

# ECON 7201

## Applied Econometrics

### Assignment 2

**Due Date** Tarik Aziz [Student ID# 3190009]

Sunday October 5, 2025 at 11:59 PM

#### Directions

Answer all questions. Submit both a PDF and Quarto file to the nexus assignment portal.

#### Git and GitHub

- (a) Create a new R project in your **econ\_3201** directory called **assignment\_2**.  
(b) Download the assignment PDF and Quarto file the **assignment\_2** folder.  
(c) Commit and push the changes to your **econ\_3201** repository on [GitHub.com](https://github.com).

#### LaTeX

Matrices are created in LaTeX using the `\begin{bmatrix}...\end{bmatrix}` command. To separate entries along the same row, use `&`. To end a line, use `\\`. To make vertical ellipses ( $\vdots$ ), use `\vdots`. Practice writing the following matrices and vectors in LaTeX. Write the following matrices in LaTeX.

$$2. \quad (a) \quad X'X = \begin{bmatrix} n & \sum_{i=1}^n x_{1i} & \sum_{i=1}^n x_{2i} \\ \sum_{i=1}^n x_{1i} & \sum_{i=1}^n x_{1i}^2 & \sum_{i=1}^n x_{1i}x_{2i} \\ \sum_{i=1}^n x_{2i} & \sum_{i=1}^n x_{1i}x_{2i} & \sum_{i=1}^n x_{2i}^2 \end{bmatrix}$$
$$(b) \quad \Omega = \begin{bmatrix} \sigma_1^2 & 0 & 0 & 0 \\ 0 & \sigma_2^2 & 0 & 0 \\ 0 & 0 & \sigma_3^2 & 0 \\ 0 & 0 & 0 & \sigma_4^2 \end{bmatrix}$$

## R

3. In this question we compare standard errors based on (incorrect) asymptotic assumptions with those based on alternate (appropriate) estimator (White). Consider one sample drawn from the following data generating process (DGP) which we will simulate in R:

```
set.seed(123)
n <- 25
x <- rnorm(n,mean=0.0,sd=1.0)
beta0 <- 1
beta1 <- 0
## x is irrelevant in this model, the data generating process is as follows:
dgp <- beta0 + beta1*x
## The residual is heteroskedastic by construction
e <- x^2*rnorm(n,mean=0.0,sd=1.0)
y <- dgp + e
```

- (a) Compute the OLS estimator of  $\beta_2$  and its standard error using the `lm()` command in R for the model  $y_i = \beta_1 + \beta_2 x_i + \epsilon_i$  based on the DGP given above.

```
model <- lm(y~x)

beta <- model$coefficients
```

```
model_sum <- summary(model)

se_beta2 <- model_sum$coefficients["x","Std. Error"]

se_beta2
```

```
[1] 0.2563086
```

- (b) Next, compute the standard error of  $\hat{\beta}_2$  by computing  $\hat{\sigma}^2(X'X)^{-1}$  in R using matrix commands, and verify that the two standard error estimates are identical.

```
#computing sigma_hat_sq
ehat <- model$residuals

sigma_hat_sq <- sum(ehat^2)/(n-2)
```

```
se2_beta2 <- sqrt(sigma_hat_sq*solve(t(x)%*%x))
se2_beta2
```

```
      [,1]
[1,] 0.2561433
```

```
all.equal(se_beta2, as.numeric(se2_beta2))
```

```
[1] "Mean relative difference: 0.000644915"
```

- (c) Compute White's heteroskedasticity consistent covariance matrix estimator using matrices in R and report the White estimator of the standard error of  $\hat{\beta}_2$ . Compare this with that from 3 (a) above.

```
S <- diag(ehat^2)
cov_white <- solve(t(x)%*%x) %*% (t(x)%*%S%*%x) %*% solve(t(x)%*%x)
```

```
se_white <- sqrt(diag(cov_white))
se_white
```

```
[1] 0.4554561
```

Comment: White Std. Error is nearly twice as large as Std. Error in 3(a)

4. Let  $\hat{\theta}$  be an estimator for the population parameter  $\theta$ .  $\hat{\theta}$  is said to be unbiased if  $E(\hat{\theta}) = \theta$ . That is, if the mean of the sampling distribution of  $\hat{\theta}$  is equal to the true population value.

