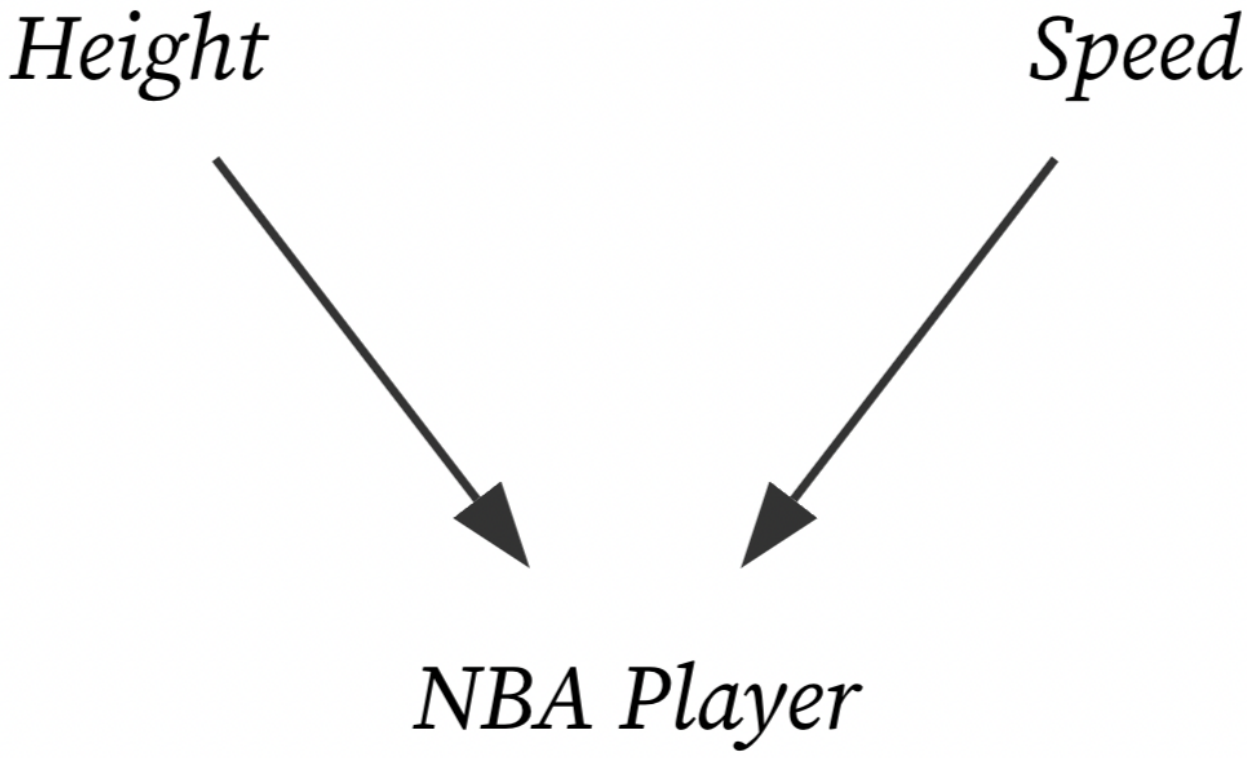
Collider bias



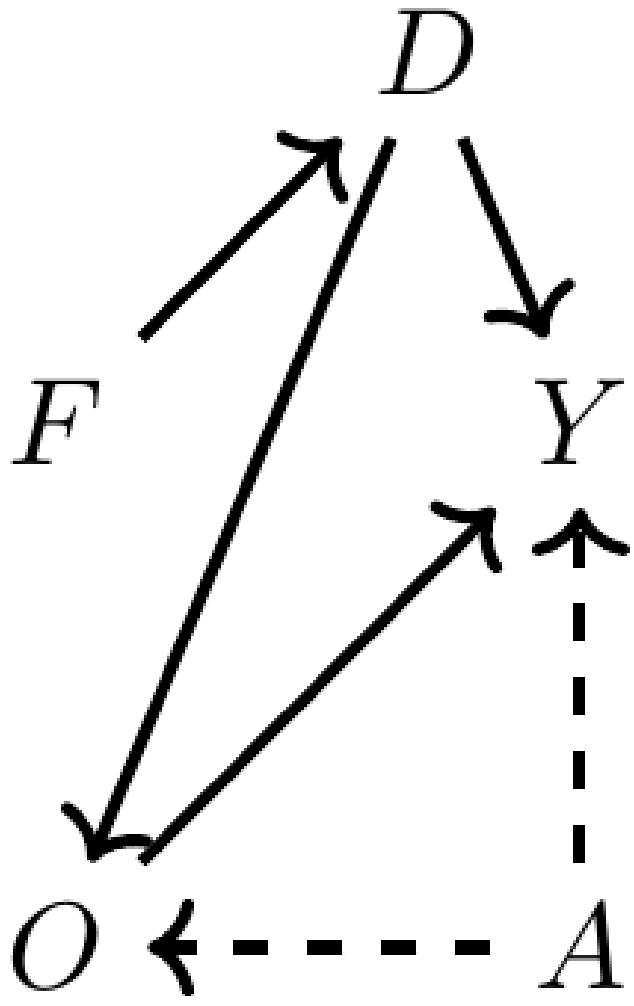
Collider bias



Collider bias



Collider bias



Collider bias

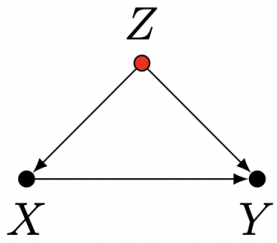
```r
# Set seed
set.seed(123)

# Simulate  data
tb <- tibble(
  female = ifelse(runif(10000) >= 0.5, 1, 0),
  ability = rnorm(10000),
  discrimination = female,
  occupation = 1 + 2*ability + 0*female - 2*discrimination + rnorm(10000),
  wage = 1-1*discrimination + 1*occupation + 2*ability + rnorm(10000)
)

# Estimate regressions
lm_1 <- lm(wage ~ female, tb)
lm_2 <- lm(wage ~ female + occupation, tb)
lm_3 <- lm(wage ~ female + occupation + ability, tb)
```

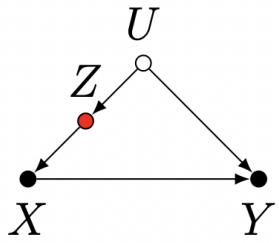
Collider bias

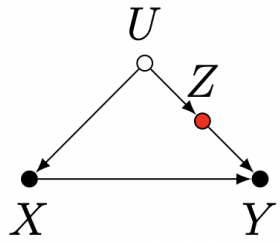
```
# Compare
stargazer(lm_1, lm_2, lm_3, type = "text",
          column.labels = c("Biased unconditional",
                            "Biased",
                            "Unbiased Conditional"))
```

```
##
## ==========================================================================================
##                                          Dependent variable:
##                    ---------------------------------------------------------------------
##                                                  wage
##                          Biased unconditional      Biased          Unbiased Conditional
##                                  (1)               (2)                    (3)
## ------------------------------------------------------------------------------------------
## female                         -3.066***         0.587***               -1.050***
##                                 (0.085)           (0.030)                (0.028)
##
## occupation                                        1.796***               0.987***
##                                                   (0.006)                (0.010)
##
## ability                                                                  2.033***
##                                                                          (0.022)
##
## Constant                        2.023***          0.222***               1.025***
##                                 (0.060)           (0.020)                (0.017)
##
## ------------------------------------------------------------------------------------------
## Observations                     10,000            10,000                 10,000
## R2                               0.114             0.912                  0.952
## Adjusted R2                      0.114             0.912                  0.952
## Residual Std. Error        4.265 (df = 9998)  1.347 (df = 9997)      0.994 (df = 9996)
## F Statistic           1,292.306*** (df = 1; 9998) 51,551.530*** (df = 2; 9997) 65,927.470*** (df = 3; 999
## ==========================================================================================
## Note:                                                          *p<0.1; **p<0.05; ***p<0
```

