Linear Models Assignment 7

Joshua Ingram

Problem 1.

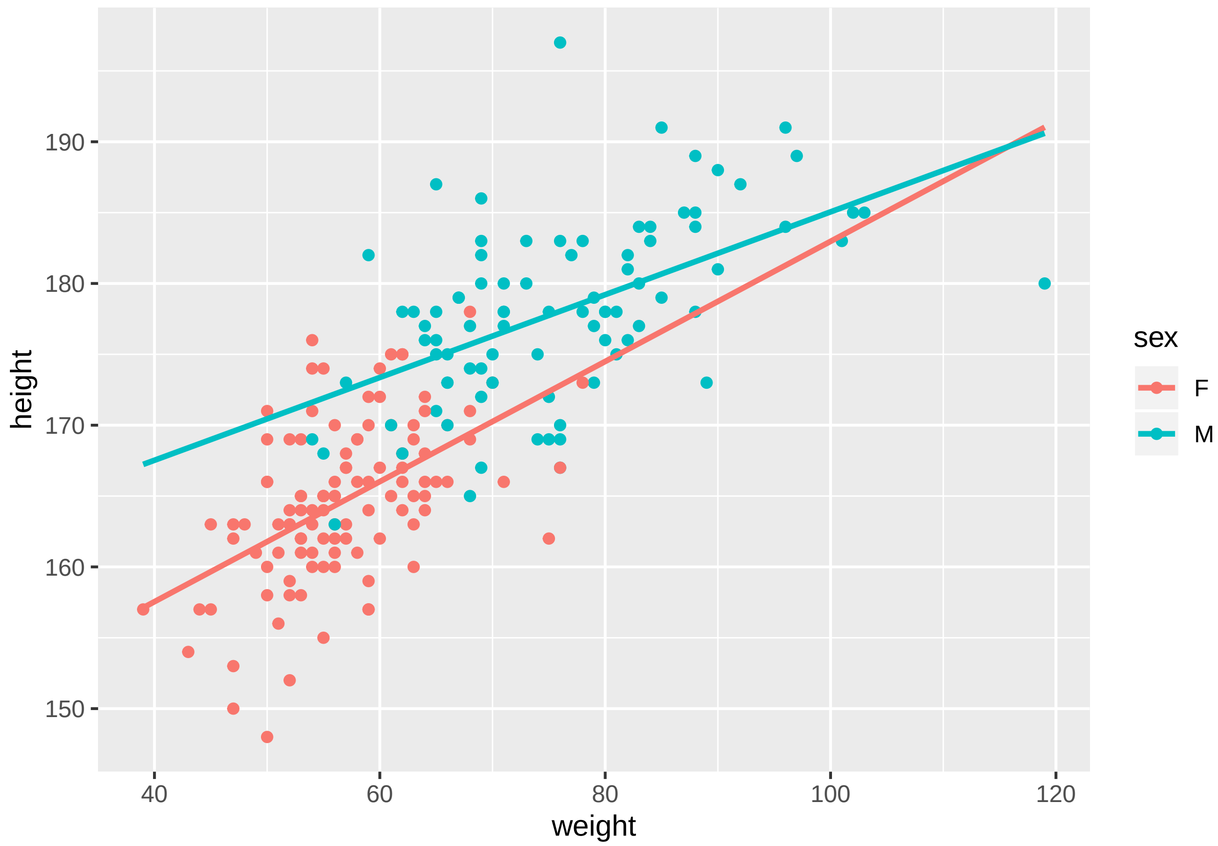
1.

Test for Association

```r
lm_davis1 <- lm(weight ~ sex, data = davis)
summary(lm_davis1)
```

```
##
## Call:
## lm(formula = weight ~ sex, data = davis)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.898  -6.395  -0.892   5.108  43.102
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  56.8919     0.8952   63.55   <2e-16 ***
## sexM         19.0058     1.3461   14.12   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.431 on 197 degrees of freedom
## Multiple R-squared:  0.5029, Adjusted R-squared:  0.5004
## F-statistic: 199.3 on 1 and 197 DF,  p-value: < 2.2e-16
```

Test for Interaction




```
lm_davis2 <- lm(height ~ weight + sex + weight:sex, data = davis)
summary(lm_davis2)$coefficients
```

```
##                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 140.5913631 4.08463197 34.419591 8.673770e-85
## weight        0.4238098 0.07128015  5.945691 1.250947e-08
## sexM         15.2487377 5.42339531  2.811659 5.432264e-03
## weight:sexM  -0.1316895 0.08507779 -1.547872 1.232745e-01
```

Comments

The association between weight and sex is statistically signficant according to our F-statistic and p-value for the overall model significance. The interaction between weight and sex in predicting height seems possible visiually, but to further study if there is significant interaction we can look at the interaction term's t-statistic and corresponding p-value. After doing so, we recieve an t-statistic of -1.548 and p-value of .1233, showing that the interaction is not statistically significant. These two differences show that association between two variables does not infer interaction between the same two in predicting a response variable.

