HW7

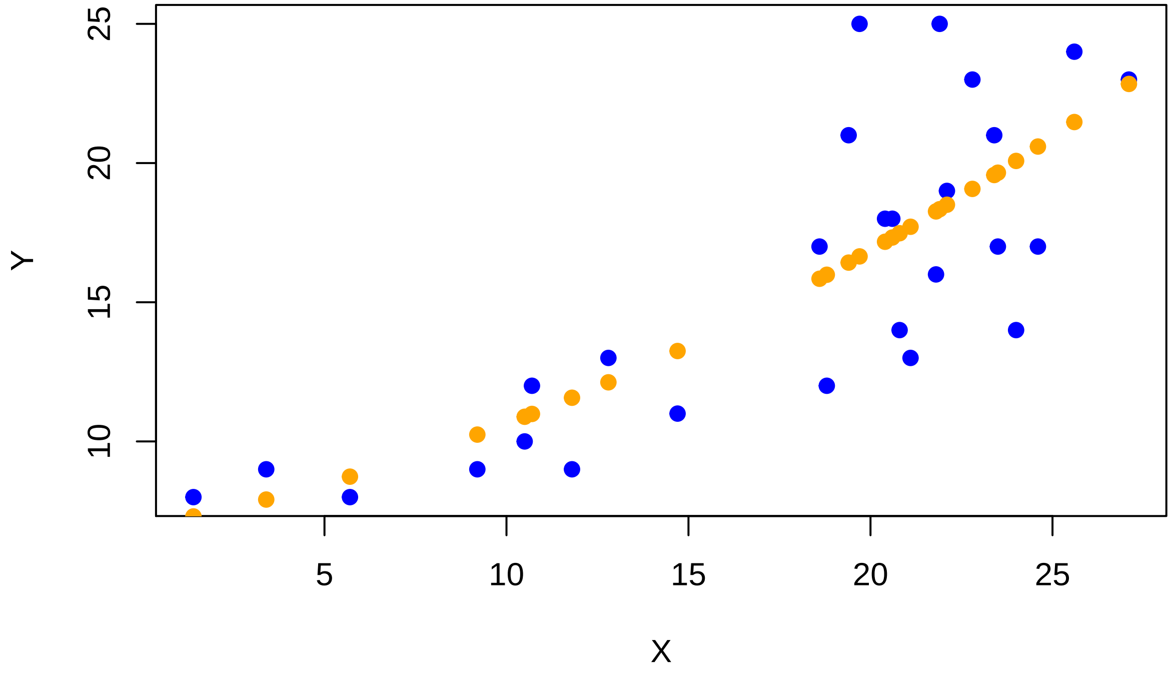
Ishita Dutta

5/30/2021

8.6

8.6a)

```
steroids = read.table("steriod+level.txt")
Y = steroids[,2]
X = steroids[,1]
X_2 = X ^ 2
qfit = lm(Y~X + X_2)
qfitvals = qfit$fitted.values
plot(X, Y, xlab="X", ylab="Y", pch=19, col = "blue")
points(X, qfitvals, xlab="X", ylab="Y", pch=19, col = "orange")
```



The quadratic line(connect dots in orange) seems to be an okay fit for this data with the R ˆ 2 value below:

```
summary(qfit)$r.squared
```

```
## [1] 0.6290178
```

8.6b)

Ho –> E(Y) = B0 + B1X1 + B2Xˆ2 Ha –> E(Y) != B0 + B1X1 + B2Xˆ2 Decision –> If p-value is less than alpha, conclude Ha

```
anova(qfit)
```

```
## Analysis of Variance Table
##
## Response: Y
##           Df Sum Sq Mean Sq F value    Pr(>F)
## X          1 485.71  485.71 39.9436 1.559e-06 ***
## X_2        1   9.11    9.11  0.7495    0.3952
## Residuals 24 291.84   12.16
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Conclusion –> p-value of B2 = 0.3952 > 0.01 = alpha, and B1 = 1.559e-06 < 0.01 = alpha so reject null hypothesis

8.6c)

