HW4

Rickey Huang

3/2/2022

Question1

Part a

```r
library(faraway)
data(chredlin)
zips <- as.integer(rownames(chredlin))
```

Part b

```r
# Fit two models
lm0 <- lm(zips~theft + fire, data = chredlin)
lmo <- lm(zips~fire, data = chredlin)
```

Part c

```r
# Show the result of the previous step
summary(lm0)
```

```
##
## Call:
## lm(formula = zips ~ theft + fire, data = chredlin)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.464  -9.557  -3.562   9.927  26.732
##
## Coefficients:
##               Estimate Std. Error   t value Pr(>|t|)
## (Intercept) 60641.7280     3.5714 16980.029   <2e-16 ***
## theft          -0.1590     0.1035    -1.536   0.1318
## fire           -0.4875     0.2481    -1.965   0.0558 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.01 on 44 degrees of freedom
## Multiple R-squared:  0.2396, Adjusted R-squared:  0.2051
## F-statistic: 6.933 on 2 and 44 DF,  p-value: 0.002414
```

```r
summary(lmo)
```

```
##
```

```
## Call:
## lm(formula = zips ~ fire, data = chredlin)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.490  -9.103  -2.646  10.023  28.924
##
## Coefficients:
##               Estimate Std. Error   t value Pr(>|t|)
## (Intercept) 60639.1843     3.2115 18882.087  < 2e-16 ***
## fire           -0.6995     0.2093    -3.342  0.00168 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.2 on 45 degrees of freedom
## Multiple R-squared:  0.1989, Adjusted R-squared:  0.1811
## F-statistic: 11.17 on 1 and 45 DF,  p-value: 0.001681
```

Part d

```r
# Find the degree of freedom
n <- length(zips)
p <- 3
df <- n-p
#  Find the critical t-score
t <- qt(0.975,df)
t
```

```
## [1] 2.015368
```

```r
# Find the standad errors using matrices
X <- model.matrix(lm0)
sigmasq <- t(lm0$fitted-zips)%*%(lm0$fitted-zips)/df
sigmasq <- sigmasq[1]
varbetahat <- sigmasq*solve(t(X)%*%X)
sebetahatcoeff <- (varbetahat^(0.5))[1,1]
sebetahattheft <- (varbetahat^(0.5))[2,2]
sebetahatfire <- (varbetahat^(0.5))[3,3]
# Construct the 95% confidence intervals
cicoeff <- c(-1,1)*t*sebetahatcoeff + lm0$coeff[1]
citheft <- c(-1,1)*t*sebetahattheft + lm0$coeff[2]
cifire <- c(-1,1)*t*sebetahatfire + lm0$coeff[3]
cicoeff
```

```
## [1] 60634.53 60648.93
```

```r
citheft
```

```
## [1] -0.36769804  0.04966002
```

```r
cifire
```

```
## [1] -0.98757378  0.01253986
```

```r
# Verify my results using the confint command in R
confint(lm0,1,level=0.95)
```

```
##                2.5 %   97.5 %
```

```
## (Intercept) 60634.53 60648.93
```

```r
confint(lm0,2,level=0.95)
```

```
##            2.5 %     97.5 %
## theft -0.367698 0.04966002
```

```r
confint(lm0,3,level=0.95)
```

```
##            2.5 %     97.5 %
## fire -0.9875738 0.01253986
```

