Homework 4

Ayush Kumar

3/14/2021

1. Consider the model $\vec{y} = X\vec{\beta} + \vec{\epsilon}$, and suppose that for a particular input value $\vec{x_0}$, $\vec{x_0}(X'X)^{-1}X'\vec{y} = \vec{x_0'}\hat{\beta}$ is the prediction made by OLS estimation, we will show/find the following:

a. $\vec{x_0'}\hat{\beta}$ is unbiased for $\vec{x_0}\beta$: Notice that E($\vec{x_0'}\hat{\beta}$)=E($\vec{x_0}(X'X)^{-1}X'\vec{y}$)=E($\vec{x_0}(X'X)^{-1}X'(X\vec{\beta}+\vec{\epsilon})$)=E($\vec{x_0}(X'X)^{-1}X'X\vec{\beta}$) + $E(\vec{x_0}(X'X)^{-1}X'\vec{\epsilon})$=$E(\vec{x_0}I\vec{\beta}) + E(\vec{x_0}(X'X)^{-1}X'\vec{\epsilon}) = E(\vec{x_0}\vec{\beta}) = \vec{x_0}\vec{\beta}$. Thus, $\vec{x_0'}\hat{\beta}$ is unbiased for $\vec{x_0}\beta$.
b. Estimate $var(\vec{x_0'}\hat{\beta})$: Notice that $var(\vec{x_0'}\hat{\beta}) = (\vec{x_0}'\hat{\beta} - \vec{x_0}'\beta)(\vec{x_0}'\hat{\beta} - \vec{x_0}'\beta)' = (\vec{x_0}(X'X)^{-1}X'\epsilon)(\vec{x_0}(X'X)^{-1}X'\epsilon)'$ now we distribute the transposition to find $(\vec{x_0}(X'X)^{-1}X'\epsilon)(\vec{x_0}(X'X)^{-1}X'\epsilon)' = (\vec{x_0}(X'X)^{-1}X'\epsilon)(\epsilon'X(X'X)^{-1}\vec{x_0}') = \sigma^2(\vec{x_0}(X'X)^{-1}\vec{x_0}')$. Thus, we have $var(\vec{x_0'}\hat{\beta}) = \sigma^2(\vec{x_0}(X'X)^{-1}\vec{x_0}')$.
c. Proof of $\vec{x_0}'\hat{\beta}$ being BLUE:

Proof. Let $\vec{x_0}\tilde{\beta} = M\vec{y}$ be unbiased for $\vec{x_0}\beta$. Let $D = M - \vec{x_0}(X'X)^{-1}X'$, so $\vec{x_0}\tilde{\beta} = (D + \vec{x_0}(X'X)^{-1}X')\vec{y}$. As we had $E(\vec{y}) = X\beta$, we have $E(\vec{x_0}\tilde{\beta}) = (D + \vec{x_0}(X'X)^{-1}X')X\beta = DX\beta + \vec{x_0}(X'X)^{-1}X'X\beta$, now as $\vec{x_0}\tilde{\beta}$ is unbiased it must be that $DX = 0 = X'D'$, now consider $var(\vec{x_0}\tilde{\beta}) = \sigma^2(\vec{x_0}(M'M)\vec{x_0}')$, as $(M'M) = (X'X)^{-1} + DD'$ we have $var(\vec{x_0}\tilde{\beta}) = \sigma^2(\vec{x_0}((X'X)^{-1} + DD')\vec{x_0}') = var(\vec{x_0}\beta) + \sigma^2\vec{x_0}DD'\vec{x_0}'$, Each variance, then, is minimized by the corresponding ith row of D being identically 0, so the variance of $\vec{x_0}\tilde{\beta}$ is at least as big as $var(\vec{x_0}\hat{\beta})$ thus $\vec{x_0}'\hat{\beta}$ is BLUE. □

2. We consider *prestige.data* to estimate the following:

a. Here is the code and summary:

```
prest<-read.table('http://math.uttyler.edu/nathan/data/prestige.data',header=TRUE)
m1<-lm(prestige ~ income + type + income*type,data = prest)
summary(m1)
```

```
##
## Call:
## lm(formula = prestige ~ income + type + income * type, data = prest)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.267  -5.296   0.313   4.339  25.020
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     13.904517   3.167179    4.39  3.0e-05 ***
## income           0.004023   0.000553    7.28  1.1e-10 ***
## typeprof        45.019022   4.290740   10.49  < 2e-16 ***
## typewc          18.980739   5.342102    3.55   0.0006 ***
```

```
## income:typeprof -0.003178   0.000605   -5.26  9.5e-07 ***
## income:typewc   -0.002171   0.000970   -2.24   0.0276 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.27 on 92 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.829,  Adjusted R-squared:  0.819
## F-statistic: 88.9 on 5 and 92 DF,  p-value: <2e-16
```