Consider the model

$$y_i = \beta_0 + \beta_1 x_{1,i} - \beta_2 x_{2,i} + \epsilon_i.$$

Lets provide empirical evidence that the ordinary least squares estimators  $\hat{\beta}_0$ ,  $\hat{\beta}_1$ , and  $\hat{\beta}_2$  are unbiased estimators of  $\beta_0$ ,  $\beta_1$ ,  $\beta_2$ , respectively, using R.

- (a) Set the seed to 1, i.e., `set.seed(1)`.

```
set.seed(1)
```

- (b) Set the number of observations `$n=100$`

```
n <- 100
```

(c) Generate the following model  $y_i = 2 + 3.5x_{1,i} - 9.2x_{2,i} + \epsilon_i$  where  $x_1 \sim N(3, 6)$ ,  $x_2 \sim N(2, 4)$ , and  $\epsilon \sim N(0, 100)$ . To create a normally distributed variable, use the `rmnorm(n, mean, sd)` command in R.

```
x1 <- rmnorm(n, mean = 3, sd = sqrt(6))
x2 <- rmnorm(n, mean = 2, sd = sqrt(4))
e <- rmnorm(n, mean = 0, sd = sqrt(100))

y <- 2+3.5*x1-9.2*x2+e
```

(d) Estimate the model coefficients using the `lm()` command. (Search `?lm()` in the console for more info).

```
lm(y ~ x1+x2)$coefficients
```

```
(Intercept)          x1          x2
  2.529657    3.586182   -9.467334
```

(e) Using a `for()` loop, replicate the model above  $M=1000$  times and save the coefficient estimates from each iteration.

```
beta0 <- 2
beta1 <- 3.5
beta2 <- 9.2
beta<- c(beta0,beta1,beta2)

x1 <- rmnorm(n, mean = 3, sd = sqrt(6))
x2 <- rmnorm(n, mean = 2, sd = sqrt(4))
X <- cbind(1,x1,x2)

e <- rmnorm(n, mean = 0, sd = sqrt(100))
y<- X%*%beta + e

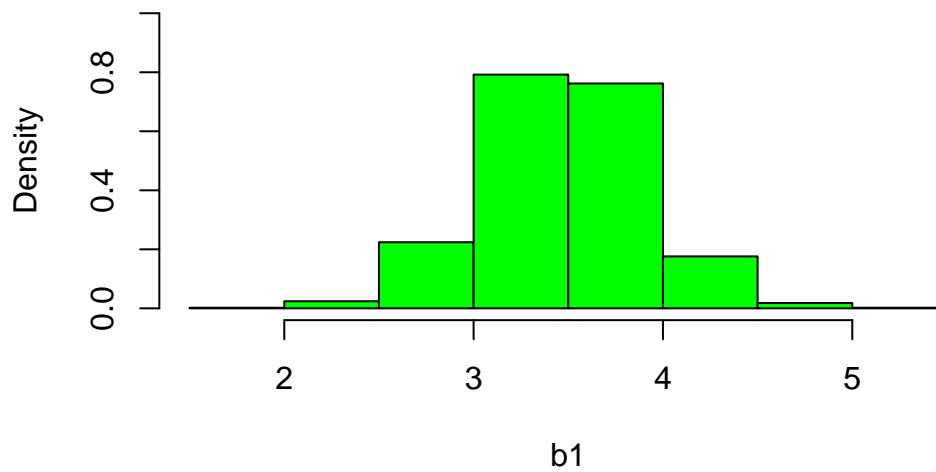
M <- 1000
beta0_hat <- numeric(M)
beta1_hat <- numeric(M)
beta2_hat <- numeric(M)
```

```
for (i in 1:M) {  
  x1 <- rnorm(n, mean = 3, sd = sqrt(6))  
  x2 <- rnorm(n, mean = 2, sd = sqrt(4))  
  X <- cbind(1,x1,x2)  
  
  e <- rnorm(n, mean = 0, sd = sqrt(100))  
  y<- X%%beta + e  
  
  beta_hat <-solve(t(X)%*%X)%*%t(X)%*%y  
  
  beta0_hat[i] <- beta_hat[1]  
  beta1_hat[i] <- beta_hat[2]  
  beta2_hat[i] <- beta_hat[3]  
}
```

(f) Using `hist()`, plot the sampling distributions of the coefficient estimates, `$\beta_1$` and `$\beta_2$`.

