Homework 6

Nathaniel Haulk

11/8/2021

Question 1

Part a

```r
## Data from the website
x = c(110.5, 105.4, 118.1, 104.5, 93.6, 84.1, 77.8, 75.6)
y = c(5.755, 5.939, 6.010, 6.545, 6.730, 6.750, 6.899, 7.862)

## binds x and y into a dataframe
d = data.frame(c(x,y))

## Creates variable fit_d that shows the slope and the y-intercept
fit_d = lm(y~x, data = d)

print(fit_d)
```

```
##
## Call:
## lm(formula = y ~ x, data = d)
##
## Coefficients:
## (Intercept)            x
##    10.13746     -0.03717
```

The least squares estimate is of β_1 is -.037. This value represent the best fit of the trend of the data that reduces the distance from all of the points to the line itself

Part B

```r
## Runs an ANOVA test
anovad = anova(fit_d)

print(anovad)
```

```
## Analysis of Variance Table
##
```

```
## Response: y
##           Df  Sum Sq Mean Sq F value   Pr(>F)
## x          1 2.42357 2.42357  18.455 0.005116 **
## Residuals  6 0.78794 0.13132
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
## used to get the t-test p-value
summaryd = summary(fit_d)

print(summaryd)
```

```
##
## Call:
## lm(formula = y ~ x, data = d)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34626 -0.27605 -0.09448  0.27023  0.53495
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.137455   0.842265  12.036    2e-05 ***
## x           -0.037175   0.008653  -4.296  0.00512 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3624 on 6 degrees of freedom
## Multiple R-squared:  0.7547, Adjusted R-squared:  0.7138
## F-statistic: 18.46 on 1 and 6 DF,  p-value: 0.005116
```

Both the F-test and the T-test show a P-value $<.01$ confirming the alternative hypothesis. This means that $H_a \neq 0$ is true.

Part C

```r
## Calculates the t_(n-2,a/2) value
print(qt(.05/2, 8-2))
```

```
## [1] -2.446912
```

```r
## Calculate the confidence intervals
confint(fit_d)
```

```
##                   2.5 %      97.5 %
## (Intercept)  8.07650745 12.19840320
## x           -0.05834895 -0.01600043
```

Part D

```r
## Calculates raw residuals
residd= resid(fit_d)

print(residd)
```

```
##          1          2          3          4          5          6          7 
## -0.2746519 -0.2802428  0.2628757  0.2922999  0.0720958 -0.2610638 -0.3462643 
##          8 
##  0.5349514
```

Based off the values from part a, we know the y-intercept and the slope of the regression line. This give us the equation:

$$\hat{y} = 10.137 - 0.0372x$$

Part E

```r
##Print the error values
print(summaryd[6])
```

```
## $sigma
## [1] 0.3623848
```

Based off the summary statistics generated in part b, we know that $\hat{\sigma}^2 = .3624$

Part F

```r
## Predicts the 95% confidence interval based on the new rice of x = 100
mu0conf = predict(fit_d, newdata = data.frame(x = 100), interval = "confidence")

print(mu0conf)
```

```
##        fit      lwr      upr
## 1 6.419986 6.096321 6.743651
```

Part G

```r
## Predicts the 95% prediction interval based on the new rice of x = 100
mu0pred = predict(fit_d, newdata = data.frame(x = 100), interval = "prediction")

print(mu0pred)
```

```
##        fit      lwr      upr
## 1 6.419986 5.476038 7.363934
```

Compared to the confidence interval (range = 0.647), the prediction interval (range = 1.8879) is much wider

Part H

```r
## Pulls the r^2 value from summaryd
d_r_squared = summaryd['r.squared']
```

The R-squared value represents how much variance that can be explained by the independent variable, in this case plant height.

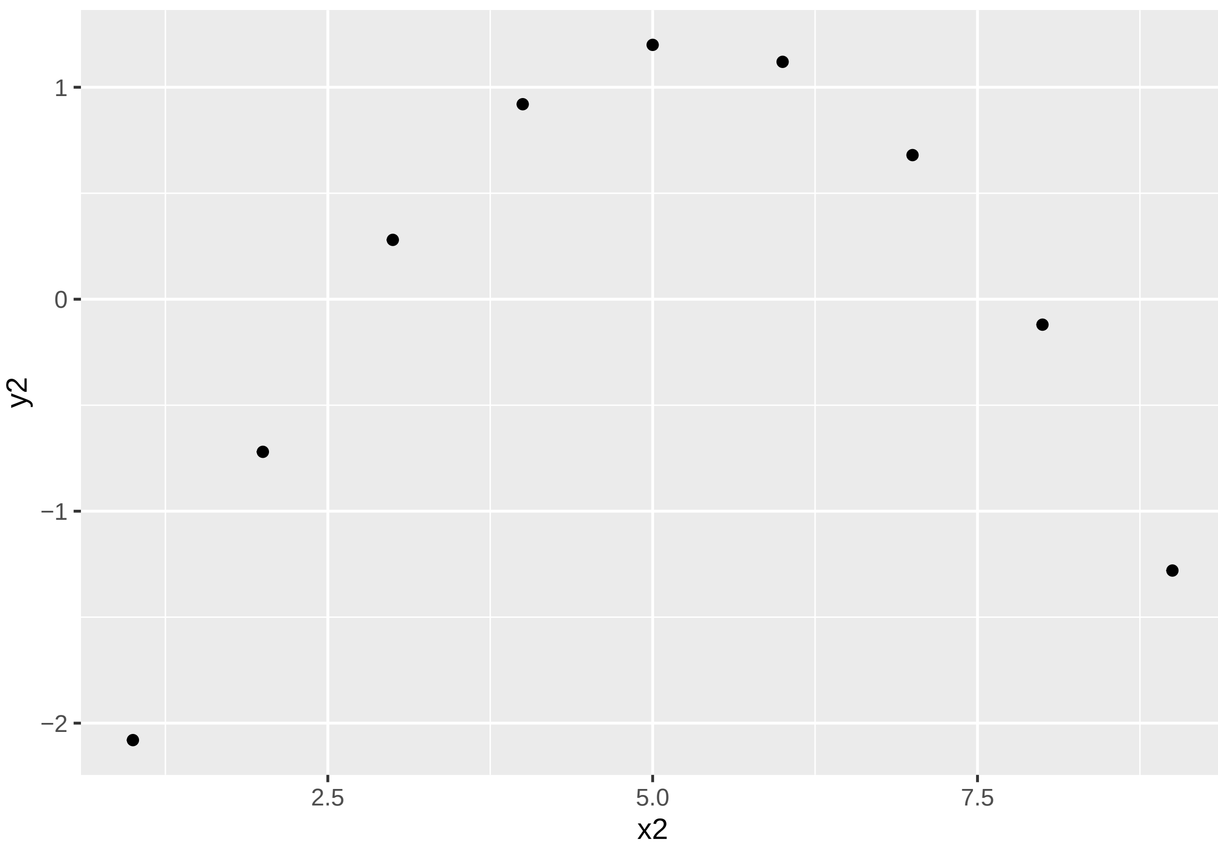
Question 2

Part A

```r
library(tidyverse, warn.conflicts = FALSE)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
## Data for question 2
x2 = c(1, 2, 3, 4, 5, 6, 7, 8, 9)
y2 = c(-2.08, -0.72, 0.28, 0.92, 1.20, 1.12, 0.68, -0.12, -1.28)

# Combines data into datafram
d2 = data.frame(x2,y2)

## Plots data
ggplot(d2, aes(x = x2, y = y2)) +
  geom_point()
```



Part B