Statistical Theory Homework 1

Bohao Tang

1: $Y = 2 + 0.5Z + X + ZX + \varepsilon$

(a) $E(Y \mid Z = z, X = x] = E_\varepsilon[2 + 0.5z + x + zx + \varepsilon]$

$= 2 + 0.5z + x + zx$

(b) $var(Y \mid Z = z, X = x) = var_\varepsilon[2 + 0.5z + x + zx + \varepsilon]$

$= var_\varepsilon[\varepsilon] = \sigma_\varepsilon^2$

(c) $Y \mid X = x, Z = z \sim N(2 + 0.5z + x + zx, \sigma_\varepsilon^2)$

$Pr(Y = y \mid Z = z, X = x) = \frac{1}{\sqrt{2\pi}\,\sigma_\varepsilon} e^{-(y - 2 - 0.5z - x - zx)^2 / 2\sigma_\varepsilon^2}$

(d) $E(Y \mid Z = 1, X = x) - E(Y \mid Z = 0, X = x] = 0.5 + x$

therefore $E_X\{E[Y \mid Z = 1, X] - E[Y \mid Z = 0, X]\} = 0.5 + EX$

$= 0.5$.

(e) $E[Y \mid Z = 1] = E_{X,\varepsilon}[2 + 0.5 + X + X + \varepsilon]$

$= 2.5$

$E[Y \mid Z = 0] = E_{X,\varepsilon}[2 + X + \varepsilon] = 2$

$\Rightarrow E[Y \mid Z = 1] - E[Y \mid Z = 0] = 0.5$

2:

(a): $E[Y|Z,X] = 2+0.5Z + X + ZX = \beta_0 + \beta_1 Z + \beta_2 X$

$$\Rightarrow \begin{cases} Z=1: & 2.5+2X = \beta_0+\beta_1+\beta_2 X \Rightarrow \beta_2 = 2 \\ Z=0: & 2+X = \beta_0 + \beta_2 X \Rightarrow \beta_2 = 1 \end{cases}$$

Contradiction

Therefore this model misspecified the distribution

(b) Then $2+0.5Z+X+ZX = \beta_0 + \beta_1 Z + \beta_2 ZX + \beta_3 X^2$

$$\Rightarrow \begin{cases} Z=1: & 2.5+2X = \beta_0+\beta_1+\beta_2 X + \beta_3 X^2 \Rightarrow \beta_3=0,\ \beta_2=2 \\ Z=0: & 2+X = \beta_0 + \beta_3 X^2 \quad \text{can't hold.} \end{cases}$$

So the model mispecified the distribution.

3:

We do $Y = 1+2A+3\sin(B)+5\sin(AB)+\varepsilon$

where A, B, ε mutal independent and $B, \varepsilon \sim N(0,1)$

Then since $N(0,1)$ is symmetric and $\sin(x)$ is odd function we have $E\sin(B)=0$

Then $E(Y|A=1) - E(Y|A=0) = 2$

See the coding part.

Coding Part

Bohao Tang

April 1, 2018

3b

True distribution is

$$Y = 1 + 2A + 3sin(B) + 5sin(AB) + \epsilon$$

Where A, B, ϵ mutual independent and $\epsilon, B \sim N(0, 1)$

```
model = function(A, B, eps){
    1 + 2*A + 3*sin(B) + 5*sin(A*B) + eps
}
```

Then

```
n = 1000

A = rbinom(n, 1, 0.5)
B = rnorm(n, 0, 1)
eps = rnorm(n, 0, 1)

Y = model(A, B, eps)

unadj.estimator = mean(Y[A==1]) - mean(Y[A==0])
print(unadj.estimator)
```

```
## [1] 2.319556
```

3c

Then $E_P[Y|A = 1] - E_P[Y|A = 0]$ is just β_1

Use the data above

```
data = data.frame(A=A, B=B, Y=Y)
fit = lm(Y ~ A + B, data = data)
summary(fit)
```

```
##
## Call:
## lm(formula = Y ~ A + B, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5427  -1.4253   0.0021   1.5819   9.9991
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.10515    0.10618   10.41   <2e-16 ***
## A            2.05957    0.15268   13.49   <2e-16 ***
## B            3.09277    0.07374   41.94   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.411 on 997 degrees of freedom
## Multiple R-squared:  0.6662, Adjusted R-squared:  0.6656
## F-statistic: 995.1 on 2 and 997 DF,  p-value: < 2.2e-16
```

We can get the result as

```
print(fit$coefficients[2])
```

```
##        A
## 2.059565
```

3d

```
unadjusted.estimator = c()
ancova.estimator = c()

n = 1000
for(i in 1:10000){
    A = rbinom(n, 1, 0.5)
    B = rnorm(n, 0, 1)
    eps = rnorm(n, 0, 1)

    Y = Y = model(A, B, eps)
    datas = data.frame(A=A, B=B, Y=Y)

    ue = mean(Y[A==1]) - mean(Y[A==0])
    unadjusted.estimator = c(unadjusted.estimator, ue)

    fit = lm(Y~A+B, data=datas)
    ancova.estimator = c(ancova.estimator, fit$coefficients[2])
}
```

3d.i.