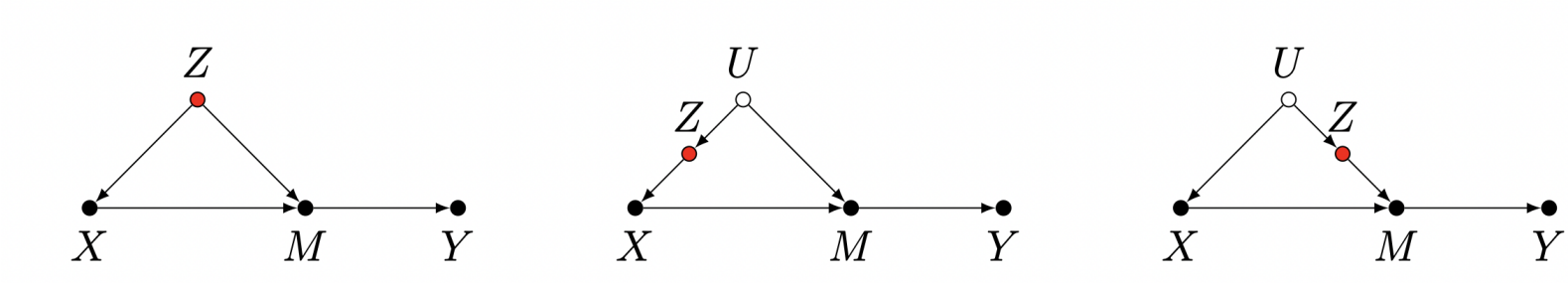
Good Control



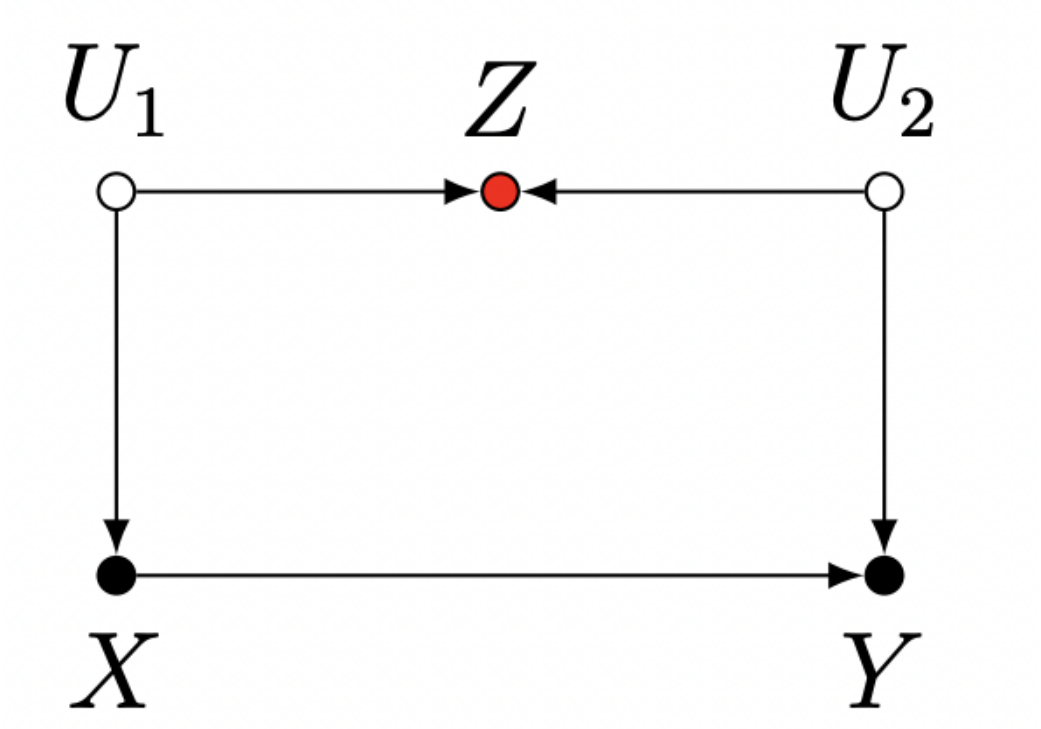




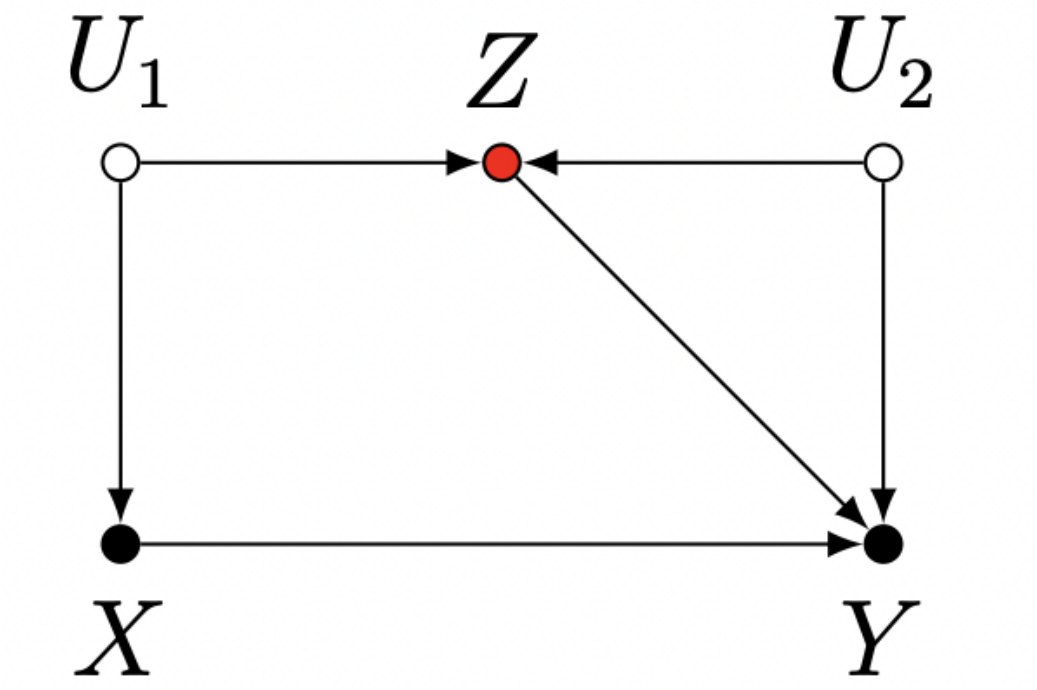
Good Control



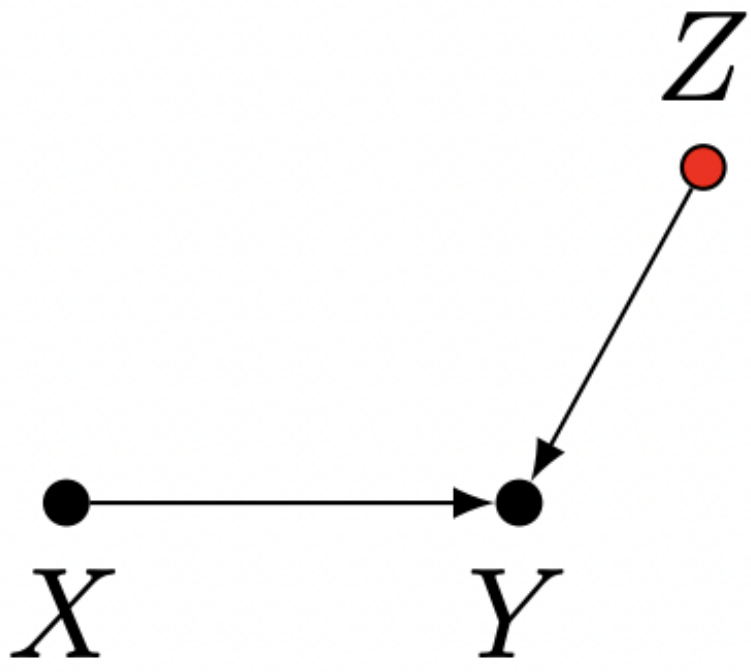
