101B - HW4

#10.6

```r
dat <- data.frame(y = c(193,230,172,91,113,125),
                  x1 = c(1.6,15.5,22.0,43.0,33.0,40.0),
                  x2 = c(851,816,1058,1201,1357,1115))


mod <- lm(y~x1+x2,dat)
summary(mod)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2, data = dat)
##
## Residuals:
##       1       2       3       4       5       6
## -24.987  24.307  11.820 -20.460  12.830  -3.511
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 350.99427   74.75307   4.695   0.0183 *
## x1           -1.27199    1.16914  -1.088   0.3562
## x2           -0.15390    0.08953  -1.719   0.1841
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 25.5 on 3 degrees of freedom
## Multiple R-squared:  0.8618, Adjusted R-squared:  0.7696
## F-statistic: 9.353 on 2 and 3 DF,  p-value: 0.05138
```

```r
#linear reg model = 350.99- 1.27x1 - 0.15x2

#Our p-value: 0.051 is greater than a = 0.05, concluding that we reject our null and our regression is

#The given t values are:
#p-value: 0.0188, 0.3562, 0.1841
#b0 = 4.695, b1 - 1.088, b2 = -0.1539

#We can conclude pvalues 2 and 3 are not significant in this model, but the p1 is significant.

#95% CI X1:
x <- lm(y~x1, dat)

confint(x, 'x1', level = 0.95)
```

```
##          2.5 %     97.5 %
## x1 -5.293793 -0.4181486
```

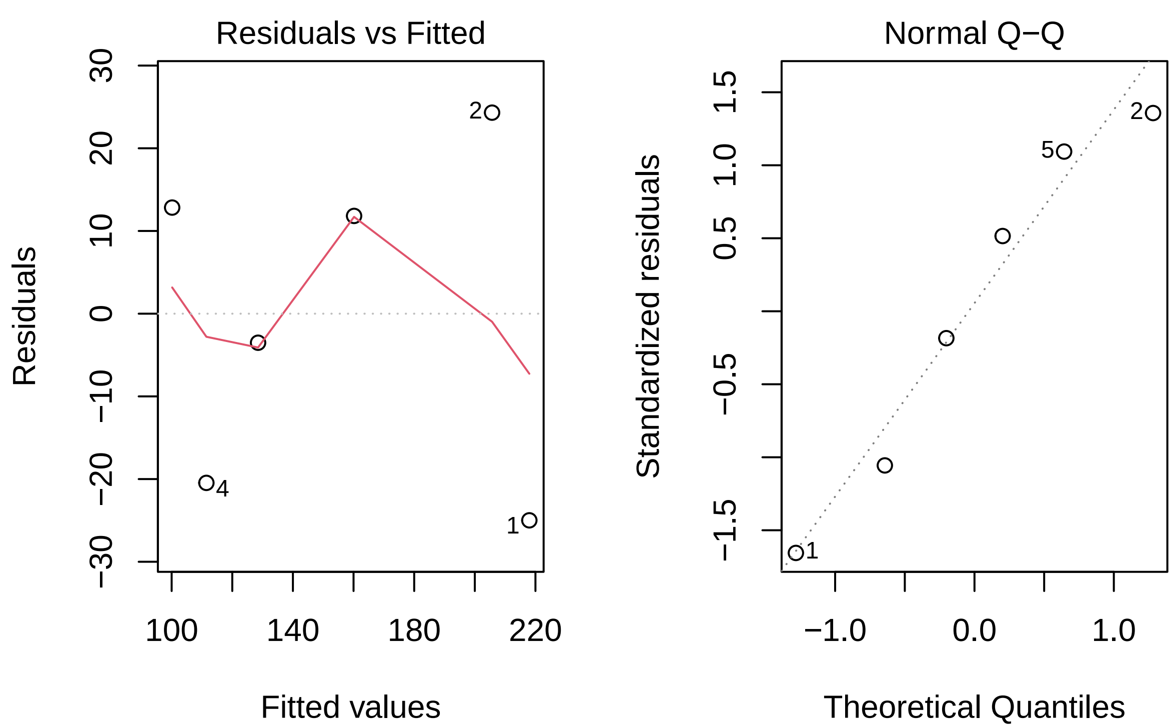
```
z <- lm(y~x2, dat)
confint(z, 'x2', level = 0.95)
```

```
##           2.5 %      97.5 %
## x2 -0.3871457 -0.07420014
```

```
res.aov <- aov(y~x1+x2,dat)
summary(res.aov)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## x1           1  10240   10240  15.751 0.0286 *
## x2           1   1921    1921   2.955 0.1841
## Residuals    3   1950     650
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow=c(1,2))
plot(res.aov,1:2)
```



```r
#The normality is fair, but the residual plot is slightly skewed.
```