```
mse = anova(qfit)[’Residuals’, ’Mean Sq’]
b0 = qfit$coefficients[1]
b1 = qfit$coefficients[2]
b2 = qfit$coefficients[3]
y_10 = qfit$fitted.values[10]
y_15 = qfit$fitted.values[15]
y_20 = qfit$fitted.values[20]
pse.yhat = sqrt(mse/sum(((X + X_2) - mean(X + X_2))^2))
B <-1- qt(.99/(2 * 2), length(X) - 3)
bh.lower10 <- y_10 - B * pse.yhat
bh.upper10 <- y_10 + B * pse.yhat
bh.lower15 <- y_15 - B * pse.yhat
bh.upper15 <- y_15 + B * pse.yhat
bh.lower20 <- y_20 - B * pse.yhat
bh.upper20 <- y_20 + B * pse.yhat

cat("10 Interval: [", bh.lower10,",",bh.upper10,"]\n")
```

```
## 10 Interval: [ 17.16747 , 17.17808 ]
```

```
cat("15 Interval: [", bh.lower15,",",bh.upper15,"]\n")
```

```
## 15 Interval: [ 11.56391 , 11.57453 ]
```

```
cat("20 Interval: [", bh.lower20,",",bh.upper20,"]\n")
```

```
## 20 Interval: [ 20.07183 , 20.08245 ]
```

8.6d)

Predicting 15 Interval: [11.56512 , 11.57332]. Interpreting true level of steroid in a 15 year old female is 99% confidence of being in this range.

8.6e)

Ho –> B2 = 0 Ha –> B2 != 0

```
n=length(Y)
anova(qfit)
```

```
## Analysis of Variance Table
##
## Response: Y
##           Df Sum Sq Mean Sq F value    Pr(>F)
## X          1 485.71  485.71 39.9436 1.559e-06 ***
## X_2        1   9.11    9.11  0.7495    0.3952
## Residuals 24 291.84   12.16
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
cat("F* --> ",anova(qfit)[2,4], "\n")
```

```
## F* -->  0.749537
```

```
cat("F(0.99, 1, n-4) --> ",qf(0.99, 1,n - 4), "\n")
```

```
## F(0.99, 1, n-4) -->  7.881134
```

Decision Rule –> Reject null hypothesis if F-statistic is greater than critical value at level 0.01. Conclusion–> Fail to reject the null hypothesis.

8.6f)

The regression function expresses the amount of predicted steroid in a female at a certain age X. The quadratic relation shows that there is a slightly higher increase than the previous year per every year increase in X.

8.15

8.15a)

```
copier = read.table("copier+maintenance.txt")
Y2 = copier[,1]
X1 = copier[,2]
X2 = read.table("X2.txt")[,1]
```

y = The response variable, number X1 = The first predictor variable, time X2 = The second predictor variable, size of 0 or 1 e = the residual error (unmeasured variable) from observations B0 = Y intercept B1 = Regression coefficient pertaining to X1 B2 = Regression coefficient pertaining to X2

8.15b)

```
fit2 = lm(Y2~X1 + X2)
summary(fit2)
```

```
##
## Call:
## lm(formula = Y2 ~ X1 + X2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.5390  -4.2515   0.5995   6.5995  14.9330
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.9225     3.0997  -0.298    0.767
## X1           15.0461     0.4900  30.706   <2e-16 ***
## X2            0.7587     2.7799   0.273    0.786
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.011 on 42 degrees of freedom
## Multiple R-squared:  0.9576, Adjusted R-squared:  0.9556
## F-statistic: 473.9 on 2 and 42 DF,  p-value: < 2.2e-16
```

Y = -0.9225 + 15.0461 X + 0.7587(Include only if small)

8.15c)

```
confint(fit2, level = .95)
```

```
##                  2.5 %    97.5 %
## (Intercept) -7.177891  5.332945
## X1          14.057283 16.035004
## X2          -4.851254  6.368698
```

We have 95% confidence that service time will fall between 14.06 and 16.04.

8.15d)

