STAT350

Assignment5

Yuhan Zhang
301345627

Q1

```
## a

```{r}
n = nrow(cement)
qt(0.95,n-2)

m0 <- lm(y ~ 1, data=cement)

summary(lm(y ~ x1, data=cement))$coef[2,3]
summary(lm(y ~ x2, data=cement))$coef[2,3]
summary(lm(y ~ x3, data=cement))$coef[2,3]
summary(lm(y ~ x4, data=cement))$coef[2,3]
summary(lm(y ~ x5, data=cement))$coef[2,3]
# the absolute value of the t-value between y and x5 is the biggest, so add x5.



m1 <- lm(y ~ x5, data=cement)

summary(lm(resid(m1) ~ x1, data=cement))$coef[2,3]
summary(lm(resid(m1) ~ x2, data=cement))$coef[2,3]
summary(lm(resid(m1) ~ x3, data=cement))$coef[2,3]
summary(lm(resid(m1) ~ x4, data=cement))$coef[2,3]
# the absolute value of the t-value between residue of m1 and x1 is the biggest, so add x1


m2 <- lm(y ~ x5 + x1, data=cement)

summary(lm(resid(m2) ~ x2, data=cement))$coef[2,3]
summary(lm(resid(m2) ~ x3, data=cement))$coef[2,3]
summary(lm(resid(m2) ~ x4, data=cement))$coef[2,3]
# the absolute values of the t-value in this step are smaller tha the cut off value, so do not use x2, x3,and x4.
```

[1] 1.795885
[1] 3.513172
[1] 4.748083
[1] -2.095893
[1] -4.817159
[1] -5.050684
[1] 7.635563
[1] 0.05109441
[1] -6.905434
[1] 0.0701137
[1] 0.5061238
[1] -1.141318
[1] -0.006174206

Therefore, we add x1 and x5 according to the result we got from above code.
The final model should include only x1 and x5. y ~ x5 +x1
```

b

```{r}
m5 <- lm(y ~ ., data=cement)
summary(m5)
qt(0.95,n-2)

e_1 = resid(lm(y ~ .-x1, data=cement))
summary(lm(e_1~cement$x1))$coef[2,3]
e_2 = resid(lm(y ~ .-x2, data=cement))
summary(lm(e_2~cement$x2))$coef[2,3]
e_3 = resid(lm(y ~ .-x3, data=cement))
summary(lm(e_3~cement$x3))$coef[2,3]
e_4 = resid(lm(y ~ .-x4, data=cement))
summary(lm(e_4~cement$x4))$coef[2,3]  #the absolute value of the t-value of x4 is the smallest, so remove x4.
e_5 = resid(lm(y ~ .-x5, data=cement))
summary(lm(e_5~cement$x5))$coef[2,3]


m4<- lm(y ~ .-x4, data=cement)
e_41 = resid(lm(y ~ .-x4-x1, data=cement))
summary(lm(e_41~cement$x1))$coef[2,3]
e_42 = resid(lm(y ~ .-x4-x2, data=cement))
summary(lm(e_42~cement$x2))$coef[2,3]
e_43 = resid(lm(y ~ .-x4-x3, data=cement))
summary(lm(e_43~cement$x3))$coef[2,3]  #the absolute value of the t-value of x4 is the smallest, so remove x3
e_45 = resid(lm(y ~ .-x4-x5, data=cement))
summary(lm(e_45~cement$x5))$coef[2,3]


m3<- lm(y ~ .-x4-x3, data=cement)
e_31 = resid(lm(y ~ .-x4-x3-x1, data=cement))
summary(lm(e_31~cement$x1))$coef[2,3]
e_32 = resid(lm(y ~ .-x4-x3-x2, data=cement))
summary(lm(e_32~cement$x2))$coef[2,3]
e_33 = resid(lm(y ~ .-x4-x3-x5, data=cement))
summary(lm(e_33~cement$x5))$coef[2,3]  #the absolute value of the t-value of x4 is the smallest, so remove x5


m2<- lm(y ~ .-x4-x3-x5, data=cement)
e_21 = resid(lm(y ~ .-x4-x3-x5-x1, data=cement))
summary(lm(e_21~cement$x1))$coef[2,3]
e_22 = resid(lm(y ~ .-x4-x3-x5-x2, data=cement))
summary(lm(e_22~cement$x2))$coef[2,3]
##the absolute value of the t-value of x1 and x2are bigger than qt, so keep x1 and x2.
```

```
[1] 1.795885
```