The summary table shows all the possible combinations, there are three possible combinations first type considered is bc with intercept 13.90, with slope 0.004, next type considered is prof with intercept equaling 58.919, with slope 0.001, and finally type considered is wc with intercept equaling 32.88, and slope equaling 0.002. b. Now we consider the other three other models as follows:

```
m2 <- lm(prestige ~ income,subset=type=='bc',data=prest)
m3 <- lm(prestige ~ income,subset=type=='wc',data=prest)
m4 <- lm(prestige ~ income,subset=type=='prof',data=prest)
summary(m2)
```

```
##
## Call:
## lm(formula = prestige ~ income, data = prest, subset = type ==
##     "bc")
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.574  -4.608   0.877   2.945  21.035
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.39e+01   2.63e+00    5.30  4.0e-06 ***
## income      4.02e-03   4.58e-04    8.78  4.7e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.02 on 42 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.647,  Adjusted R-squared:  0.639
## F-statistic: 77.1 on 1 and 42 DF,  p-value: 4.71e-11
```

```
summary(m3)
```

```
##
## Call:
## lm(formula = prestige ~ income, data = prest, subset = type ==
##     "wc")
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -12.27  -6.38  -0.64   5.26  25.02
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.29e+01   5.34e+00    6.16  4.1e-06 ***
## income      1.85e-03   9.89e-04    1.87    0.075 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.02 on 21 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.143,  Adjusted R-squared:  0.102
## F-statistic: 3.51 on 1 and 21 DF,  p-value: 0.075
```

```
summary(m4)
```

```
##
## Call:
## lm(formula = prestige ~ income, data = prest, subset = type ==
##     "prof")
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.244  -5.101  -0.563   6.991  15.128
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.89e+01   2.98e+00   19.74   <2e-16 ***
## income      8.45e-04   2.52e-04    3.35   0.0023 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.49 on 29 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.279,  Adjusted R-squared:  0.254
## F-statistic: 11.2 on 1 and 29 DF,  p-value: 0.00226
```

Notice that for each of the models we have the same slope, and intercept recorded as in 2.*a* consequently we note that models generated are equivalent for the respective intercept.

3. We begin our analysis by loading *prostate.data* and then proceed with the question:

a. Here is the code:

```
prost<-read.table('http://math.uttyler.edu/nathan/data/prostate.data',header=TRUE)
t1<-lm(lpsa~lcavol,data=prost)
t2<-lm(lpsa~lweight,data=prost)
t3<-lm(lpsa~age,data=prost)
t4<-lm(lpsa~lbph,data=prost)
t5<-lm(lpsa~svi,data=prost)
t6<-lm(lpsa~lcp,data=prost)
t7<-lm(lpsa~gleason,data=prost)
t8<-lm(lpsa~pgg45,data=prost)
list_1<-rep(0,8)
```

```
list_2<-rep(0,8)
for(i in 1:8){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%
  arrange(list_1)
```

```
## # A tibble: 8 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 1     233.
## 2 model 5     271.
## 3 model 6     273.
## 4 model 8     289.
## 5 model 7     294.
## 6 model 2     295.
## 7 model 4     305.
## 8 model 3     305.
```

```
m1<-t1
```

Notice that $m1$ is simply the model $lm(lpsa \sim lcavol, data = prost)$ now we precede with question b. b. Here is the code:

```
t1<-lm(lpsa~lweight+lcavol,data=prost)
t2<-lm(lpsa~age+lcavol,data=prost)
t3<-lm(lpsa~lbph+lcavol,data=prost)
t4<-lm(lpsa~svi+lcavol,data=prost)
t5<-lm(lpsa~lcp+lcavol,data=prost)
t6<-lm(lpsa~gleason+lcavol,data=prost)
t7<-lm(lpsa~pgg45+lcavol,data=prost)
list_1<-rep(0,7)
list_2<-rep(0,7)
for(i in 1:7){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%arrange(AIC)
```

```
## # A tibble: 7 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 1     225.
## 2 model 4     226.
## 3 model 3     229.
## 4 model 7     232.
## 5 model 5     234.
## 6 model 6     234.
## 7 model 2     235.
```

```
m2<-t1
```

