Homework 5 Solution

1. (a) Since $\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$,

$$\begin{aligned}\mathbf{X}'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) &= \mathbf{X}'\mathbf{Y} - \mathbf{X}'\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}'\mathbf{Y} - \mathbf{X}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y} = \mathbf{X}'\mathbf{Y} - \mathbf{X}'\mathbf{Y} \\ &= \mathbf{0}\end{aligned}$$

(b) Note that $\sum \hat{e}_i = \mathbf{1}'\hat{\mathbf{e}}$, where $\mathbf{1}$ is an $n \times 1$ vector of 1's, and from part (a),

$$\mathbf{X}'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) = \mathbf{X}'\hat{\mathbf{e}} = \mathbf{0}$$

Now, the first column of $\mathbf{X}$ is $\mathbf{1}$, so that the row of $\mathbf{X}'$ is $\mathbf{1}'$, and since $\mathbf{X}'\hat{\mathbf{e}} = \mathbf{0}$, it follows that the inner product of first row of $\mathbf{X}'$ and $\hat{\mathbf{e}}$ is 0, i.e.

$$\sum \hat{e}_i = \mathbf{1}'\hat{\mathbf{e}} = 0$$

2. Since $\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$, and $\mathrm{Var}(\mathbf{Y}) = \mathrm{Var}(\mathbf{e}) = \mathbf{V}$ (why?), we have that

$$\begin{aligned}\mathrm{Var}(\hat{\boldsymbol{\beta}}) &= \mathrm{Var}((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}) \\ &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathrm{Var}(\mathbf{Y})\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1} \\ &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{V}\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\end{aligned}$$

Unfortunately, it does not simplify any further unless $\mathbf{V}$ has special properties.

3. (a) Using the hint,

$$\begin{aligned}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) &= (\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{X}\hat{\boldsymbol{\beta}} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{X}\hat{\boldsymbol{\beta}} - \mathbf{X}\boldsymbol{\beta}) \\ &= (\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}})'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \\ &\quad + (\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}})'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) + (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) \\ &= (\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}})'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \\ &\quad + 2(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) \\ &= (\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}})'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \\ &\quad + 2(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'(\mathbf{X}'\mathbf{Y} - \mathbf{X}'\mathbf{X}\hat{\boldsymbol{\beta}}) \\ &= (\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}})'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \\ &\quad + 2(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'(\mathbf{X}'\mathbf{Y} - \mathbf{X}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}) \\ &= (\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}})'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \\ &\quad + 2(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'(\mathbf{X}'\mathbf{Y} - \mathbf{X}'\mathbf{Y}) \\ &= (\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}})'(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})\end{aligned}$$

(b) Since the second term on the right hand side $(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})$ is quadratic, it follows that $(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \geq 0$. Therefore, this term is smallest (i.e., zero) when $\boldsymbol{\beta} = \hat{\boldsymbol{\beta}}$, at which point $(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$ minimized.

4. First, set x and y from the `Heights` dataset

```
y<-Heights$dheight
x<-Heights$mheight
```

(a) The regression summary is given by

```
> heights.lm<-lm(y~x)
> summary(heights.lm)

Call:
lm(formula = y ~ x)

Residuals:
   Min     1Q Median     3Q    Max
-7.397 -1.529  0.036  1.492  9.053

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 29.91744    1.62247   18.44   <2e-16 ***
x            0.54175    0.02596   20.87   <2e-16 ***
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 2.266 on 1373 degrees of freedom
Multiple R-squared:  0.2408,Adjusted R-squared:  0.2402
F-statistic: 435.5 on 1 and 1373 DF,  p-value: < 2.2e-16
```

(b) We can use, for example, the following to verify the numbers above

```
n<-nrow(Heights)
X<-matrix(c(rep(1,n),x),ncol=2)
bhat<-solve(t(X)%*%X)%*%t(X)%*%y
beta0hat<-bhat[1,1]
beta1hat<-bhat[2,1]

yhat<-X%*%bhat
ehat<-y-yhat
rss<-t(ehat)%*%(ehat)
t(y)%*%y - t(yhat)%*%yhat
sigma2hat<-rss/(n-1-1)
varhatbhat<-sigma2hat[1,1]*solve(t(X)%*%X)
se0<-sqrt(varhatbhat[1,1])
se1<-sqrt(varhatbhat[2,2])

syy<-sum((y-mean(y))^2)
ssreg<-syy-rss
fstat<-ssreg/sigma2hat

r2<-1-rss/syy
adjr2<-1-(1-r2)*(n-1)/(n-1-1)
```