2.

I couldn't find a dataset where there was no significant association but a significant interaction between two independent variables (after looking for several hours), so instead I attempted to make a theoretical example. My apologies.

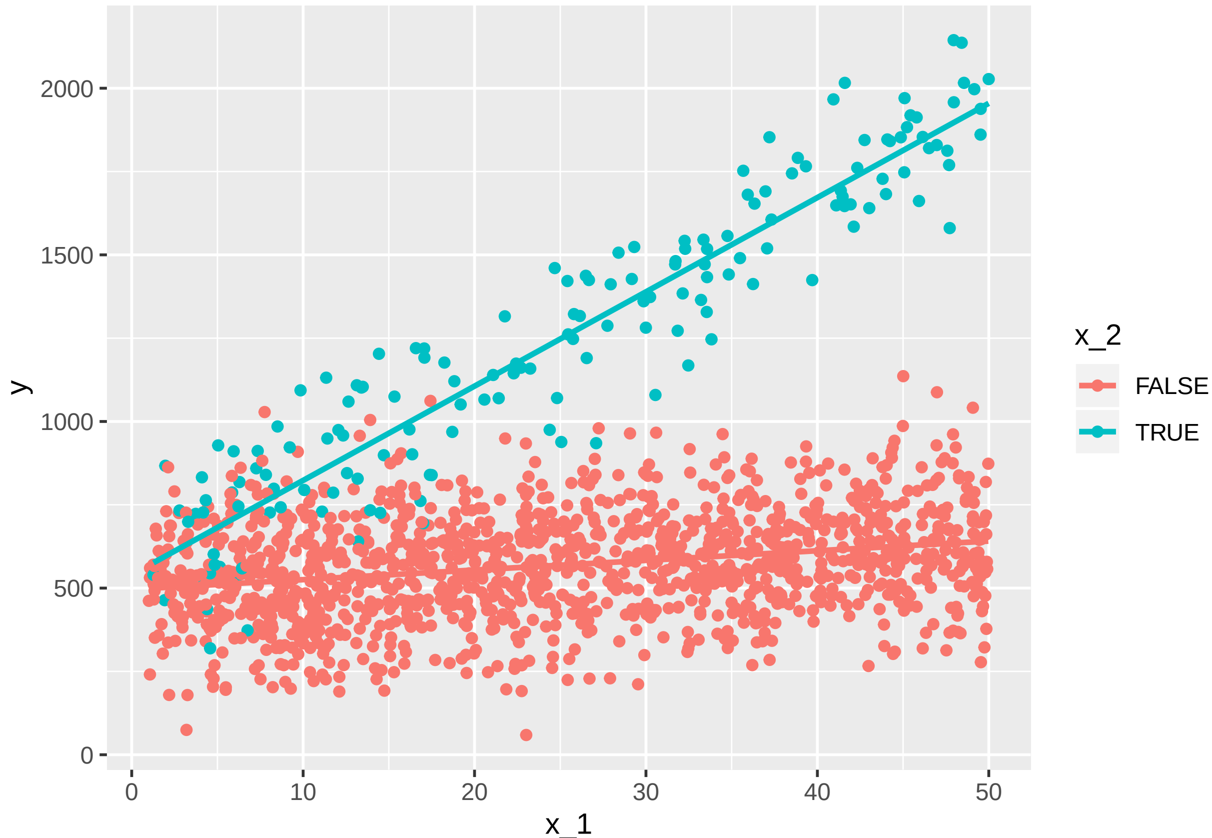
```r
# theoretical setup
x_1 <- runif(1500, 1, 50)
x_2 <- rbinom(1500, 1, .1)
x_2 <- as.logical(x_2)
eps <- rnorm(1500, 0, 150)
y <-  500 + (3 * x_1) + (59*x_2) + (25 * x_1 * x_2) + eps
df_theor <- data.frame(y, x_1, x_2)
```