We have:

```
unadjusted.mean = mean(unadjusted.estimator)
print(unadjusted.mean)
```

```
## [1] 1.994968
```

```
unadjusted.var = var(unadjusted.estimator)
print(unadjusted.var)
```

```
## [1] 0.06785065
```

```
ancova.mean = mean(ancova.estimator)
print(ancova.mean)
```

```
## [1] 1.997695
```

```
ancova.var = var(ancova.estimator)
print(ancova.var)
```

```
## [1] 0.02233898
```

3d.ii.

We have

```r
unadjusted.bias = unadjusted.mean - 1
print(unadjusted.bias)
```

```
## [1] 0.9949675
```

```r
ancova.bias = ancova.mean - 1
print(ancova.bias)
```

```
## [1] 0.9976953
```

3d.iii.

We have

```r
relative.efficiency = unadjusted.var / ancova.var
print(relative.efficiency)
```

```
## [1] 3.03732
```

3d.iv.

We have

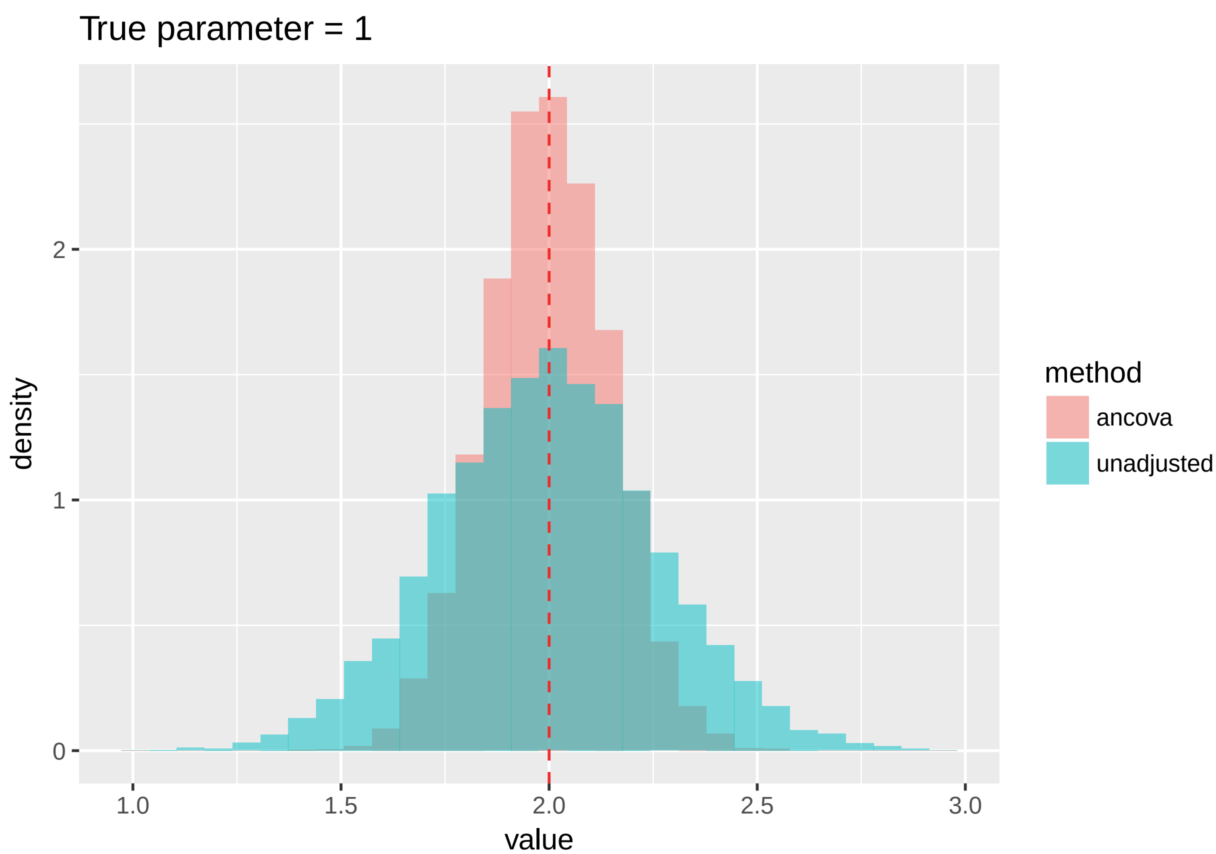
```r
library(ggplot2)

unadjusted = data.frame(value = unadjusted.estimator)
unadjusted$method = "unadjusted"

ancova = data.frame(value = ancova.estimator)
ancova$method = "ancova"

estimators = rbind(unadjusted, ancova)

ggplot(estimators, aes(value, fill = method)) +
    geom_histogram(alpha = 0.5, aes(y = ..density..), position = 'identity') +
    geom_vline(xintercept = 2, linetype="dashed", color="firebrick2") +
    ggtitle("True parameter = 1")
```