M-Bias



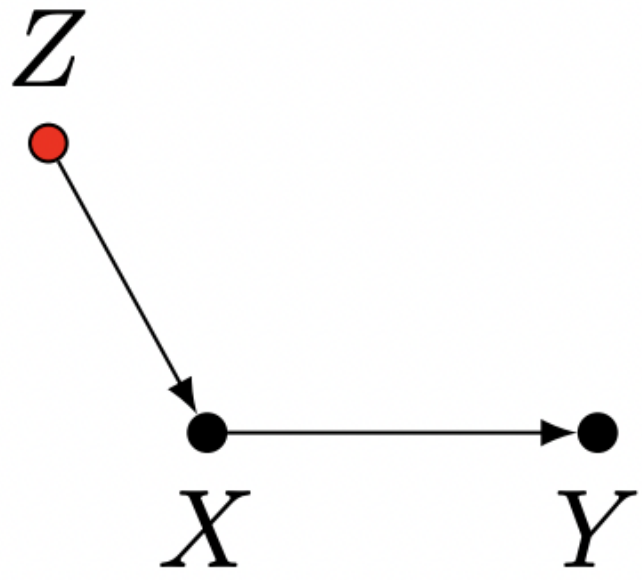
Damned if you do, damned if you don't



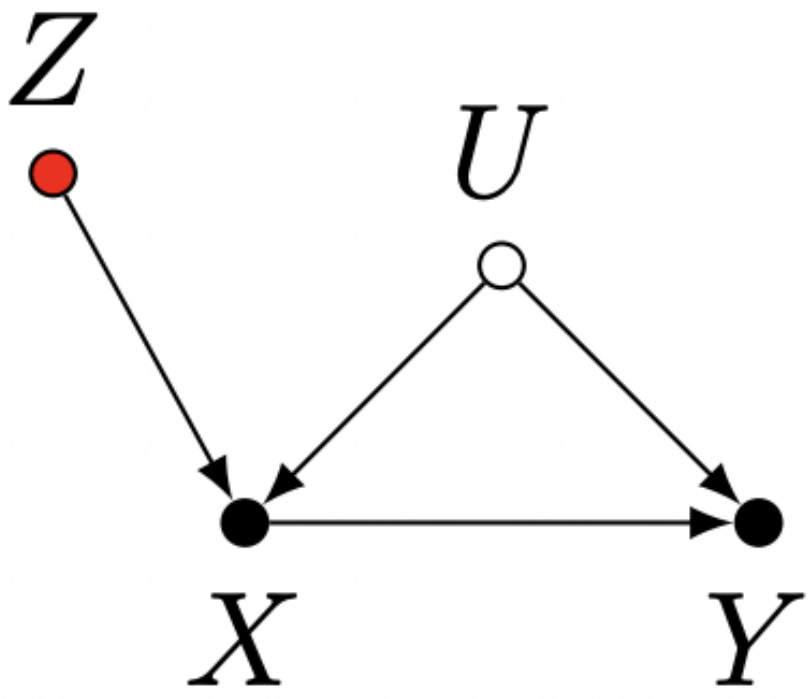
Neutral Control (or even good)