Therefore, we keep x1 and x2 according to the result we got from above code.
The final model should include only x1 and x2. y ~ x2 +x1

c

```{r}
install.packages("car")
library(car)

summary(lm(y ~ x4+x5, data=cement))
vif(lm(y ~ x4+x5, data=cement))
```

```
Error in install.packages : Updating loaded packages

Call:
lm(formula = y ~ x4 + x5, data = cement)

Residuals:
     Min       1Q   Median       3Q      Max 
-10.4644  -6.0598  -0.2828   5.7494  11.5305 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  111.259      5.254  21.177 1.23e-09 ***
x4             6.922      3.372   2.053   0.0672 .  
x5            -7.556      3.322  -2.274   0.0462 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.589 on 10 degrees of freedom
Multiple R-squared:  0.788,	Adjusted R-squared:  0.7457 
F-statistic: 18.59 on 2 and 10 DF,  p-value: 0.0004278

      x4       x5 
663.7627 663.7627 
```

There is a strong multicollinearity between x4 and x5, which leads to large SE of slope for x4, x5. Therefore, the t statistic between x4 and x5 would be smaller and p-value would be larger.

d

```{r}
m5 <- lm(y ~ ., data=cement)
summary(m5)
qt(0.95,n-2)

e_1 = resid(lm(y ~ .-x1, data=cement))
summary(lm(e_1~cement$x1))$coef[2,3]
e_2 = resid(lm(y ~ .-x2, data=cement))
summary(lm(e_2~cement$x2))$coef[2,3]
e_3 = resid(lm(y ~ .-x3, data=cement))
summary(lm(e_3~cement$x3))$coef[2,3]
e_4 = resid(lm(y ~ .-x4, data=cement))
summary(lm(e_4~cement$x4))$coef[2,3]  #the absolute value of the t-value of x4 is the smallest, so remove x4.
e_5 = resid(lm(y ~ .-x5, data=cement))
summary(lm(e_5~cement$x5))$coef[2,3]

m4<- lm(y ~ .-x4, data=cement)
e_41 = resid(lm(y ~ .-x4-x1, data=cement))
summary(lm(e_41~cement$x1))$coef[2,3]
e_42 = resid(lm(y ~ .-x4-x2, data=cement))
summary(lm(e_42~cement$x2))$coef[2,3]
e_43 = resid(lm(y ~ .-x4-x3, data=cement))
summary(lm(e_43~cement$x3))$coef[2,3]  #the absolute value of the t-value of x4 is the smallest, so remove x3
e_45 = resid(lm(y ~ .-x4-x5, data=cement))
summary(lm(e_45~cement$x5))$coef[2,3]

check1 <- lm(y~. -x4, data=cement)
summary(lm(resid(check1)~.,data=cement))$coef[2,3]  #not add x4

m3<- lm(y ~ .-x4-x3, data=cement)
e_31 = resid(lm(y ~ .-x4-x3-x1, data=cement))
summary(lm(e_31~cement$x1))$coef[2,3]
e_32 = resid(lm(y ~ .-x4-x3-x2, data=cement))
summary(lm(e_32~cement$x2))$coef[2,3]
e_33 = resid(lm(y ~ .-x4-x3-x5, data=cement))
summary(lm(e_33~cement$x5))$coef[2,3]  #the absolute value of the t-value of x4 is the smallest, so remove x5

check2 <- lm(y~. -x4-x3, data=cement)
summary(lm(resid(check2)~ x4,data=cement))$coef[2,3]  #not add x4
summary(lm(resid(check2)~ x3,data=cement))$coef[2,3]  #not add x3

m2<- lm(y ~ .-x4-x3-x5, data=cement)
e_21 = resid(lm(y ~ .-x4-x3-x5-x1, data=cement))
summary(lm(e_21~cement$x1))$coef[2,3]
e_22 = resid(lm(y ~ .-x4-x3-x5-x2, data=cement))
summary(lm(e_22~cement$x2))$coef[2,3]
##the absolute value of the t-value of x1 and x2are bigger than qt, so keep x1 and x2.

check3 <- lm(y ~. -x4-x3-x5,  data=cement)
summary(lm(resid(check3)~ x4,data=cement))$coef[2,3] #not add x3
summary(lm(resid(check3)~ x3,data=cement))$coef[2,3] #not add x4
summary(lm(resid(check3)~ x5,data=cement))$coef[2,3] #not add x5
```