Part e

```r
# Find the t-scores for each parameter
tcoeff <- (lm0$coeff[1] - 0)/sebetahatcoeff
ttheft <- (lm0$coeff[2] - 0)/sebetahattheft
tfire <- (lm0$coeff[3] - 0)/sebetahatfire
tcoeff
```

```
## (Intercept)
##    16980.03
```

```r
ttheft
```

```
##     theft
## -1.535764
```

```r
tfire
```

```
##      fire
## -1.964828
```

```r
# Find the p-values of each parameter
pvalcoeff <- 2*pt(tcoeff,df)
pvaltheft <- 2*pt(ttheft,df)
pvalfire <- 2*pt(tfire,df)
pvalcoeff
```

```
## (Intercept)
##           2
```

```r
pvaltheft
```

```
##     theft
## 0.1317566
```

```r
pvalfire
```

```
##       fire
## 0.05576805
```

```r
# Since the pvalcoeff is bigger than 0, we want to use 2(1-pt(t,df)) to compute its p-value
pvalcoeff <- 2*(1 - pt(tcoeff,df))
pvalcoeff
```

```
## (Intercept)
##           0
```

```r
# Using summary command in R to test my results
summary(lm0)
```

```
##
## Call:
## lm(formula = zips ~ theft + fire, data = chredlin)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.464  -9.557  -3.562   9.927  26.732
##
## Coefficients:
##              Estimate Std. Error   t value Pr(>|t|)
## (Intercept) 60641.7280     3.5714 16980.029   <2e-16 ***
## theft          -0.1590     0.1035    -1.536   0.1318
## fire           -0.4875     0.2481    -1.965   0.0558 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.01 on 44 degrees of freedom
## Multiple R-squared:  0.2396, Adjusted R-squared:  0.2051
## F-statistic: 6.933 on 2 and 44 DF,  p-value: 0.002414
```

Part g

```r
# Find SSE for both models
SSEO <- (t(lmO$residuals)%*%lmO$residuals)[1]
SSEo <- (t(lmo$residuals)%*%lmo$residuals)[1]
q <- dim(model.matrix(lmo))[2]
# Find the F-statistic
Fstat <- ((SSEo-SSEO)/(p-q))/(SSEO/(n-p))
Fstat
```

```
## [1] 2.358571
```

```r
# Find the p-value
pvalF <- 1-pf(Fstat,p-q,n-p)
pvalF
```

```
## [1] 0.1317566
```

```r
# Find the Fstar
Fstar <- qf(0.95,p-q,n-p)
Fstar
```

```
## [1] 4.061706
```

Part h

```r
# Fit the trival model omega0
lmo0 <- lm(zips~1)
# Find the SSE for this model
SSEo0 <- (t(lmo0$residuals)%*%lmo0$residuals)[1]
q0 <- dim(model.matrix(lmo0))[2]
# Find the F-statistic
Fstat0 <- ((SSEo0-SSEO)/(p-q0))/(SSEO/(n-p))
Fstat0
```

```
## [1] 6.932705
```

```
# Find the p-value
pvalF0 <- 1-pf(Fstat0,p-q0,n-p)
pvalF0
```