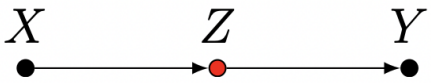
Neutral Control (or even bad)

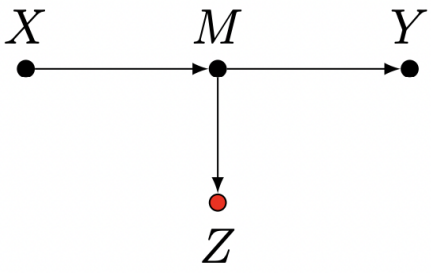


Bias Amplification

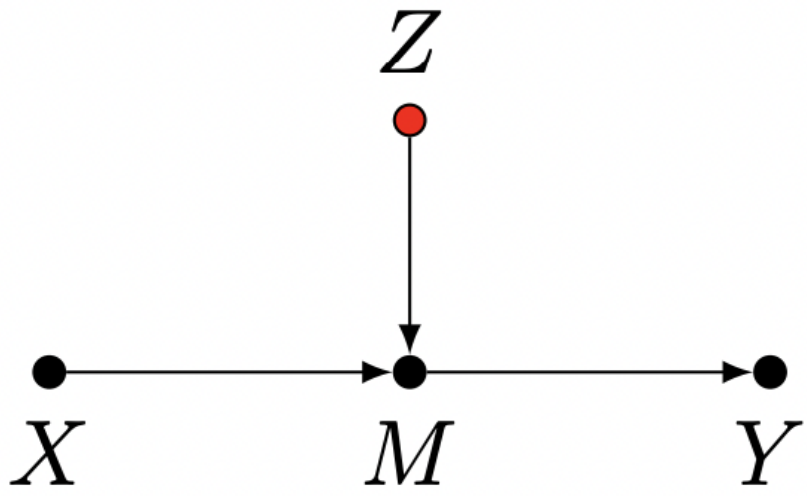


Overcontrol Bias

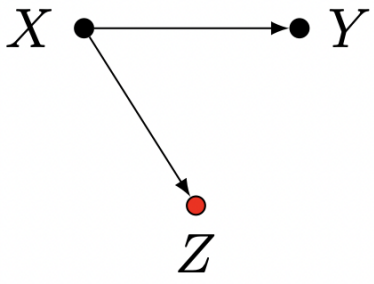


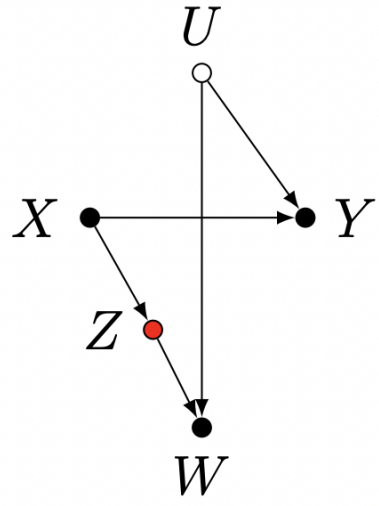


Neutral Control (or even good)

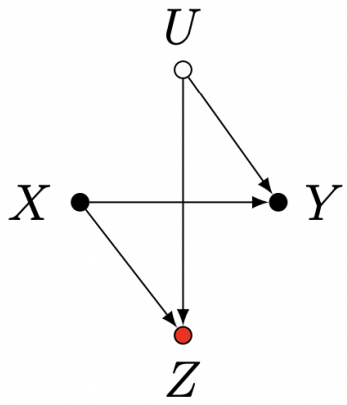


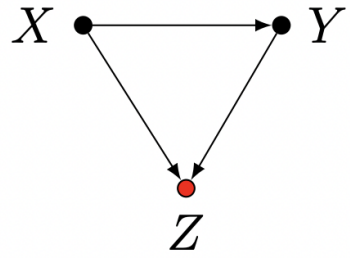
Neutral Control (or even good in case of sample selection)





Colliding Bias





Potential Outcomes

- **Potential outcomes** formally encode counterfactuals (Neyman-Rubin)

Potential Outcomes

- **Potential outcomes** formally encode counterfactuals (Neyman-Rubin)
  - Y_{1i}: outcome that unit i would have if treated;

Potential Outcomes

- **Potential outcomes** formally encode counterfactuals (Neyman-Rubin)
  - Y_{1i}: outcome that unit i would have if treated;
  - Y_{0i}: outcome that unit i would have if untreated;

