Assignment2 Rpart

-"Mert Göksel" -"Bilge Özkır" -"Aisuluu Baktybekova"

6/14/2021

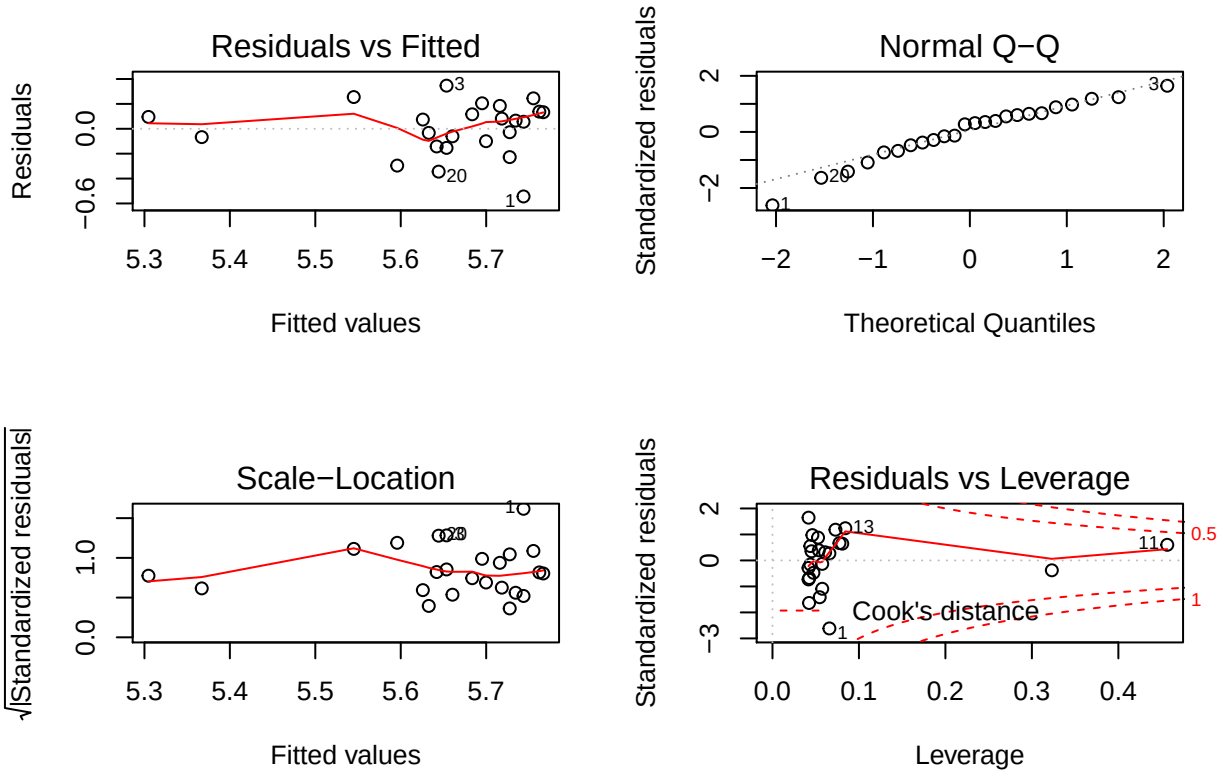
Q3

```
library(tidyverse, quietly = T)
```

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

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.6
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

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

```
sweet.ind <- c(
  5.2,
  5.5,
  6.0,
  5.9,
  5.8,
  6.0,
  5.8,
  5.6,
  5.6,
  5.9,
  5.4,
  5.6,
  5.8,
  5.5,
  5.3,
  5.3,
  5.7,
  5.5,
  5.7,
  5.3,
  5.9,
  5.8,
```

```
  5.8,
  5.9
)
pectin <- c(
  220,
  227,
  259,
  210,
  224,
  215,
  231,
  268,
  239,
  212,
  410,
  256,
  306,
  259,
  284,
  383,
  271,
  264,
  227,
  263,
  232,
  220,
  246,
  241
)
df <- data.frame(pectin=pectin, sweet.ind=sweet.ind)

#a
fit1 <- fit <- lm(data = df, formula = sweet.ind~pectin)
par(mfrow = c(2, 2))
plot(fit1)
```

