Problem Set 2: Research questions and estimands

Alejandro Sarria

2025-09-10

1. Simulate the data

```
set.seed(1)

delta <- 5
N <- 100
beta_0 <- 10
beta_1 <- 1.1

#quiz 1 scores
q1_scores <- rnorm(N, mean = 65, sd = 3)

#errors
u0 <- rnorm(N, mean = 0, sd = 1)
u1 <- rnorm(N, mean = 0, sd = 1)

#potential outcomes
y0 <- beta_0 + beta_1 * q1_scores + u0
y1 <- beta_0 + beta_1 * q1_scores + delta + u1

#assign treatment
treatment <- sample(c(rep(0, N/2), rep(1, N/2)))

#outcome
y_obs <- ifelse(treatment == 1, y1, y0)

df <- data.frame(
  student_id = 1:N,
```

```
  quiz1 = q1_scores,
  treatment = treatment,
  y0 = y0,
  y1 = y1,
  y_obs = y_obs
)

head(df)
```

```
  student_id    quiz1 treatment       y0       y1    y_obs
1          1 63.12064         1 78.81234 84.84210 84.84210
2          2 65.55093         1 82.14814 88.79490 88.79490
3          3 62.49311         1 77.83150 85.32901 85.32901
4          4 69.78584         0 86.92246 91.43352 86.92246
5          5 65.98852         1 81.93279 85.30214 85.30214
6          6 62.53859         1 80.55974 86.29012 86.29012
```

2. Interpretations

(a) δ: The effect of tutoring on the the scores of second quizzes across all students. In this scenario, a δ of 5 suggests that, on average, students that received tutoring after their first quiz scored 5 points more on the second quiz compared to students that did not.
(b) Y^0 and Y^1 intercepts: The baseline quiz 2 score predicted when the quiz 1 score (x) is zero, before adding the treatment effect or error terms. That is, the lowest possible grade on quiz 2 without factoring negative errors. Given how quiz 1 scores are distributed, it is unlikely that any simulated student would obtain this grade on quiz 2.
(c) β_1: The coefficient for quiz 1 scores on quiz 2 scores. That is, how much each additional point on quiz 1 affects the quiz 2 scores regardless of treatment. In this case, it is 1.1, meaning that each point on quiz 1 predicts an additional 1.1 points on quiz 2.

3. The effect of tutoring on student performance

(a) SATE and δ

```
SATE <- sum(y1-y0) / N
print(SATE)
```

```
[1] 5.067482
```

$SATE$ and δ are close but not identical because of the introduction of a small error term u_1 in the formulas for y^0 and y^1. Otherwise, they would be identical.

(b) $\widehat{\text{SATE}}$

```
SATE_hat <- mean(y_obs[treatment == 1]) - mean(y_obs[treatment == 0])
print(SATE_hat)
```

```
[1] 5.125852
```

The $\widehat{\text{SATE}}$ is different from δ and further off from it than $SATE$ because of the noise introduced in the calculation by sampling from the errors and the randomness of the treatment assignment.

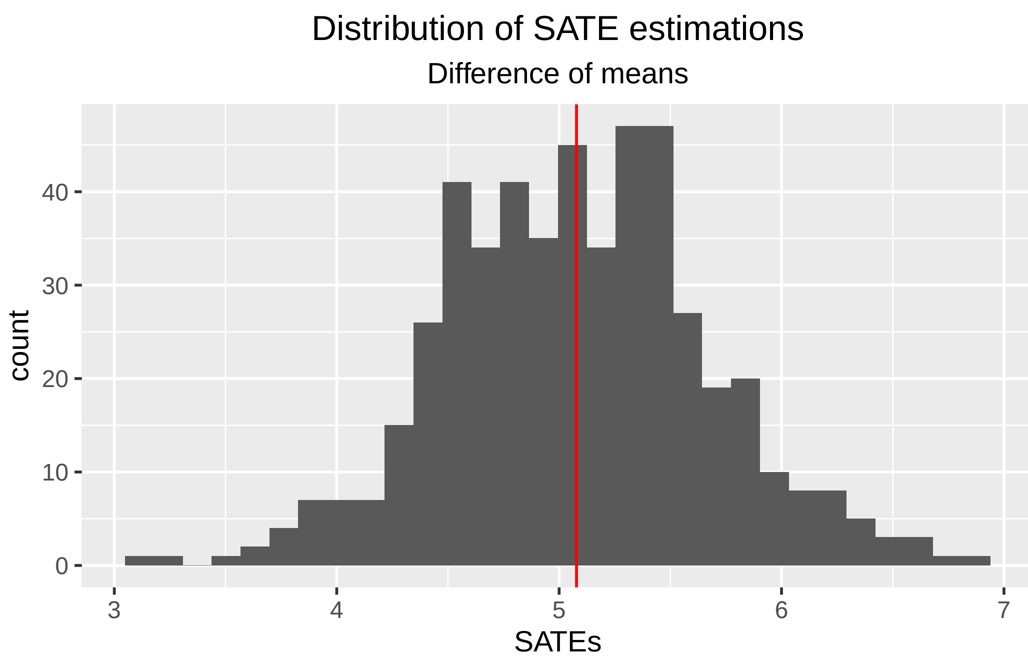
(c) $\widehat{\text{SATE}}$ distribution

```
SATEs <- c()

for (i in 1:500) {
  treatment <- sample(c(rep(0, N/2), rep(1, N/2)))
  y_obs_temp <- ifelse(treatment == 1, y1, y0)
  SATE_hat <- mean(y_obs_temp[treatment == 1]) - mean(y_obs_temp[treatment == 0])
  SATEs <- c(SATEs, SATE_hat)
}

SATEs_mean <- mean(SATEs)
SATEs_sd <- sd(SATEs)

p <- ggplot() +
  aes(SATEs) +
  geom_histogram() +
  geom_vline(xintercept=SATEs_mean, color='red') +
  labs(title="Distribution of SATE estimations",
       subtitle = "Difference of means") +
  theme(plot.title = element_text(hjust=0.5),
        plot.subtitle = element_text(hjust=0.5))
p
```




```
[1] "Mean: 5.078"
```

```
[1] "Standard Deviation: 0.596"
```