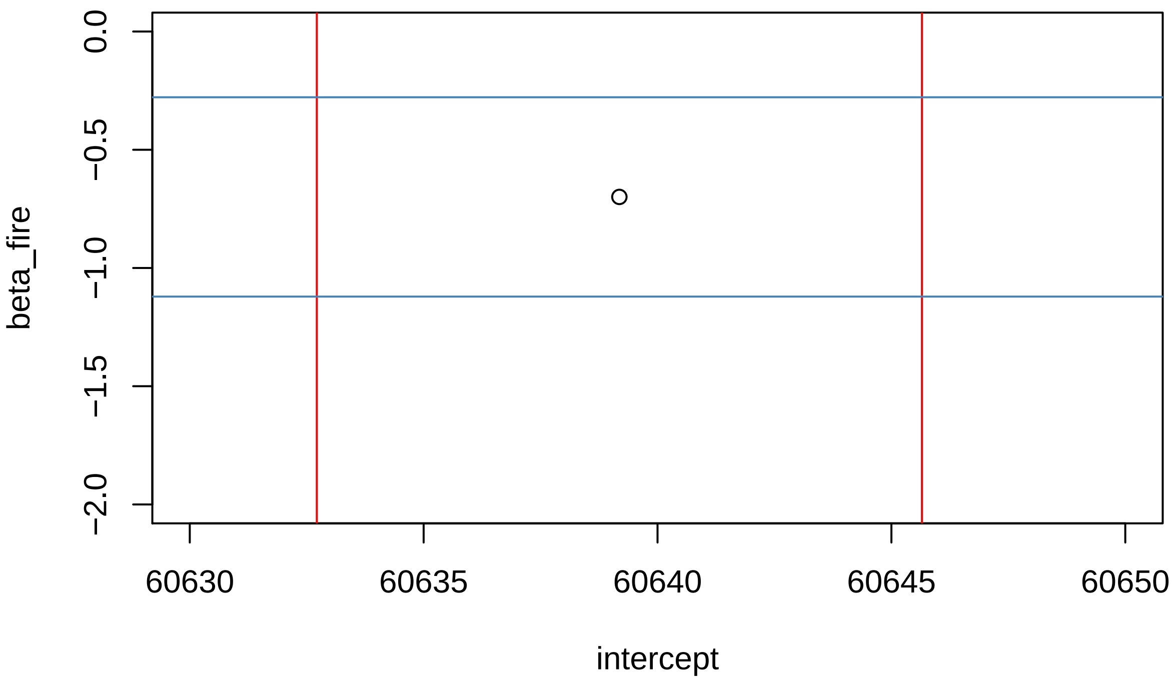
```
## [1] 0.002414016
```

```
# Find the Fstar
Fstar0 <- qf(0.95,p-q0,n-p)
Fstar0
```

```
## [1] 3.209278
```

j

```
# First plot the estimate
plot(lmo$coeff[1],lmo$coeff[2], xlab = "intercept", ylab = "beta_fire",xlim=c(60630,60650), ylim=c(-2,0
# Construct the rectangle
vlow <- confint(lmo)[1]
vhigh <- confint(lmo)[3]
hlow <- confint(lmo)[2]
hhigh <- confint(lmo)[4]
# Draw the lines
abline(v = vhigh,col = 'red')
abline(v = vlow,col = 'red')
abline(hlow,0, col = 'steelblue')
abline(hhigh,0, col = 'steelblue')
```