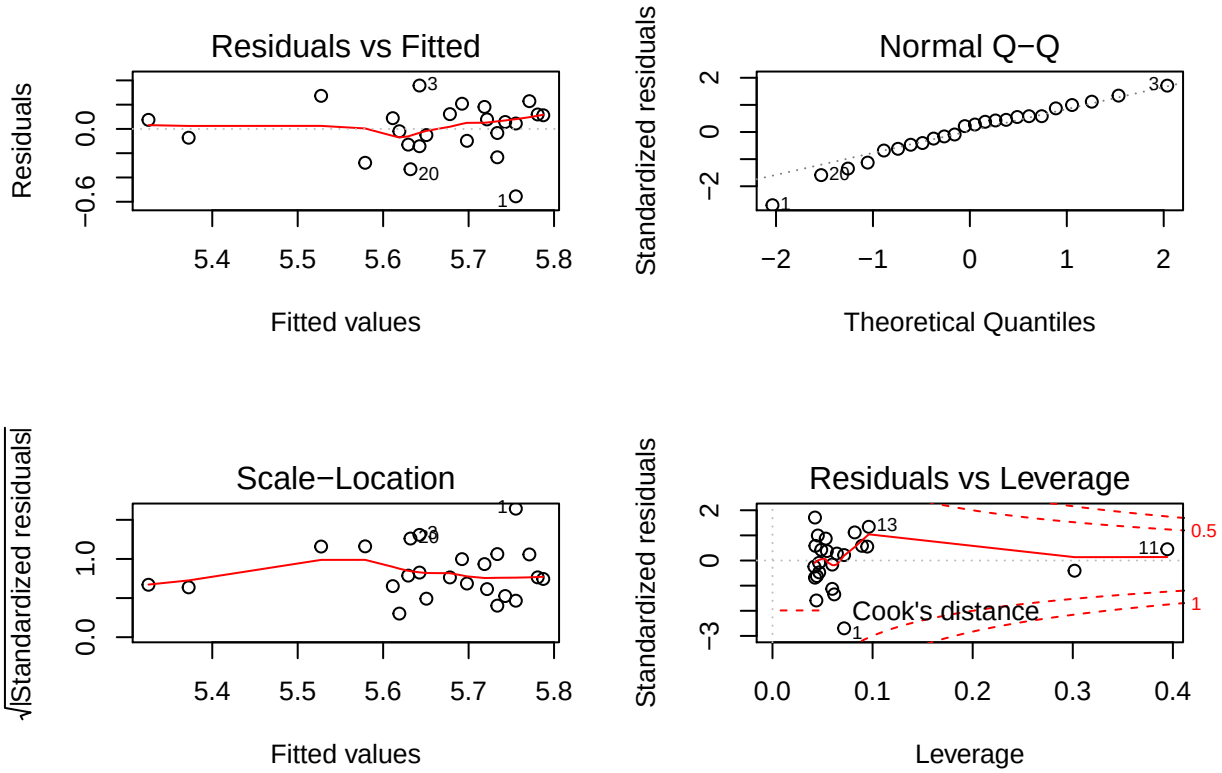



```
#From residuals vs fitted we see that no apparent pattern exist but the line is
#not smooth thus we cant say the data is linear. we can try transforming data

#From qqplot we can suspect this being not normal as first 3 points are very
#far away from the line, testing with shapiro is advised.
shapiro.test(fit1$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  fit1$residuals
## W = 0.9608, p-value = 0.4547
```

```
#p value is bigger than 0.05 thus we can assume normality.

#homogenity of variance is not constant. transformation required.

#no points in leverage plot is higher or lesser than |3| thus is good.

#Conclusion: transformation required, testing normality is advised.

#Because these assumptions are required to have a model that works as intended.
#Drawing a line is easy, but drawing a line that can give you good predictions is
#after all what we are after.
```