Potential Outcomes

- **Potential outcomes** formally encode counterfactuals (Neyman-Rubin)
  - Y_{1i}: outcome that unit i would have if treated;
  - Y_{0i}: outcome that unit i would have if untreated;
- Connect observed outcomes to potential outcomes (**consistency**)

Potential Outcomes

- **Potential outcomes** formally encode counterfactuals (Neyman-Rubin)
  - Y_{1i}: outcome that unit i would have if treated;
  - Y_{0i}: outcome that unit i would have if untreated;
- Connect observed outcomes to potential outcomes (**consistency**)
  - $Y_i = Y_i(D_i)$ we observe the potential outcome of observed treatment;

Potential Outcomes

- **Potential outcomes** formally encode counterfactuals (Neyman-Rubin)
  - Y_{1i}: outcome that unit i would have if treated;
  - Y_{0i}: outcome that unit i would have if untreated;
- Connect observed outcomes to potential outcomes (**consistency**)
  - $Y_i = Y_i(D_i)$ we observe the potential outcome of observed treatment;
- **Causal effect** for unit i: $\rho_i = Y_{1i} - Y_{0i}$

Potential Outcomes

- **Potential outcomes** formally encode counterfactuals (Neyman-Rubin)
  - Y_{1i}: outcome that unit i would have if treated;
  - Y_{0i}: outcome that unit i would have if untreated;
- Connect observed outcomes to potential outcomes (**consistency**)
  - $Y_i = Y_i(D_i)$ we observe the potential outcome of observed treatment;
- **Causal effect** for unit i: $\rho_i = Y_{1i} - Y_{0i}$
- **Ignorability assumption**: $D_i \perp\!\!\!\perp (Y_{1i}, Y_{0i})$

Simulating Ignorability

```r
# Set random seed
set.seed(10003)

# Imagine we had a constant individual-level treatment effect
true_effect <- 2
# Our hypothetical population contains 1,000 units - imagine we could observe both Y(1) and Y(0)
N <- 1000 # Population size
dataset <- data.frame(Y0 = rnorm(N, mean = 0, sd = 3))
dataset$Y1 <- dataset$Y0 + true_effect
```

Simulating Ignorability

```
head(as_tibble(dataset))
```

```
## # A tibble: 6 x 2
##       Y0     Y1
##    <dbl>  <dbl>
## 1 -3.13  -1.13
## 2 -0.297  1.70
## 3 -1.10   0.903
## 4 -1.41   0.587
## 5  1.75   3.75
## 6  5.64   7.64
```

Simulating Ignorability

```
# Randomized treatment (.5 probability of treatment)
dataset$D <- rbinom(N, 1, .5) # Not *exactly* half, but independent
# Treatment is a "light switch" - affects what we observe
dataset$Y <- dataset$Y1*dataset$D + dataset$Y0*(1-dataset$D)
# Let's see the data now
head(as_tibble(dataset))
```

```
## # A tibble: 6 x 4
##       Y0     Y1     D     Y
##    <dbl>  <dbl> <int> <dbl>
## 1 -3.13  -1.13      1 -1.13
## 2 -0.297  1.70      1  1.70
## 3 -1.10   0.903     0 -1.10
## 4 -1.41   0.587     0 -1.41
## 5  1.75   3.75      1  3.75
## 6  5.64   7.64      1  7.64
```