(d) $\widehat{\text{SATE}}$ using regressions

```
model_q1 <- lm(formula= y_obs ~ treatment + quiz1,
               data = df)
summary(model_q1)
```

```

Call:
lm(formula = y_obs ~ treatment + quiz1, data = df)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.35124 -0.58413 -0.08723  0.67581  2.44665 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 12.10847    2.42130   5.001 2.54e-06 ***
treatment    5.10753    0.19844  25.739  < 2e-16 ***
```

```
quiz1        1.06617    0.03701  28.810  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9922 on 97 degrees of freedom
Multiple R-squared:  0.9392,    Adjusted R-squared:  0.9379
F-statistic: 748.6 on 2 and 97 DF,  p-value: < 2.2e-16
```

```r
model_no_q1 <- lm(formula= y_obs ~ treatment,
                  data = df)
summary(model_no_q1)
```

```
Call:
lm(formula = y_obs ~ treatment, data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-8.2079 -1.8759  0.3782  1.9095  8.1017

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  81.7487     0.4316 189.427  < 2e-16 ***
treatment     5.1259     0.6103   8.399 3.55e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.052 on 98 degrees of freedom
Multiple R-squared:  0.4185,    Adjusted R-squared:  0.4126
F-statistic: 70.54 on 1 and 98 DF,  p-value: 3.551e-13
```

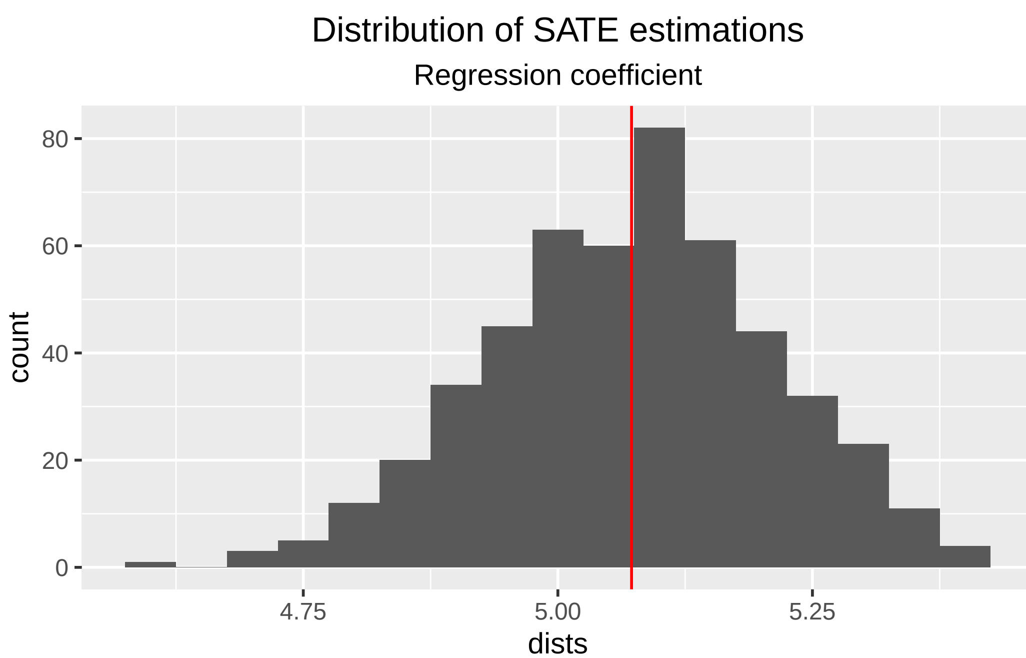
The estimate using the quiz 1 scores (5.107) is closer to the true SATE (5.067) that the estimate from the model not using the scores (5.125). My guess for why this happens is that in the data generation process the quiz 2 scores are indeed a function quiz 1 scores. The model that does not contain these scores is not able to see that, so it overestimates the effect that the treatment had on the higher scores of quiz 2.

(e) $\widehat{\text{SATE}}$ using regressions distribution

```r
dists <- c()

for (i in 1:500) {
  treatment <- sample(c(rep(0, N/2), rep(1, N/2)))
  y_obs_temp <- ifelse(treatment == 1, y1, y0)
  df_temp <- data.frame(treament = treatment,
                        y_obs = y_obs_temp,
                        quiz1 = q1_scores)
  model_temp <- lm(formula = y_obs ~ treatment + quiz1,
                   data = df_temp)
  dists <- c(dists, coef(model_temp)[2])
}

p_reg <- ggplot() +
  aes(dists) +
  geom_histogram(binwidth=0.05) +
  geom_vline(xintercept=mean(dists), color='red') +
  labs(title="Distribution of SATE estimations",
       subtitle = "Regression coefficient") +
  theme(plot.title = element_text(hjust=0.5),
        plot.subtitle = element_text(hjust=0.5))
p_reg
```



Looking at the distributions, the regression method for estimating SATE is better than the difference in means method based on their range. The regression method has values around 4.5 and 5.5 while the difference in means method has values between 3 and 7, showing that the latter can output values way off the actual SATE. Making the regression method more reliable.