lab 3

Mason Wong

16th March 2022

Question 1

(a)

```
olympic = read.table("~/Desktop/R-programming/3002-labs/lab-2/olympic.txt", sep = "\t", header = TRUE)
head(olympic, 6)
```

```
##   HighJump DiscusThrow LongJump Year
## 1    71.25     1147.50  249.750   -4
## 2    74.80     1418.90  282.875    0
## 3    71.00     1546.50  289.000    4
## 4    75.00     1610.00  294.500    8
## 5    76.00     1780.00  299.250   12
## 6    76.25     1759.25  281.500   20
```

```
summary(olympic)
```

```
##     HighJump       DiscusThrow       LongJump          Year
##  Min.   :71.00   Min.   :1148   Min.   :249.8   Min.   :-4.00
##  1st Qu.:76.19   1st Qu.:1775   1st Qu.:296.2   1st Qu.:22.00
##  Median :78.97   Median :2033   Median :308.2   Median :52.00
##  Mean   :80.92   Mean   :2053   Mean   :311.3   Mean   :47.65
##  3rd Qu.:86.25   3rd Qu.:2435   3rd Qu.:331.6   3rd Qu.:74.00
##  Max.   :92.75   Max.   :2657   Max.   :350.5   Max.   :96.00
##  NA's   :3       NA's   :3
```

```
tail(olympic, 6)
```

```
##    HighJump DiscusThrow LongJump Year
## 18    88.50      2657.4   328.50   76
## 19    92.75      2624.0   336.25   80
## 20    92.50      2622.0   336.25   84
## 21       NA          NA   343.25   88
## 22       NA          NA   341.50   92
## 23       NA          NA   334.75   96
```

```
convert_inch_to_m = function(x) {
  return(x/39.3701)
}
olympicMetric = olympic %>%
  mutate(across(c(HighJump, DiscusThrow, LongJump), convert_inch_to_m)) %>%
  mutate(Year = Year + 1900)
```

(b)

```
olympicLm = lm(HighJump ~ LongJump, data = olympicMetric)
SSE_H1 = sum((olympicLm$residuals)^2)
df_H1 = olympicLm$df.residual
summary(olympicLm)
```

```
##
## Call:
## lm(formula = HighJump ~ LongJump, data = olympicMetric)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.13574 -0.07615  0.01865  0.05390  0.12339
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.07790    0.25016   0.311    0.759
## LongJump     0.25355    0.03199   7.925  2.8e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08177 on 18 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.7773, Adjusted R-squared:  0.7649
## F-statistic: 62.81 on 1 and 18 DF,  p-value: 2.8e-07
```

(c)

- Under H_1 we don't have constraint on β_0 so we have the full model of $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ for $i = 1, 2, 3..., n$ where y_i is the HighJump variable and x_i is the LongJump variable.

- Under H_0 we do have the constrain that $\beta = 0$ so we have the restricted model is $y_i = \beta_1 x_i + \varepsilon_i$ for $i = 1, 2, 3..., n$

(d)

```
olympic_H0 = lm(HighJump ~ -1 + LongJump, data = olympicMetric)
SSE_H0 = sum((olympic_H0$residuals)^2)
df_H0 = olympic_H0$df.residual
summary(olympic_H0)
```

```
##
## Call:
## lm(formula = HighJump ~ -1 + LongJump, data = olympicMetric)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.13078 -0.07432  0.01922  0.05185  0.13826
##
## Coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## LongJump 0.263490   0.002282   115.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07981 on 19 degrees of freedom
##   (3 observations deleted due to missingness)
```

```
## Multiple R-squared:  0.9986, Adjusted R-squared:  0.9985
## F-statistic: 1.333e+04 on 1 and 19 DF,  p-value: < 2.2e-16
```

(e)

The observed f statistic is 0.0969809. The corresponding p value is 0.7590611 which is the same as if we did the command `anova(olympic_H0, olympicLm)$'Pr(>F)'[2]`

(f) We have that:

- we are testing the intercept here:

$$H_0 : \beta_0 = 0 \qquad vs \qquad H_1 : \beta_0 \neq 0$$

- The manually computed f statistic for this test was:

$$0.0969809$$

- The t-test test statistic, testing the same test for the intercept β_0 was:

$$0.311$$

- We note that $(0.311)^2 = 0.0969809$