```
fit2.2 = lm( Y2 ~ X2)
summary(fit2)$r.squared
```

```
## [1] 0.9575707
```

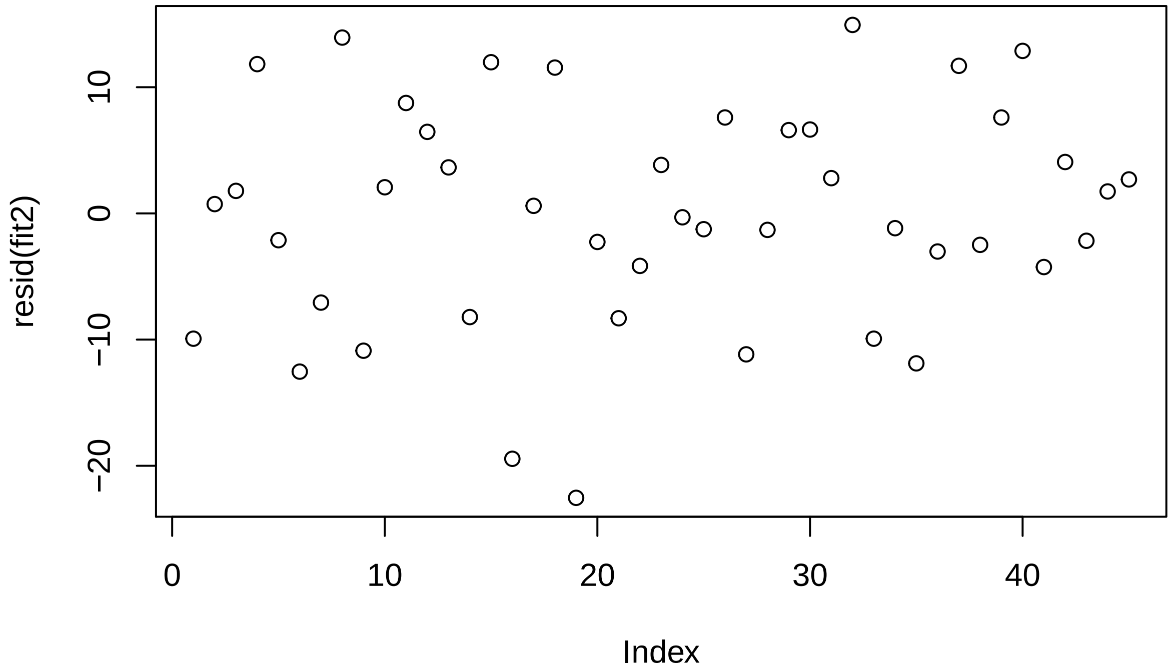
```
summary(fit2.2)$r.squared
```

