Problem Sheet 4

Theory of Inference

Dom Hutchinson

```r
set.seed(16111998)
```

Question 1

```r
conf <- read.table("data/confound.txt")

# Instrumental variables
Zx<-fitted(lm(x~v+w-1,data=conf))
Zz<-fitted(lm(z~v+w-1,data=conf))

m<-lm(y~Zx+Zz,data=conf)
summary(m)
```

```
##
## Call:
## lm(formula = y ~ Zx + Zz, data = conf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -40.784  -9.510  -0.189   9.340  44.697
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.6192     0.5318  -1.164   0.2448
## Zx            1.2302     0.9700   1.268   0.2052
## Zz            1.8345     0.8419   2.179   0.0297 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.55 on 647 degrees of freedom
## Multiple R-squared:  0.03251,    Adjusted R-squared:  0.02952
## F-statistic: 10.87 on 2 and 647 DF,  p-value: 2.27e-05
```

$\beta_1 = 1$ and $\beta_2 = 2$.

Question 2

Instrumental variables should be independent of the random error related to the observed variables. However, by generating our instrumental variable using the observed data this no longer holds & we have failed to break the correlation with the hidden random error.

Question 3

$RSE=\sqrt{RSS/\text{df}} \implies RSS=\text{df}\cdot RSE^2$. Thus
$RSS_0=98\cdot(0.3009)^2=8.8729994$.
$RSS_1=95\cdot(0.3031)^2=8.7276129$

Part a)

$$\begin{aligned} F &= \frac{(RSS_0-RSS_1)/q}{RSS_1/(n-p)} \\ &= \frac{(8.8729994-8.7276129)/(98-95)}{8.7276129/95} \\ &= 0.5275101 \end{aligned}$$

Part b)

$$p=\mathbb{P}(F_{3,95}>0.5275101)=0.6644552$$

Part c)

Question 4

Part a)

```
X<-model.matrix(~cars$speed+I(cars$speed^2)) # 1 s s^2
y<-cars$dist
head(X)
```

```
##   (Intercept) cars$speed I(cars$speed^2)
## 1           1          4              16
## 2           1          4              16
## 3           1          7              49
## 4           1          7              49
## 5           1          8              64
## 6           1          9              81
```

Part b)

```
# p=3, n=50
qrx<-qr(X) # QR decomposition
Q  <-qr.Q(qrx,complete=TRUE) # extract Q, nxn orthogonal matrix
R  <-qr.R(qrx) # extract R, pxp upper triangular matrix
```

Part c)

```
all.equal(
   t(Q),
   solve(Q)
   )
```

```
## [1] TRUE
```

```
x<-runif(dim(Q)[1]) # Generate random n matrix
all.equal(
```

```
    sum((Q%*%x)^2),
    sum(x^2)
    )
```

```
## [1] TRUE
```

Part d)

```
n=dim(Q)[1]; p=dim(R)[1]
f=head(t(Q)%*%y,p)
r=tail(t(Q)%*%y,n-p)
```

Part e)

```
beta_hat=solve(R)%*%f
beta_hat
```

```
##                       [,1]
## (Intercept)      2.4701378
## cars$speed       0.9132876
## I(cars$speed^2) 0.0999593
```

Part f)

```
all.equal(
    sum(r^2),
    sum((y-X%*%beta_hat)^2)
    )
```

```
## [1] TRUE
```

Part g)

```
sigma_hat2=sum(r^2)/(n-p)
sigma_hat2
```

```
## [1] 230.3131
```

Part h)

```
Sigma_beta_hat=solve(R)%*%t(solve(R))*sigma_hat2
Sigma_beta_hat
```

```
##                 (Intercept)  cars$speed I(cars$speed^2)
## (Intercept)     219.5483705 -28.9523122     0.872858710
## cars$speed      -28.9523122   4.1380528    -0.131439753
## I(cars$speed^2)   0.8728587  -0.1314398     0.004351805
```

Part i)

```
lm.fit<-lm(dist~speed+I(speed^2),data=cars)
beta_hat.fit<-coef(lm.fit)
```

```r
Sigma_beta_hat.fit<-vcov(lm.fit)

all.equal(
   as.numeric(beta_hat.fit),
   as.numeric(beta_hat)
   )
```

```
## [1] TRUE
```

```r
all.equal(
   as.numeric(Sigma_beta_hat.fit),
   as.numeric(Sigma_beta_hat)
   )
```

```
## [1] TRUE
```