Test for Association

```r
lm_theor1 <- lm(x_1 ~ x_2, data = df_theor)
summary(lm_theor1)
```

```
##
## Call:
## lm(formula = x_1 ~ x_2, data = df_theor)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -24.1605 -12.9100  -0.8012  11.8314  25.3153
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  24.6568     0.3873  63.670   <2e-16 ***
## x_2TRUE       0.7748     1.1857   0.653    0.514
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.18 on 1498 degrees of freedom
## Multiple R-squared:  0.0002849,  Adjusted R-squared:  -0.0003824
## F-statistic: 0.427 on 1 and 1498 DF,  p-value: 0.5136
```

Test for Interaction




```r
lm_theor2 <- lm(y ~ x_1 + x_2 + x_1:x_2)
summary(lm_theor2)
```

```
##
## Call:
## lm(formula = y ~ x_1 + x_2 + x_1:x_2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -503.73  -98.93   -0.04   98.51  514.97
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 496.3980     8.3206  59.659   <2e-16 ***
## x_1           2.8932     0.2929   9.877   <2e-16 ***
## x_2TRUE      43.8658    25.3354   1.731   0.0836 .
## x_1:x_2TRUE  25.3980     0.8661  29.324   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 151.2 on 1496 degrees of freedom
## Multiple R-squared:  0.7417, Adjusted R-squared:  0.7411
## F-statistic:  1432 on 3 and 1496 DF,  p-value: < 2.2e-16
```

Comments

Clearly (by setup) there is no relationship between x_1 and x_2 (unless by random chance). However, there is a clear relationship between y and x_1, x_2, and the interaction between x_1 and x_2. We checked the association between x_1 and x_2 above, where we recieved an f-statistic of 2.594 and p-value of .1075. This association is statistically insignificant. However, when looking at the interaction between x_1 and x_2 in predicting y, we see that there is a significant interaction. We recieve a t-value of 26.949 and p-value of basically 0. Here we observe that interaction does not infer association.

Problem 2.

1.

a.

$$\begin{bmatrix} 1 & 2 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 0 & 1 \end{bmatrix}' = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 0 \\ 0 & 1 & 2 & 1 \end{bmatrix}$$

b.

$$\begin{bmatrix} 1 & 3 & 1 \\ 2 & 2 & 5 \end{bmatrix} \times \begin{bmatrix} 0 & 2 & 0 \\ 1 & 1 & 1 \\ 3 & 1 & 2 \end{bmatrix} = \begin{bmatrix} 6 & 6 & 5 \\ 17 & 11 & 12 \end{bmatrix}$$

c.

$$\begin{bmatrix} 1 & 2 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \end{bmatrix} = \begin{bmatrix} \beta_0 + 2\beta_1 + \epsilon_1 \\ \beta_0 + \beta_1 + \beta_2 + \epsilon_2 \\ \beta_0 + \beta_1 + 2\beta_2 + \epsilon_3 \\ \beta_0 + \beta_2 + \epsilon_4 \end{bmatrix}$$

2.

a.

```r
A <- matrix(c(1, 4, 2, 5, 3, 6), 2, 3)
B <- matrix(c(-5, 3, 1, 0, 2, -4), 2, 3)

# Matrix Addition
A + B
```

```
##      [,1] [,2] [,3]
## [1,]   -4    3    5
## [2,]    7    5    2
```

```r
# Matrix Subtraction
A - B
```

```
##      [,1] [,2] [,3]
## [1,]    6    1    1
## [2,]    1    5   10
```

```r
# Multiplication by scalar -1
-B
```

```
##      [,1] [,2] [,3]
## [1,]    5   -1   -2
## [2,]   -3    0    4
```

b.

```r
a <- t(c(2, 0, 1, 3))
b <- c(-1, 6, 0, 9)
a %*% b
```

```
##      [,1]
## [1,]   25
```

c.

```r
# p.1
A <- matrix(c(1, 3, 2, 4), 2, 2)
B <- matrix(c(0, 2, 3, 1), 2, 2)
A%*%B
```

```
##      [,1] [,2]
## [1,]    4    5
## [2,]    8   13
```

```r
# p.2
A <- matrix(c(2, 0, 0, 3), 2, 2)
B <- matrix(c((1/2), 0, 0, (1/3)), 2, 2)
A%*%B
```

```
##      [,1] [,2]
## [1,]    1    0
## [2,]    0    1
```

d.

```r
A <- matrix(c(1, 2, 3, 4), 2, 2)
B <- matrix(c(5, 6, 7, 8), 2, 2)
A
```

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4
```

```
B
```

```
##      [,1] [,2]
## [1,]    5    7
## [2,]    6    8
```

```
# results of the matrix multiplications below are not equal
A%*%B
```

```
##      [,1] [,2]
## [1,]   23   31
## [2,]   34   46
```

```
B%*%A
```

```
##      [,1] [,2]
## [1,]   19   43
## [2,]   22   50
```

e.