```
## [1] 0.005051256
```

Without X1, there is no correlation in the data and X2, making X1 essential for the data to be coherent.

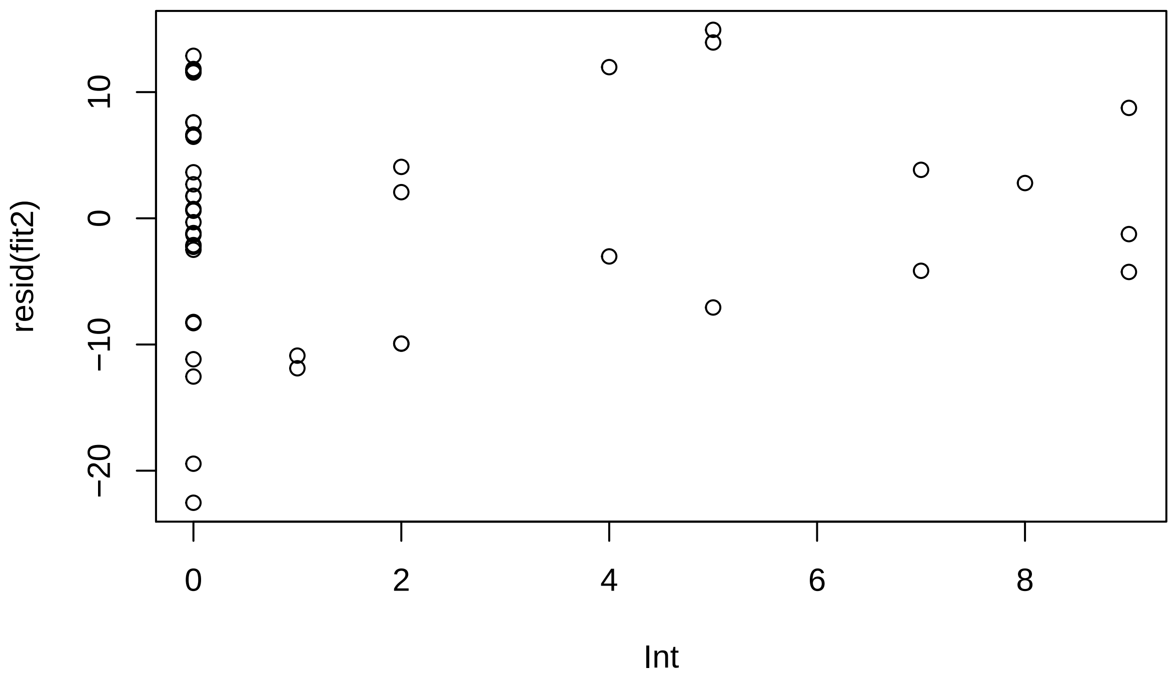
8.15e)

```
plot(resid(fit2))
```



```
Int = X1 * X2

plot(Int, resid(fit2))
```



It does not seem beneficial to add an interaction term from the residuals.

8.21

8.21a)

Hard hat: E{Y} = (B0 + B2) + B1X1 Bump cap: E{Y} = (B0 +B3) + B1X1 None: E{Y} = B0 + B1X1

8.21b)

Ho: B3 >= 0 Ha: B3 < 0

H0: B2 = B3 HA: B2 != B3

9.11

9.11a)

```
data1 = read.table("job+proficiency.txt")
names(data1)=c("Y","x1","x2","x3","x4")
lm.a=lm(Y~., data1)
library(leaps)
```

```r
all<-regsubsets(Y~., data=data1,nbest=1)
best3 = order(summary(all)$adjr2)[1:4]
kable(cbind(summary(all)$which[best3,], adjr2=summary(all)$adjr2[best3]))
```

	(Intercept)	x1	x2	x3	x4	adjr2
1	1	0	0	1	0	0.7962344
2	1	1	0	1	0	0.9269043
4	1	1	1	1	1	0.9554702
3	1	1	0	1	1	0.9560482

9.11b)

The other factor we can use here is which variables are being considered and whether they really make a difference in the grand scheme. For example, X2 seems to not make too much of a difference on the correlation coefficient, whereas X1 suddenly seems very useful, creating a 0.11 increase in the correlation coefficient.

9.16

9.16a)

```r
data1 = read.table("kidney+function.txt")
names(data1)=c("Y","x1","x2","x3")
data2=data.frame(Y=data1[,1],X1=data1[,2]-mean(X1),X2=data1[,3]-mean(X2),X3=data1[,4]-2.286957,X1_2=I((
lm.a=lm(Y~., data2)
library(leaps)
all<-regsubsets(Y~., data=data2,nbest=1)
best6 = order(summary(all)$cp)[1:6]
library(knitr)
kable(cbind(summary(all)$which[best6,], cp=summary(all)$cp[best6]))
```

	(Intercept)	X1	X2	X3	X1_2	X2_2	X3_2	cp
4	1	0	1	1	1	0	1	5.488956
3	1	0	1	1	1	0	0	6.079702
5	1	1	1	1	1	0	1	6.297696
6	1	1	1	1	1	1	1	7.000000
2	1	0	1	0	1	0	0	26.707302
1	1	0	0	0	1	0	0	49.296232

9.16b)

Though the numbers are close, the best regressions we want to find are for X4, X3, and X5 in that order. This is because they are the three smallest Cp values, indicating the three values with the least amount of error.