(c) From Problem 2.13.3, we obtained the result with

```
> m1 <- lm(dheight ~ mheight, data=Heights)
> predict(m1, data.frame(mheight=64), interval="prediction", level=.99)
       fit      lwr      upr
1 64.58925 58.74045 70.43805
```

With R, we can run, for example,

```
sxx<-sum((x-mean(x))^2)
ystar<-beta0hat+beta1hat*64
sepred<-sqrt(sigma2hat*(1+1/n+(64-mean(x))^2/sxx))
qt(0.005,n-2,lower.tail=F)

ystar-qt(0.005,n-2,lower.tail=F)*sepred
ystar+qt(0.005,n-2,lower.tail=F)*sepred
```

and obtain a 99% prediction interval for a daughter whose mother is 64 inches tall as

$$(58.74045,\ 70.43805)$$

and $y_* = 64.58925$.

5. R codes are given by (check all intermediate numbers)

```
bgsg2.lm<-lm(BMI18~WT9+ST9, data=BGSgirls)
summary(bgsg2.lm)

y<-BGSgirls$BMI18
x1<-BGSgirls$WT9
x2<-BGSgirls$ST9

n<-nrow(BGSgirls)
p<-2

X<-matrix(c(rep(1,n),x1,x2),ncol=p+1)
bhat<-solve(t(X)%*%X)%*%t(X)%*%y
yhat<-X%*%bhat
ehat<-y-yhat
summary(ehat)

rss<-t(ehat)%*%(ehat)
t(y)%*%y - t(yhat)%*%yhat
sigma2hat<-rss/(n-p-1)
sqrt(sigma2hat)

varhatbhat<-sigma2hat[1,1]*solve(t(X)%*%X)
sebhat<-sqrt(diag(varhatbhat))
tstat<-bhat/sebhat
pt(abs(tstat),df=n-p-1,lower.tail=F)*2
```

```
syy<-sum((y-mean(y))^2)
ssreg<-syy-rss
r2<-1-rss/syy
adjr2<-1-(rss/(n-p-1))/(syy/(n-1))
1-(1-r2)*(n-1)/(n-p-1)

fstat<-(ssreg/p)/sigma2hat
pf(fstat,p,n-p-1,lower.tail = F)
```

Now, for the prediction, you can either do

```
predict(bgsg2.lm, newdata=data.frame(WT9=c(30,20,50,25,35),ST9=c(60,20,100,80,45)),
        interval="prediction", level=.95)
```

or manually (one set of input at a time)

```
xstar<-c(1, 30, 60)
ystar<-t(xstar)%*%bhat
ystar
sepred<-sqrt(sigma2hat*(1+t(xstar)%*%solve(t(X)%*%X)%*%xstar))
ystar-qt(0.025,n-p-1,lower.tail=F)*sepred
ystar+qt(0.025,n-p-1,lower.tail=F)*sepred
```

You can have multiple sets of inputs in this way, but this is a challenge.

6. R codes are given by (again check all intermediate numbers)

```
water.lm<-lm(BSAAM~APMAM+APSAB+APSLAKE+OPBPC+OPRC+OPSLAKE,data=water)
summary(water.lm)

y<-water$BSAAM
x1<-water$APMAM
x2<-water$APSAB
x3<-water$APSLAKE
x4<-water$OPBPC
x5<-water$OPRC
x6<-water$OPSLAKE

n<-nrow(water)
p<-6

X<-matrix(c(rep(1,n),x1,x2,x3,x4,x5,x6),ncol=p+1)
bhat<-solve(t(X)%*%X)%*%t(X)%*%y
yhat<-X%*%bhat
ehat<-y-yhat
summary(ehat)

rss<-t(ehat)%*%(ehat)
t(y)%*%y - t(yhat)%*%yhat
sigma2hat<-rss/(n-p-1)
```

```
sqrt(sigma2hat)

varhatbhat<-sigma2hat[1,1]*solve(t(X)%*%X)
sebhat<-sqrt(diag(varhatbhat))
tstat<-bhat/sebhat
pt(abs(tstat),df=n-p-1,lower.tail=F)*2

syy<-sum((y-mean(y))^2)
ssreg<-syy-rss
r2<-1-rss/syy
adjr2<-1-(rss/(n-p-1))/(syy/(n-1))
1-(1-r2)*(n-1)/(n-p-1)

fstat<-(ssreg/p)/sigma2hat
pf(fstat,p,n-p-1,lower.tail = F)
```