Pset4

Gabriel Wang

December 8, 2025

1 Part 1: Reading

```
 # ------------------------
# 1. Create one simulated dataset and print summary
# ------------------------

n0 <- 1000   # any moderate sample size

C <- rnorm(n0)
X <- 0.6*C + rnorm(n0)
Y <- 1.0*X + 0.4*C + rnorm(n0)

# Fit the true data-generating model
true_model <- lm(Y ~ X + C)
stargazer(true_model)
> print(summary(true_model))

Call:
lm(formula = Y ~ X + C)

Residuals:
    Min      1Q  Median      3Q     Max
-3.2747 -0.6593 -0.0103  0.7197  3.1937

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.002381   0.031652   0.075     0.94
X           1.009568   0.031083  32.480   <2e-16 ***
C           0.326033   0.037769   8.632   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘’. 0.1 ‘’  1

Residual standard error: 1.001 on 997 degrees of freedom
Multiple R-squared:  0.6554,    Adjusted R-squared:  0.6547
F-statistic: 947.9 on 2 and 997 DF,  p-value: < 2.2e-16
```

Table 1:

	Dependent variable:
	Y
X	1.004***
	(0.011)
C	0.407***
	(0.013)
Constant	0.009
	(0.011)
Observations	8,000
R^2	0.677
Adjusted R^2	0.677
Residual Std. Error	0.986 (df = 7997)
F Statistic	8,380.917*** (df = 2; 7997)
Note:	*p<0.1; **p<0.05; ***p<0.01

Question a

```r
# Sample sizes we want to study
n_vals <- c(50, 200, 500, 1000)

# A data frame to store all simulation results
results <- data.frame()

# Number of simulations for each sample size
R <- 2000

for (n in n_vals) {

  # A vector to store the 2000 estimated coefficients for this n
  beta_hats <- numeric(R)

  for (r in 1:R) {

    # Confounder C
    C <- rnorm(n)

    # Treatment X depends on C (confounding relationship)
    X <- 0.6 * C + rnorm(n)

    # Outcome Y follows the true data-generating model:
    # Y = 1.0*X + 0.4*C + error
    Y <- 1.0 * X + 0.4 * C + rnorm(n)

    # Estimate the linear model and store only the coefficient on X  (1)
    beta_hats[r] <- coef(lm(Y ~ X + C))[2]
  }

  # Combine results into a long data frame for plotting
  results <- rbind(
    results,
    data.frame(
      n = rep(n, R),
      beta_hat = beta_hats
    )
  )
}

```

```r
# -----------------------
# Plot sampling distributions for different sample sizes
# -----------------------

library(ggplot2)

ggplot(results, aes(x = beta_hat)) +
  geom_density(fill = "skyblue", alpha = 0.5) +

  # One panel per sample size
  facet_wrap(~ n, scales = "free", ncol = 3) +

  # True value of  1 (red dashed line)
  geom_vline(xintercept = 1, color = "red", linetype = "dashed", size = 1) +

  theme_minimal() +
  labs(
    title = "Sampling Distribution of  1 as Sample Size Increases",
    subtitle = "Red dashed line = true  1 = 1.0",
    x = "Estimated  1",
    y = "Density"
  )

  ## proof of CLT
# Step 1: summarize simulated sampling distribution
results_summary <- results %>%
  group_by(n) %>%
  summarize(
    mean_beta = mean(beta_hat),
    sd_beta = sd(beta_hat)
  )

# Step 2: compute theoretical standard deviation = 1 / sqrt(n)
sd_theoretical <- data.frame(
  n = n_vals,
  sd_theoretical = 1 / sqrt(n_vals)
)

# Step 3: merge simulated results with theoretical SD
summary_combined <- results_summary %>%
  left_join(sd_theoretical, by = "n")

# Step 4: print with stargazer
stargazer(summary_combined,
          summary = FALSE,

          title = "Simulated vs. Theoretical Standard Deviations of Beta1",
          rownames = FALSE)
```

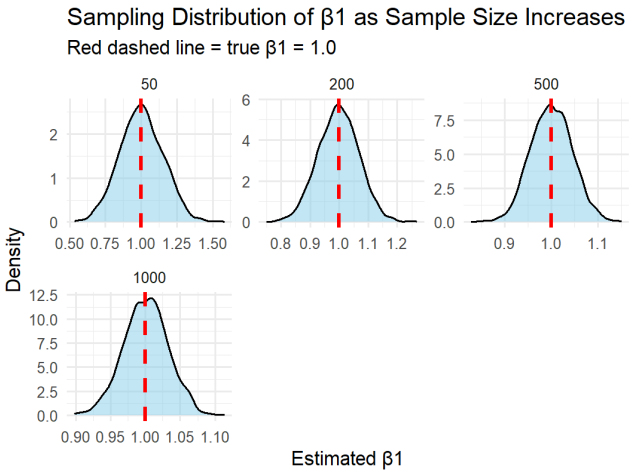

Figure 1: Distribution as N increases

Table 2: Simulated vs. Theoretical Standard Deviations of Beta1

n	mean_beta	sd_beta	sd_theoretical
50	0.999761980621645	0.144303258648783	0.14142135623731
200	0.999088603861918	0.0723473587392907	0.0707106781186548
500	1.00062695672353	0.0454422380872791	0.0447213595499958
1000	1.00023682102431	0.0314158881354247	0.0316227766016838