Part k

```
# Install the package "ellipse"
# (Uncomment the next line if it is not installed before)
#install.packages("ellipse")
```

```r
library(ellipse)
```

```
##
## Attaching package: 'ellipse'

## The following object is masked from 'package:graphics':
##
##     pairs
```

```r
ellipse(lmo)
```

```
##        (Intercept)       fire
##   [1,]    60641.75 -0.5320027
##   [2,]    60641.26 -0.5004592
##   [3,]    60640.76 -0.4697170
##   [4,]    60640.25 -0.4398999
##   [5,]    60639.73 -0.4111280
##   [6,]    60639.22 -0.3835172
##   [7,]    60638.70 -0.3571785
##   [8,]    60638.19 -0.3322181
##   [9,]    60637.68 -0.3087365
##  [10,]    60637.18 -0.2868282
##  [11,]    60636.68 -0.2665815
##  [12,]    60636.19 -0.2480778
##  [13,]    60635.72 -0.2313917
##  [14,]    60635.26 -0.2165903
##  [15,]    60634.82 -0.2037334
##  [16,]    60634.39 -0.1928725
##  [17,]    60633.99 -0.1840516
##  [18,]    60633.60 -0.1773060
##  [19,]    60633.24 -0.1726629
##  [20,]    60632.90 -0.1701411
##  [21,]    60632.58 -0.1697508
##  [22,]    60632.29 -0.1714933
##  [23,]    60632.03 -0.1753619
##  [24,]    60631.80 -0.1813408
##  [25,]    60631.60 -0.1894060
##  [26,]    60631.43 -0.1995250
##  [27,]    60631.29 -0.2116571
##  [28,]    60631.18 -0.2257535
##  [29,]    60631.11 -0.2417573
##  [30,]    60631.07 -0.2596041
##  [31,]    60631.05 -0.2792222
##  [32,]    60631.08 -0.3005323
##  [33,]    60631.13 -0.3234489
##  [34,]    60631.22 -0.3478795
##  [35,]    60631.34 -0.3737258
##  [36,]    60631.49 -0.4008838
##  [37,]    60631.67 -0.4292440
##  [38,]    60631.88 -0.4586923
##  [39,]    60632.13 -0.4891101
##  [40,]    60632.40 -0.5203750
##  [41,]    60632.69 -0.5523610
##  [42,]    60633.02 -0.5849393
##  [43,]    60633.37 -0.6179788
```

```
## [44,] 60633.74 -0.6513464
## [45,] 60634.13 -0.6849077
## [46,] 60634.55 -0.7185277
## [47,] 60634.98 -0.7520709
## [48,] 60635.43 -0.7854022
## [49,] 60635.89 -0.8183876
## [50,] 60636.37 -0.8508940
## [51,] 60636.86 -0.8827907
## [52,] 60637.36 -0.9139493
## [53,] 60637.87 -0.9442441
## [54,] 60638.38 -0.9735534
## [55,] 60638.89 -1.0017590
## [56,] 60639.41 -1.0287473
## [57,] 60639.92 -1.0544097
## [58,] 60640.43 -1.0786429
## [59,] 60640.94 -1.1013493
## [60,] 60641.44 -1.1224375
## [61,] 60641.93 -1.1418224
## [62,] 60642.41 -1.1594262
## [63,] 60642.88 -1.1751778
## [64,] 60643.33 -1.1890140
## [65,] 60643.77 -1.2008788
## [66,] 60644.18 -1.2107247
## [67,] 60644.58 -1.2185119
## [68,] 60644.95 -1.2242091
## [69,] 60645.31 -1.2277933
## [70,] 60645.63 -1.2292501
## [71,] 60645.93 -1.2285737
## [72,] 60646.21 -1.2257667
## [73,] 60646.45 -1.2208405
## [74,] 60646.67 -1.2138149
## [75,] 60646.86 -1.2047182
## [76,] 60647.01 -1.1935870
## [77,] 60647.13 -1.1804662
## [78,] 60647.23 -1.1654085
## [79,] 60647.29 -1.1484747
## [80,] 60647.31 -1.1297328
## [81,] 60647.31 -1.1092584
## [82,] 60647.27 -1.0871339
## [83,] 60647.20 -1.0634484
## [84,] 60647.09 -1.0382973
## [85,] 60646.96 -1.0117818
## [86,] 60646.79 -0.9840087
## [87,] 60646.59 -0.9550899
## [88,] 60646.37 -0.9251417
## [89,] 60646.11 -0.8942849
## [90,] 60645.83 -0.8626435
## [91,] 60645.52 -0.8303451
## [92,] 60645.18 -0.7975197
## [93,] 60644.82 -0.7642994
## [94,] 60644.44 -0.7308181
## [95,] 60644.03 -0.6972105
## [96,] 60643.61 -0.6636120
## [97,] 60643.17 -0.6301579
```

```
## [98,]    60642.71 -0.5969829
## [99,]    60642.24 -0.5642205
## [100,]   60641.75 -0.5320027
```

```r
plot(ellipse(lmo),type = "l",xlim=c(60630,60650), ylim=c(-2,0))
points(lmo$coeff[1],lmo$coeff[2])
vlow <- confint(lmo)[1]
vhigh <- confint(lmo)[3]
hlow <- confint(lmo)[2]
hhigh <- confint(lmo)[4]
# Draw the lines
abline(v = vhigh,col = 'red')
abline(v = vlow,col = 'red')
abline(hlow,0, col = 'steelblue')
abline(hhigh,0, col = 'steelblue')
```