Therefore, the final model should include only x1 and x2.y~x1, x2.

e

```{r}
AIC(lm(y ~ x5 + x1, data=cement))
AIC(lm(y ~ x1 + x2, data=cement))
AIC(lm(y ~ x1 + x2, data=cement))
```

```
[1] 67.98473
[1] 63.81786
[1] 63.81786
```

Q2

2. $\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \hline \beta_4 \\ \beta_5 \end{pmatrix} = \begin{pmatrix} \beta_a \\ \hline \beta_b \end{pmatrix}$. $\beta_a = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ 0 \\ 0 \end{pmatrix}$. $\beta_b = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \beta_4 \\ \beta_5 \end{pmatrix}$.

a) $H_0: \beta_b = \beta = 0$ $[\beta_a = 0]$.

$H_a: \beta_b = \beta \neq 0$

$SS_R(\beta_b) = SS_R(\beta) - 0$

$$F = \frac{SS_R(\beta_b)/5}{SS_{Res}(\beta)/n-1-5}$$

$$= \frac{SS_R(\beta_b)/5}{SS_{Res}(\beta)/n-6} = \frac{SS_R(\beta)/5}{SS_{Res}(\beta)\,n-6}$$

b) $H_0: \beta_b = 0$

$H_a: \beta_b \neq 0$

$SS_R(\beta_b|\beta_a) = SS_R(\beta) - SS_R(\beta_a)$

$= SS_{Res}(\beta_a) - SS_{Res}(\beta)$. $\left[\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \hline \beta_4 \\ \beta_5 \end{pmatrix} = \begin{pmatrix} \beta_a \\ \hline \beta_b \end{pmatrix}, \beta_a = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ 0 \\ 0 \end{pmatrix} \right]$

$$F = \frac{SS_R(\beta_b|\beta_a)/2}{SS_{Res}(\beta)/n-5-1} = \frac{SS_R(\beta_b|\beta_a)/2}{SS_{Res}(\beta)/n-6}$$

Further calculation:

$SS_R(\beta) = \hat{\beta}' X' y$.

$$X = \begin{array}{c} \begin{array}{cc} X_a & X_b \end{array} \\ \left[\begin{array}{ccc|cc} x_{11} & x_{21} & x_{31} & x_{41} & x_{51} \\ x_{12} & x_{22} & x_{32} & x_{42} & x_{52} \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{array}\right] \end{array}$$

$\hat{\beta}_a = (X_a' X)^{-1} X_a' y$.

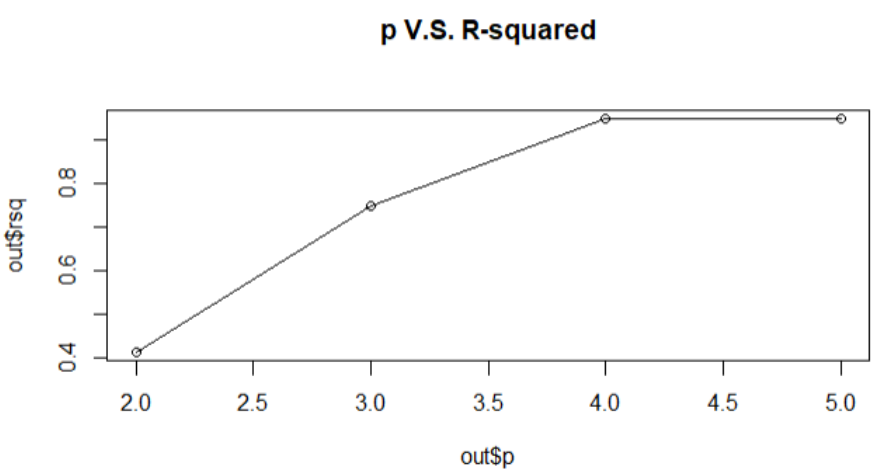
$SS_R(\beta_a) = \hat{\beta}_a X_a' y$.

$SS_R(\beta_b|\beta_a) = SS_R(\beta) - SS_R(\beta_a)$.