The simulated standard deviation *sd_beta* closely matches the theoretical value $\frac{1}{\sqrt{N}}$. The mean of the coefficient is also converging to 1. This confirms that the estimator's variance decreases at the rate predicted by the Central Limit Theorem.

Question b

```r
# ---------------------------
# 2. Bootstrap standard error
# ---------------------------

B <- 2000   # number of bootstrap replications
boot <- numeric(B)

for (b in 1:B) {

  # Resample rows with replacement
  resample <- sample(1:n, size = n, replace = TRUE)

  # Fit model on bootstrap sample
  boot_model <- lm(Y[resample] ~ X[resample] + C[resample])

  # Store coefficient on treatment variable X
  boot[b] <- coef(boot_model)[2]
}

# Bootstrapped standard error of  1
boot_se <- sd(boot)
boot_se
```

The boot se is 0.03130765.

Question c

```r
for (n in n_vals) {
```

```r

  # A vector to store the 2000 estimated coefficients for this n
  beta_hats_omitted <- numeric(R)

  for (r in 1:R) {

    # Confounder C
    C <- rnorm(n)

    # Treatment X depends on C (confounding relationship)
    X <- 0.6 * C + rnorm(n)

    # Outcome Y follows the true data-generating model:
    # Y = 1.0*X + 0.4*C + error
    Y <- 1.0 * X + 0.4 * C + rnorm(n)

    # Estimate the linear model and store only the coefficient on X  (1)
    beta_hats_omitted [r] <- coef(lm(Y ~ X))[2]
  }

  # Combine results into a long data frame for plotting
  results_omitted <- rbind(
    results_omitted,
    data.frame(
      n = rep(n, R),
      beta_hats_omitted = beta_hats_omitted
    )
  )
}

# ------------------------
# Plot sampling distributions for different sample sizes
# ------------------------

ggplot(results_omitted, aes(x = beta_hats_omitted)) +
  geom_density(fill = "skyblue", alpha = 0.5) +

  # One panel per sample size
  facet_wrap(~ n, scales = "free", ncol = 3) +

  # True value of  1 (red dashed line)
  geom_vline(xintercept = 1, color = "red", linetype = "dashed", size = 1) +

  theme_minimal() +
  labs(
    title = "Sampling Distribution of  1 with omitted confounding variable as Sample Size
        Increases",
    subtitle = "Red dashed line = true  1 = 1.0",
    x = "Estimated  1 with omitted confounding variable",
    y = "Density"
  )
```

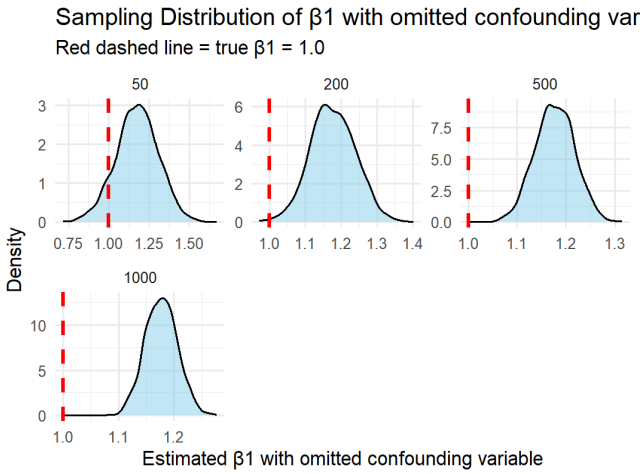

Figure 2: Omitting Confounders

When the confounding variable C is omitted, the sampling distribution of the treatment coefficient becomes centered around a biased value rather than the true parameter $\beta_1 = 1$. In the data-generating process,

$$X = 0.6C + u, \qquad Y = 1.0X + 0.4C + \varepsilon,$$

the omitted-variable bias formula is

$$\hat{\beta}_1^{omit} = \beta_1 + \beta_2 \frac{\mathrm{Cov}(X, C)}{\mathrm{Var}(X)}.$$

Since $\beta_1 = 1$, $\beta_2 = 0.4$, and $\frac{\mathrm{Cov}(X,C)}{\mathrm{Var}(X)} = 0.6$, the expected biased coefficient is

$$\hat{\beta}_1^{omit} = 1 + 0.4(0.6) = 1.24.$$

This matches the simulation results: the sampling distribution is centered around approximately 1.24 instead of 1.0. Although the distribution becomes narrower as n increases, it converges to the *wrong* value.