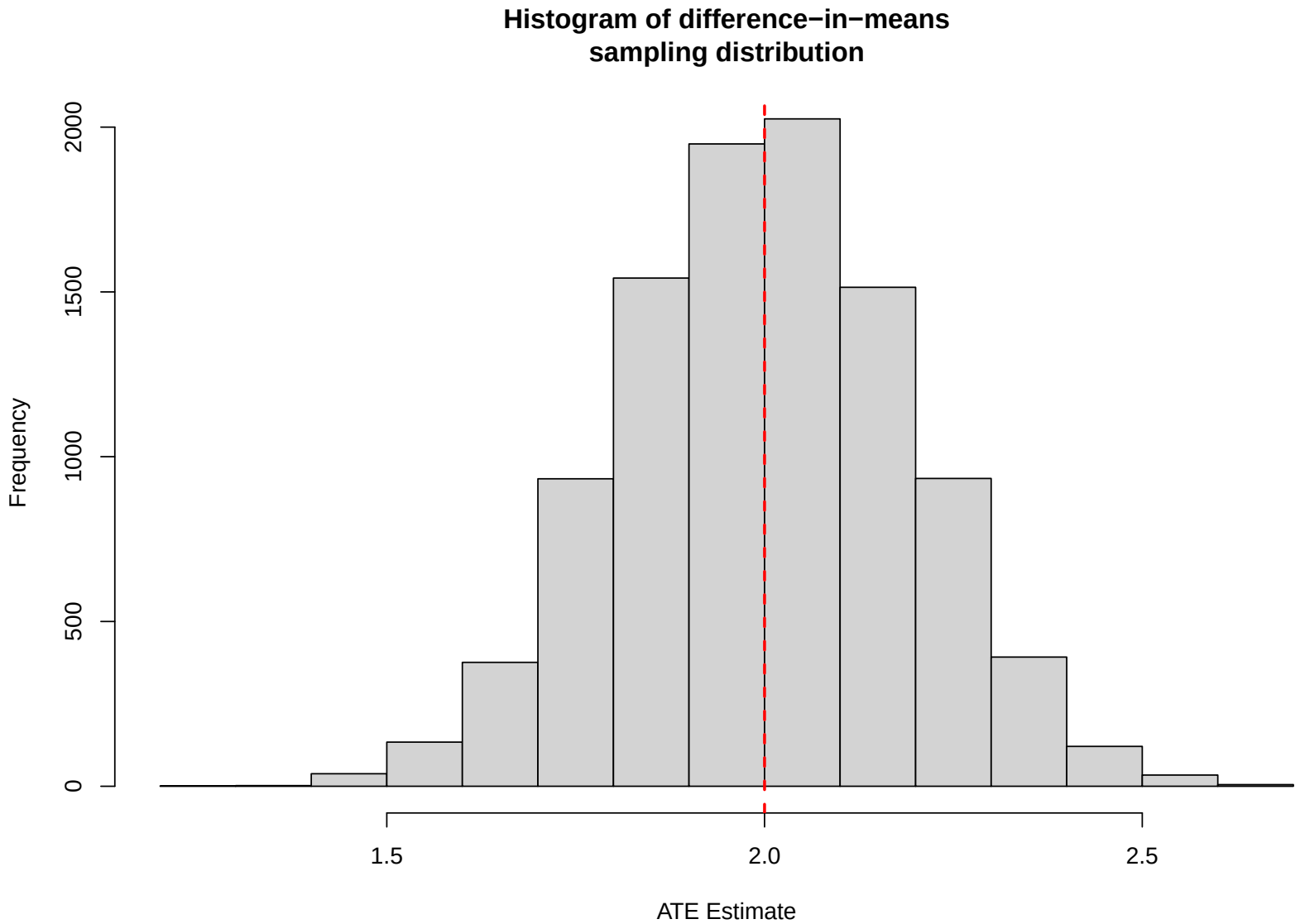
Simulating Ignorability

```
simDataset <- dataset
nIter <- 10000 # number of iterations to run
est_effect <- rep(NA, nIter) # placeholder
for (i in 1:nIter){
# randomly assign treatment again
simDataset$D <- rbinom(N, 1, .5)
# observed outcome
simDataset$Y <- simDataset$Y1*simDataset$D + simDataset$Y0*(1-simDataset$D)
# difference-in-means
est_effect[i] <- mean(simDataset$Y[simDataset$D == 1]) -
mean(simDataset$Y[simDataset$D == 0])
}
# What's the average of treatment effect estimates in repeated samples
mean(est_effect)
```

```
## [1] 2.000468
```

Simulating Ignorability

```
hist(est_effect, xlab="ATE Estimate", ylab="Frequency", main="Histogram of difference-in-means\nsampling d
abline(v=true_effect, col="red", lty=2, lwd=2)
```



Simulating Ignorability

```
newDataset <- dataset # placeholder to not override dataset
# Probability of treatment depends on $Y(1)$
newDataset$D <- rbinom(N, 1, pnorm(newDataset$Y1))
#pnorm is the normal CDF - high positive values = high probabilities
#what is the probability that each observation in our dataset will be assigned to treatment?
quantile(pnorm(newDataset$Y1))
```

```
##           0%          25%          50%          75%         100%
## 1.295055e-20 4.937063e-01 9.753327e-01 9.999673e-01 1.000000e+00
```

```
#now how many units are treated?
mean(newDataset$D)
```

```
## [1] 0.738
```