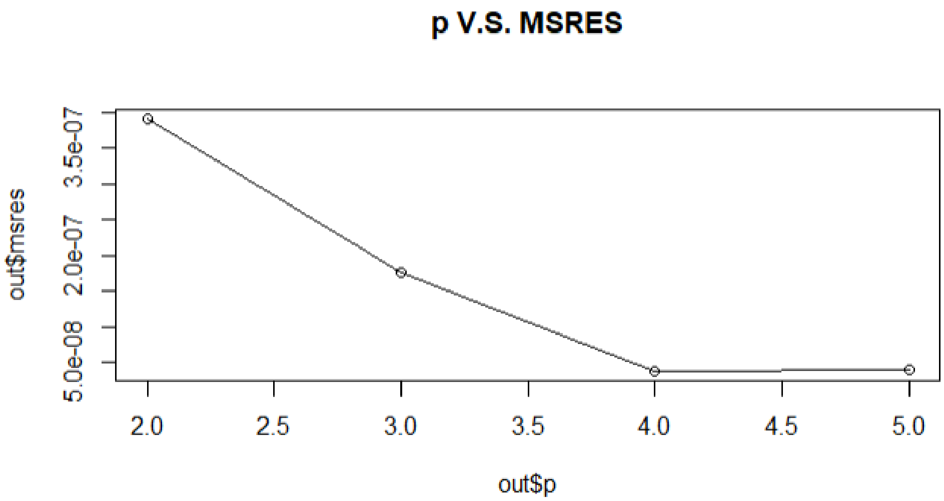
$$F_0 = \frac{SS_R(\beta_b|\beta_a)/1}{MS_{Res}}$$

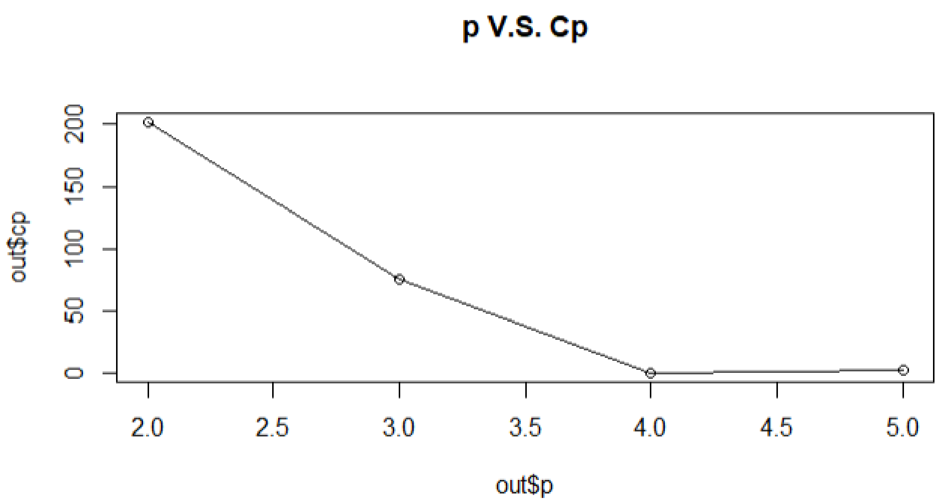
Q3

````
# Q3
```{r}
setwd("C:/Users/carol/Desktop/stat350")
reactor <- read.csv("reactor.csv")[,-1]

head(reactor)

install.packages("leaps")
library(leaps)

regsubsets.out <-
    regsubsets(y ~ .,
               data = reactor,
               nbest = 1,
               nvmax = 4)
regsubsets.out
summary.out <- summary(regsubsets.out)
as.data.frame(summary.out$outmat)


plot(regsubsets.out, scale = "Cp", main = "Cp")
names(summary.out)
summary.out$rsq

msres <-  summary.out$rss/(28-2:5)
msres

summary.out$cp
out <- data.frame(p = 2:5, rsq = summary.out$rsq, msres = msres, cp=summary.out$cp )

plot(x=out$p,y=out$rsq, type = "o", main = "p V.S. R-squared") ##a
plot(x=out$p,y=out$msres, type = "o", main = "p V.S. MSRES") ##b
plot(x=out$p,y=out$cp, type = "o", main = "p V.S. Cp")  ##c
```
````




```
a. R-square can not be used to determine which is best model as R^2 is increasing continuously.
b. form the table p V.S. MSRES,  p=4 (k=3) is the best, as when p=4, Msres=3.862218e-08, which is the smallest
c. From the table p V.S. Cp, cp p=4 is the best, as at p=4. Cp=0.556, which is the minimum value of cp.
```


p V.S. MSRES
out$msres
5.0e-08
2.0e-07
3.5e-07
2.0
2.5
3.0
3.5
4.0
4.5
5.0
out$p


p V.S. Cp
out$cp
0
50
100
150
200
2.0
2.5
3.0
3.5
4.0
4.5
5.0
out$p