This implies that statistical tests based on this biased sampling distribution (t-tests, confidence intervals, p-values) will be misleading. The estimator becomes precise but incorrect, showing why the confounder must be included for valid inference.

2 Part 2: Simulation

Question 1

```
data <- read_dta("C:/WindowsD/NYU/summer/quant1/lab/assignment/ 9/gss.dta")
View(data)
library(stargazer)
library(sandwich)
library(tidyverse)
data$educ
dim(data)
# if the individual's highest schoor year completed is larger than 16, I code him
    higheduc =1
data$higheduc <- ifelse(data$educ > 16, 1, 0)
range(data$educ)
difference_mean <- t.test(gun ~ higheduc, data )
print(difference_mean)
> print(difference_mean)

        Welch Two Sample t-test

data:  gun by higheduc
t = -0.99138, df = 277.6, p-value =
0.3224
alternative hypothesis: true difference in means between group 0 and group 1 is not equal
     to 0
95 percent confidence interval:
```

```
 -0.10560930  0.03486502
sample estimates:
mean in group 0 mean in group 1
      0.3372694       0.3726415
```

Using the 2018 General Social Survey, I test whether gun ownership differs between highly educated individuals (16+ years of schooling) and others.

Hypotheses:

$$H_0 : \mu_{\text{high}} = \mu_{\text{low}}, \qquad H_1 : \mu_{\text{high}} \neq \mu_{\text{low}}.$$

Choice of test: I use a two-sample t-test because the population variance is unknown. Although the sample size is large, the t-test is the standard approach and is asymptotically equivalent to the z-test.

Significance level: $\alpha = 0.05$.

Results: The test yields a p-value of 0.3224. Since $p > 0.05$, I fail to reject the null hypothesis.

Interpretation: Statistically, there is no evidence of a difference in mean gun-ownership rates across high and low education groups. Substantively, education does not appear to predict gun ownership in this sample.

Question 2

```

m_lpm <- lm(data = data, gun ~ educ + income + educ:income)
# Fit a Linear Probability Model (LPM) predicting 'gun' using:
#   - educ: education level
#   - income: respondent income
#   - educ:income: interaction between education and income
# The model estimates how education, income, and their interaction affect gun policy
    support.

vm_lpm <- vcovHC(m_lpm, type = 'HC1')
# Compute heteroskedasticity-consistent (HC1) robust -variancecovariance matrix
# This generates robust standard errors to correct for heteroskedasticity in the LPM.

stargazer(m_lpm, se = list(sqrt(diag(vm_lpm))))
# Produce a formatted regression table using stargazer.
# The 'se' option replaces the default standard errors with the robust SEs
# (square roots of the diagonal elements of vm_lpm).

# calculate the pvalue
t_values <- coef(m_lpm) / sqrt(diag(vm_lpm))
print(t_values)
> print(t_values)
(Intercept)        educ      income
  0.5260547   2.8729479   4.6404480
educ:income
 -3.5376800
```

Table 3:

	Dependent variable:
	gun
educ	0.015^{***}
	(0.005)
income	0.005^{***}
	(0.001)
educ:income	-0.0003^{***}
	(0.0001)
Constant	0.037
	(0.071)
Observations	1,443
R^2	0.043
Adjusted R^2	0.041
Residual Std. Error	0.467 (df = 1439)
F Statistic	21.564^{***} (df = 3; 1439)
Note:	*p<0.1; **p<0.05; ***p<0.01

Using the same data, I fit a linear probability model predicting gun ownership from education, income, and their interaction.

Education

The coefficient for education is 0.015, meaning that each additional year of schooling is associated with a 1.5 percentage-point increase in the probability of owning a gun, holding income constant. The standard error is 0.005, indicating a very precise estimate. The t-value (approximately 2.87) implies that the coefficient is statistically different from zero. The p-value is below 0.01, so the effect is statistically significant.

Income

The coefficient for income is 0.005, meaning that higher income is associated with a higher probability of gun ownership, all else equal. The standard error is 0.001, also indicating high precision. The t-value (about 4.64) shows strong statistical significance, and the p-value is below 0.01.

Interaction: Education × Income

The interaction term has a coefficient of –0.0003. This negative sign indicates that the effect of education on gun ownership becomes weaker as income increases, and similarly, the effect of income becomes weaker at higher levels of education. The standard error is 0.0001, suggesting a precise estimate. The t-value (|t| 3.54) shows that the coefficient differs significantly from zero, and the p-value is below 0.01, confirming statistical significance.