Notice that $m2$ is simply the model $lm(lpsa \sim lweight + lcavol, data = prost)$ now we precede with question c. c. Here is the code:

```
t1<-lm(lpsa~lweight+lcavol+age,data=prost)
t2<-lm(lpsa~lweight+lcavol+lbph,data=prost)
t3<-lm(lpsa~lweight+lcavol+svi,data=prost)
t4<-lm(lpsa~lweight+lcavol+lcp,data=prost)
t5<-lm(lpsa~lweight+lcavol+gleason,data=prost)
t6<-lm(lpsa~lweight+lcavol+pgg45,data=prost)
list_1<-rep(0,6)
list_2<-rep(0,6)
for(i in 1:6){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%arrange(AIC)
```

```
## # A tibble: 6 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 3     217.
## 2 model 6     223.
## 3 model 4     225.
## 4 model 5     225.
## 5 model 2     225.
## 6 model 1     226.
```

```
m3<-t3

t1<-lm(lpsa~lweight+lcavol+svi+age,data=prost)
t2<-lm(lpsa~lweight+lcavol+svi+lbph,data=prost)
t3<-lm(lpsa~lweight+lcavol+svi+lcp,data=prost)
t4<-lm(lpsa~lweight+lcavol+svi+gleason,data=prost)
t5<-lm(lpsa~lweight+lcavol+svi+pgg45,data=prost)
list_1<-rep(0,5)
list_2<-rep(0,5)
for(i in 1:5){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%arrange(AIC)
```

```
## # A tibble: 5 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 2     216.
## 2 model 5     217.
## 3 model 1     218.
## 4 model 4     218.
## 5 model 3     218.
```

```
m4<-t2
#The AIC for m4 is 216.9

t1<-lm(lpsa~lweight+lcavol+svi+lbph+age,data=prost)
t2<-lm(lpsa~lweight+lcavol+svi+lbph+lcp,data=prost)
t3<-lm(lpsa~lweight+lcavol+svi+lbph+gleason,data=prost)
t4<-lm(lpsa~lweight+lcavol+svi+lbph+pgg45,data=prost)
list_1<-rep(0,4)
list_2<-rep(0,4)
for(i in 1:4){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%arrange(AIC)
```

```
## # A tibble: 4 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 1     216.
## 2 model 4     217.
## 3 model 3     217.
## 4 model 2     218.
```

```
m5<-t1
#The AIC for m5 is 215.9

t1<-lm(lpsa~lweight+lcavol+svi+lbph+age+lcp,data=prost)
t2<-lm(lpsa~lweight+lcavol+svi+lbph+age+gleason,data=prost)
t3<-lm(lpsa~lweight+lcavol+svi+lbph+age+pgg45,data=prost)
list_1<-rep(0,4)
list_2<-rep(0,4)
for(i in 1:4){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%arrange(AIC)
```

```
## # A tibble: 4 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 3     216.
## 2 model 2     217.
## 3 model 4     217.
## 4 model 1     218.
```

```
#We note model 3 has the lowest AIC 216.5 but as it is higher
#than m5 we conclude m4 was the best model we have using this criterion.
```

Notice that $m5, lm(lpsa \sim lweight + lcavol + svi + lbph + age, data = prost)$ was the model with lowest AIC, using forward selection.

d. Here is the code:

```
mod.1<-lm(lpsa~lweight+lcavol+svi+lbph+age+lcp+gleason+pgg45,data=prost)
```

the AIC for *mod*.1 is 218.9522.