```
A_trans <- t(A%*%B)

B_trans <- t(B)%*%t(A)

# two matrices below are equal, verifying that (AB)' = B'A'
A_trans
```

```
##      [,1] [,2]
## [1,]   23   34
## [2,]   31   46
```

```
B_trans
```

```
##      [,1] [,2]
## [1,]   23   34
## [2,]   31   46
```

f.

```
A <- matrix(c(0, 2, 3, 1), 2, 2)
A_square <- A%*%A
A_cube <- A%*%A_square
A_square
```

```
##      [,1] [,2]
## [1,]    6    3
## [2,]    2    7
```

```r
A_cube
```

```
##      [,1] [,2]
## [1,]    6   21
## [2,]   14   13
```

g.

```r
A <- matrix(c(2, 1, 5, 3), 2, 2)
# inverse of A
A_inverse <- solve(A)
A_inverse
```

```
##      [,1] [,2]
## [1,]    3   -5
## [2,]   -1    2
```

```r
b <- c(4, 5)
x <- A_inverse%*%b
x
```

```
##      [,1]
## [1,]  -13
## [2,]    6
```

h.

```r
A <- matrix(c(1, 3, 2, 5), 2, 2)
b <- c(7, 2)

x <- solve(A)%*%b
x
```

```
##      [,1]
## [1,]  -31
## [2,]   19
```

i.

```r
A <- matrix(c(1, 3, 2, 6), 2, 2)
b <- c(7, 2)

#x <- solve(A)%*%b
#x
```

R gives an error, stating that the system is singular. This system of equations has no solution.

Problem 3.

1.

Matrix transpose function

```
my.transpose <- function(A){
  dims <- dim(A)
  A_trans <- c()
  for (i in 1:dims[1]){
    vect <- A[i,]
    A_trans <- c(A_trans, vect)
  }
  A_trans <- matrix(A_trans, dims[2], dims[1])
  return(A_trans)
}
```

Matrix multiplication function

```
my.product <- function(A, B){
  dims_A <- dim(A)
  dims_B <- dim(B)
  vect <- c()
  for (i in 1:dims_B[2]){
    for (j in 1:dims_A[1]){
      dot_prod <- sum(A[j,] * B[,i])
      vect <- c(vect, dot_prod)
    }
  }
  AB <- matrix(vect, dims_A[1], dims_B[2])
  return(AB)
}
```

Functionality checks

```
# checking my.transpose()
A <- matrix(c(1, 1, 1, 1, 2, 1, 1, 0, 0, 1, 2, 1), 4, 3)
# my function output
my.transpose(A)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    1    1    1
## [2,]    2    1    1    0
## [3,]    0    1    2    1
```

```
# t() output
t(A)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    1    1    1
## [2,]    2    1    1    0
## [3,]    0    1    2    1
```

```r
# checking my.product()
B <- matrix(c(1, 2, 3, 2, 1, 5), 2, 3)
C <- matrix(c(0, 1, 3, 2, 1, 1, 0, 1, 2), 3, 3)
# my function output
my.product(B,C)
```

```
##      [,1] [,2] [,3]
## [1,]    6    6    5
## [2,]   17   11   12
```

```r
# %*% operator output
B%*%C
```

```
##      [,1] [,2] [,3]
## [1,]    6    6    5
## [2,]   17   11   12
```

2.

```r
my.is.symm <- function(A){
  transpose <- my.transpose(A)
  dims <- dim(A)
  n_entries <- dims[1] * dims[2]
  for (i in 1:n_entries){
    if (A[i] != transpose[i]){
      bool <- FALSE
      break
    } else{
      bool <- TRUE
    }
  }
  return(bool)
}
```

Functionality Checks

```r
B <- matrix(c(-5, 2, 7, 1, 2, 3, 3, 6, -4), 3, 3)
C <- matrix(c(-5, 1, 3, 1, 2, 6, 3, 6, -4), 3, 3)
# results from my.is.symm()
my.is.symm(B)
```

```
## [1] FALSE
```

```r
my.is.symm(C)
```

```
## [1] TRUE
```