##10.9 #A

```r
l1<- c(yield = 1,1,1,1,1,1,1,1,
       concentration = 1,1,2,2,1,1,2,2,
       temperature = 150,180,150,180,150,180,150,180)

m1 <- matrix(l1, nrow = 8, ncol = 3, byrow = FALSE)
colnames(m1) <- c("y","x1","x2")
m1
```

```
##      y x1  x2
## [1,] 1  1 150
## [2,] 1  1 180
## [3,] 1  2 150
## [4,] 1  2 180
## [5,] 1  1 150
## [6,] 1  1 180
## [7,] 1  2 150
## [8,] 1  2 180
```

```r
l2 <- c(1,1,1,1,1,1,1,1,
                1.00,1.00,2.00,2.00,1.00,1.00,2.00,2.00,
              150,180,150,180,150,180,150,180)
m2 <- matrix(l2, nrow = 3, ncol = 8, byrow = TRUE)

m2%*%m1
```

```
##         y   x1     x2
## [1,]    8   12   1320
## [2,]   12   20   1980
## [3,] 1320 1980 219600
```

#B #The matrix is not a diagonal. We have obtained values other than zero outside of the diagonal.

#C

```r
m1[,"x2"] <- (m1[,"x2"] - 165)/15
m1[,"x2"]
```

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

```r
m1[,"x1"] <- (m1[,"x1"] - 1.5)/0.5
m1
```

```
##      y x1 x2
## [1,] 1 -1 -1
## [2,] 1 -1  1
## [3,] 1  1 -1
## [4,] 1  1  1
```

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

```r
inv <- c(1,1,1,1,1,1,1,1,
            -1,-1,1,1,-1,-1,1,1,
            -1,1,-1,1,-1,1,-1,1)

inv1 <- matrix(inv, nrow = 3, ncol = 8, byrow = TRUE)
inv1
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## [1,]    1    1    1    1    1    1    1    1
## [2,]   -1   -1    1    1   -1   -1    1    1
## [3,]   -1    1   -1    1   -1    1   -1    1
```

```r
inv1 %*% m1
```

```
##      y x1 x2
## [1,] 8  0  0
## [2,] 0  8  0
## [3,] 0  0  8
```

```r
#The matrix is a diagonal because all of the values outside of the diagnol are equal to 0.
```

#D

```r
l1<- c(yield = 1,1,1,1,1,1,1,1,
       concentration = 1,1,2,2,1,1,2,2,
       temperature = 150,180,150,180,150,180,150,180)
m1 <- matrix(l1, nrow = 8, ncol = 3, byrow = FALSE)
colnames(m1) <- c("y","x1","x2")


m1[,"x2"] <- (m1[,"x2"] - 150)/30
m1[,"x2"]
```

```
## [1] 0 1 0 1 0 1 0 1
```

```r
m1[,"x1"] <- (m1[,"x1"] - 1.0)/1.0
m1
```

```
##      y x1 x2
## [1,] 1  0  0
## [2,] 1  0  1
## [3,] 1  1  0
## [4,] 1  1  1
## [5,] 1  0  0
## [6,] 1  0  1
## [7,] 1  1  0
## [8,] 1  1  1
```

```r
list2 <- c(1,1,1,1,1,1,1,1,
        0,0,1,1,0,0,1,1,
        0,1,0,1,0,1,0,1)
m3 <- matrix(list2, nrow = 3, ncol = 8, TRUE)
m3
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## [1,]    1    1    1    1    1    1    1    1
## [2,]    0    0    1    1    0    0    1    1
## [3,]    0    1    0    1    0    1    0    1
```

```r
m3 %*% m1
```

```
##      y x1 x2
## [1,] 8  4  4
## [2,] 4  4  2
## [3,] 4  2  4
```

```r
#The matrix is not a diagonal. We have obtained values other than zero outside of the diagonal.
```

#E From theis exercise I have learned that there are many ways to manipulate data and through vectors and matrices. Orthogonal designed matrix are easiest to deal with the orthagonal design.