e. Here is the code:

```
t1<-lm(lpsa~lweight+lcavol+svi+lbph+age+lcp+gleason,data=prost)
t2<-lm(lpsa~lweight+lcavol+svi+lbph+age+lcp+pgg45,data=prost)
t3<-lm(lpsa~lweight+lcavol+svi+lbph+age+gleason+pgg45,data=prost)
t4<-lm(lpsa~lweight+lcavol+svi+lbph+lcp+gleason+pgg45,data=prost)
t5<-lm(lpsa~lweight+lcavol+svi+age+lcp+gleason+pgg45,data=prost)
t6<-lm(lpsa~lweight+lcavol+lbph+age+lcp+gleason+pgg45,data=prost)
t7<-lm(lpsa~lweight+svi+lbph+age+lcp+gleason+pgg45,data=prost)
t8<-lm(lpsa~lcavol+svi+lbph+age+lcp+gleason+pgg45,data=prost)
list_1<-rep(0,8)
list_2<-rep(0,8)
for(i in 1:8){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%
  arrange(list_1)
```

```
## # A tibble: 8 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 2     217.
## 2 model 1     218.
## 3 model 3     218.
## 4 model 4     220.
## 5 model 5     221.
## 6 model 8     225.
## 7 model 6     227.
## 8 model 7     257.
```

```
mod.2<-t2
```

Thus, we have the lowest AIC for the model $lm(lpsa \sim lweight + lcavol + svi + lbph + age + lcp + pgg45, data = prost)$.

f. Here is the code:

```
t1<-lm(lpsa~lweight+lcavol+svi+lbph+age+lcp,data=prost)
t2<-lm(lpsa~lweight+lcavol+svi+lbph+age+pgg45,data=prost)
t3<-lm(lpsa~lweight+lcavol+svi+lbph+lcp+pgg45,data=prost)
t4<-lm(lpsa~lweight+lcavol+svi+age+lcp+pgg45,data=prost)
t5<-lm(lpsa~lweight+lcavol+lbph+age+lcp+pgg45,data=prost)
t6<-lm(lpsa~lweight+svi+lbph+age+lcp+pgg45,data=prost)
t7<-lm(lpsa~lcavol+svi+lbph+age+lcp+pgg45,data=prost)
list_1<-rep(0,7)
list_2<-rep(0,7)
```

```
for(i in 1:7){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%
  arrange(list_1)
```

```
## # A tibble: 7 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 2     216.
## 2 model 1     218.
## 3 model 3     218.
## 4 model 4     219.
## 5 model 7     223.
## 6 model 5     225.
## 7 model 6     256.
```

```
mod.3<-t2
#model.3 AIC=216.5

t1<-lm(lpsa~lweight+lcavol+svi+lbph+age,data=prost)
t2<-lm(lpsa~lweight+lcavol+svi+lbph+pgg45,data=prost)
t3<-lm(lpsa~lweight+lcavol+svi+age+pgg45,data=prost)
t4<-lm(lpsa~lweight+lcavol+lbph+age+pgg45,data=prost)
t5<-lm(lpsa~lweight+svi+lbph+age+pgg45,data=prost)
t6<-lm(lpsa~lcavol+svi+lbph+age+pgg45,data=prost)
list_1<-rep(0,6)
list_2<-rep(0,6)
for(i in 1:6){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%
  arrange(list_1)
```

```
## # A tibble: 6 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 1     216.
## 2 model 2     217.
## 3 model 3     218.
## 4 model 6     222.
## 5 model 4     223.
## 6 model 5     258.
```

```
mod.4<-t1

t1<-lm(lpsa~lweight+lcavol+svi+lbph,data=prost)
t2<-lm(lpsa~lweight+lcavol+svi+age,data=prost)
t3<-lm(lpsa~lweight+lcavol+lbph+age,data=prost)
t4<-lm(lpsa~lweight+svi+lbph+age,data=prost)
```

```
t5<-lm(lpsa~lcavol+svi+lbph+age,data=prost)
list_1<-rep(0,5)
list_2<-rep(0,5)
for(i in 1:5){
  list_1[i]<-AIC(get(paste0("t",i)))
  list_2[i]<-paste("model",i)
}
tibble(Model_Name=list_2,AIC=list_1)%>%
  arrange(list_1)
```

```
## # A tibble: 5 x 2
##   Model_Name   AIC
##   <chr>      <dbl>
## 1 model 1     216.
## 2 model 2     218.
## 3 model 5     221.
## 4 model 3     226.
## 5 model 4     261.
```

```
#The model did not improve
#Thus, the lowest AIC was 215.9 which is higher than that of mod.3 so we
#conclude that mod.4 is the best model according to this criterion
```

The best model according to this criteria was $lm(lpsa \sim lweight + lcavol + svi + lbph + age, data = prost)$

g. Both methods arrived at the same conclusion, i.e. the model with lowest AIC by both methods was $lm(lpsa \sim lweight + lcavol + svi + lbph + age, data = prost)$ with $AIC = 215.9$.