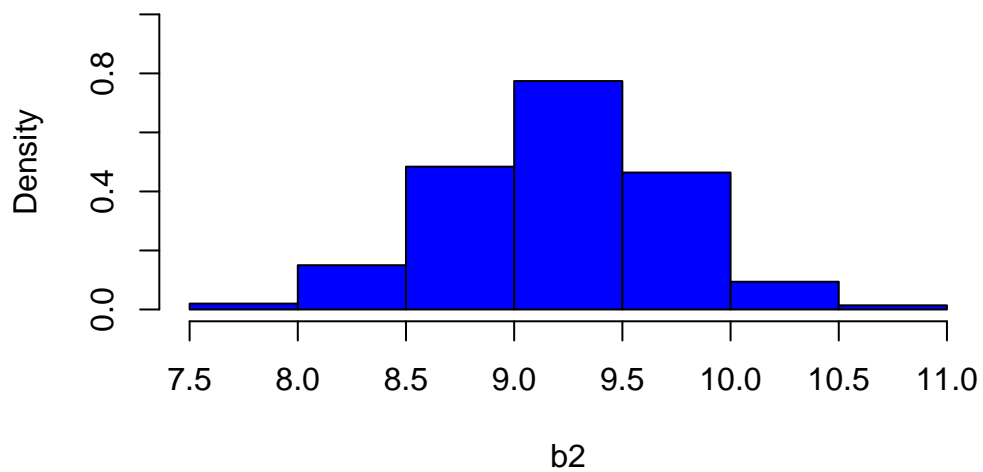
```
# Histogram of beta1_hat  
hist(beta1_hat,  
      probability = T,  
      ylim = c(0,1),  
      main = "Histogram of b1 Estimates",  
      col = "green",  
      xlab = "b1")
```

### Histogram of b1 Estimates



```
# Histogram of beta2_hat  
hist(beta2_hat,  
      probability = T,  
      ylim = c(0,1),  
      main = "Histogram of b2 Estimates",  
      col = "blue",  
      xlab = "b2")
```

### Histogram of b2 Estimates

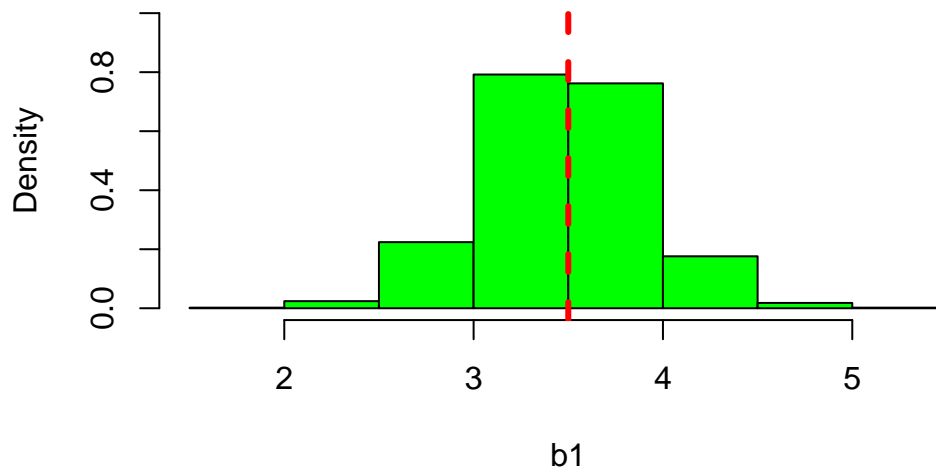


(g) Add a vertical line to each figure at the mean of the respective variable.

Search `?abline()` in your console.

```
# Histogram of beta1_hat
hist(beta1_hat,
      probability = T,
      ylim = c(0,1),
      main = "Histogram of b1 Estimates",
      col = "green",
      xlab = "b1")
abline(v=3.5,
       col="red",
       lwd=3,
       lty="dashed")
```

**Histogram of b1 Estimates**



```
# Histogram of beta2_hat
hist(beta2_hat,
      probability = T,
      ylim = c(0,1),
      main = "Histogram of b2 Estimates",
      col = "blue",
      xlab = "b2")
abline(v=9.2,
       col="red",
       lwd=3,
       lty="dashed")
```

### Histogram of b2 Estimates