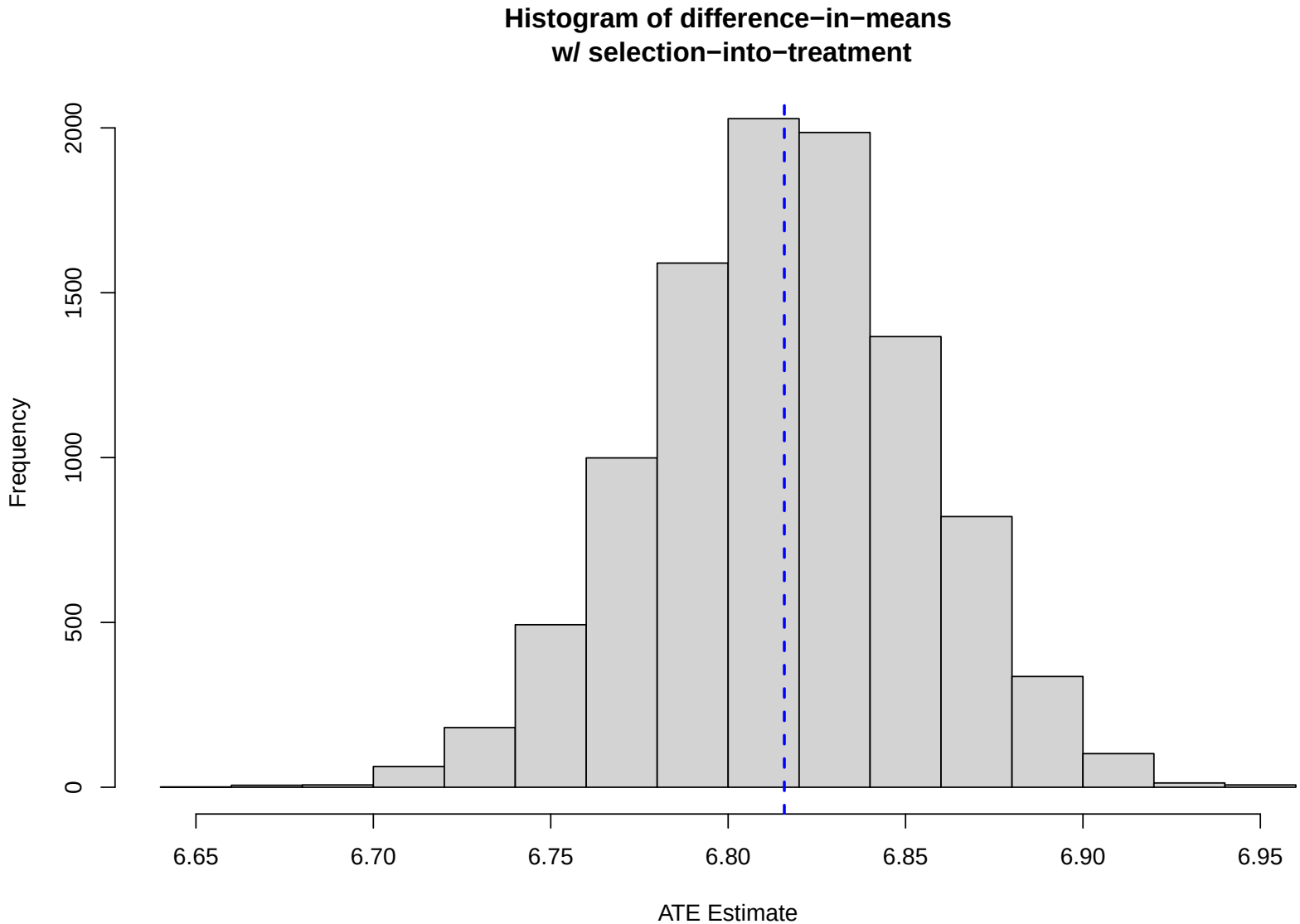
Simulating Ignorability

```r
newSimDataset <- newDataset
nIter <- 10000 # number of iterations to run
est_effect_bias <- rep(NA, nIter) # placeholder
for (i in 1:nIter){
  # non-randomly assign treatment (High Y(1) more likely to be treated)
  newSimDataset$D <- rbinom(N, 1, pnorm(newSimDataset$Y1))
  # observed outcome
  newSimDataset$Y <- newSimDataset$Y1*newSimDataset$D + newSimDataset$Y0*(1-newSimDataset$D)
  # difference-in-means
  est_effect_bias[i] <- mean(newSimDataset$Y[newSimDataset$D == 1]) -
                        mean(newSimDataset$Y[newSimDataset$D == 0])
}
# What's the average of treatment effect estimates in repeated samples
mean(est_effect_bias)
```

```
## [1] 6.815834
```

Simulating ignorability

```
hist(est_effect_bias, xlab="ATE Estimate", ylab="Frequency", main="Histogram of difference-in-means\n w/ s
abline(v=true_effect, col="red", lty=2, lwd=2)
abline(v=mean(est_effect_bias), col="blue", lty=2, lwd=2)
```



Conclusions

- Control for **confounders** / do not control for **colliders**;

Conclusions

- Control for **confounders** / do not control for **colliders**;
- Not all **pre-treatment** covariates are good controls;

Conclusions

- Control for **confounders** / do not control for **colliders**;
- Not all **pre-treatment** covariates are good controls;
- Not all **post-treatment** covariates are bad controls;

Conclusions

- Control for **confounders** / do not control for **colliders**;
- Not all **pre-treatment** covariates are good controls;
- Not all **post-treatment** covariates are bad controls;
- Difference-in-means is an **unbiased estimator** of average treatment effect (ATE) under completely random assignment condition.