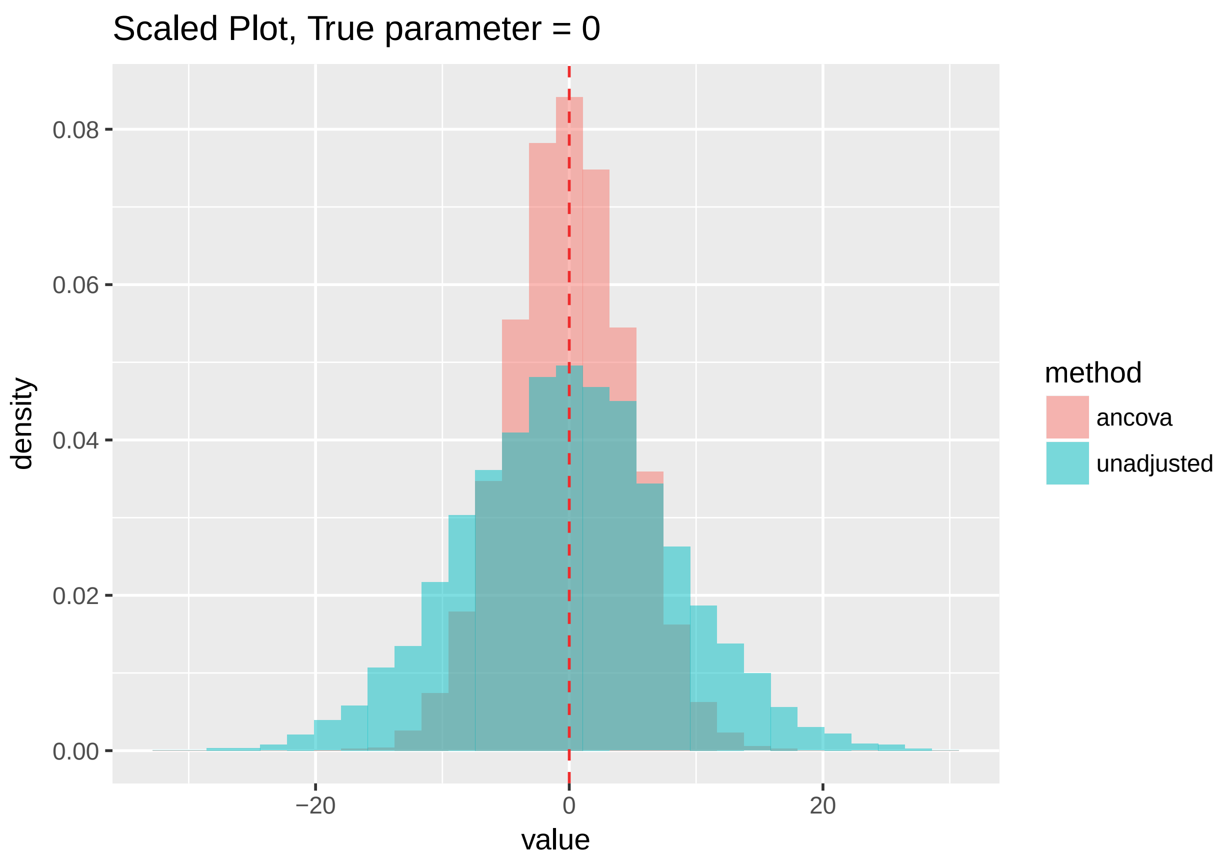
and

```
unadjusted.scaled = data.frame(value = sqrt(n) * (unadjusted.estimator - 2))
unadjusted.scaled$method = "unadjusted"

ancova.scaled = data.frame(value = sqrt(n) * (ancova.estimator - 2))
ancova.scaled$method = "ancova"

estimators.scaled = rbind(unadjusted.scaled, ancova.scaled)

ggplot(estimators.scaled, aes(value, fill = method)) +
    geom_histogram(alpha = 0.5, aes(y = ..density..), position = 'identity') +
    geom_vline(xintercept = 0, linetype="dashed", color="firebrick2") +
    ggtitle("Scaled Plot, True parameter = 0")
```


Scaled Plot, True parameter = 0
density
0.08
0.06
0.04
0.02
0.00
−20
0
20
value
method
ancova
unadjusted