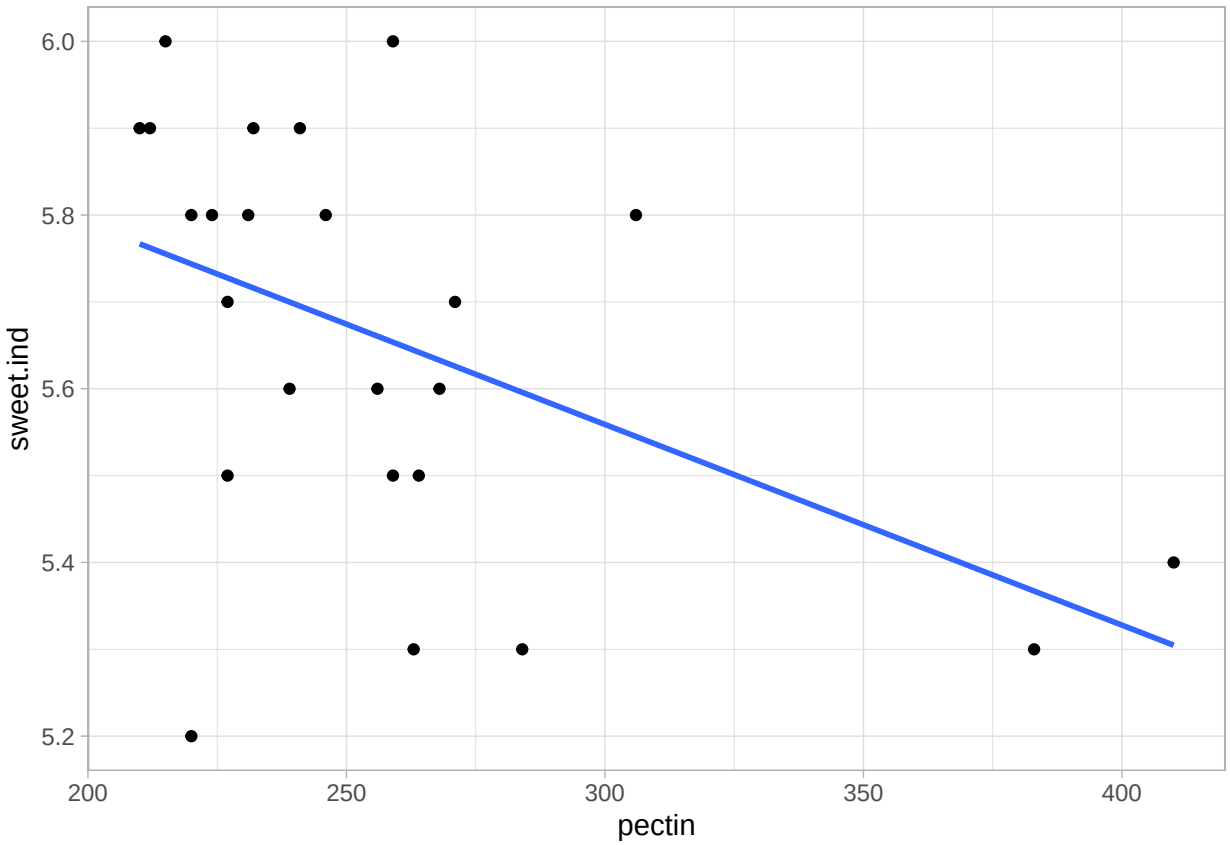
```
#b
fit2 <- lm(data = df, formula = sweet.ind~log(pectin)) #this works better
par(mfrow = c(2, 2))
plot(fit2)
```




```
#all extremities are lower in this version. Meaning this version of regression
#is better than non log version
fit2$call
```

```
## lm(formula = sweet.ind ~ log(pectin), data = df)
```

```
df %>% ggplot(aes(x=pectin, y=sweet.ind)) + geom_point() +
  geom_smooth(method = "lm", se = F, formula = y~x) + theme_light()
```




```
#c
summary(fit2)
```

```
##
## Call:
## lm(formula = sweet.ind ~ log(pectin), data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.55526 -0.10588  0.05096  0.11984  0.35744
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.4801     1.4568   6.508 1.51e-06 ***
## log(pectin)  -0.6906     0.2631  -2.625   0.0155 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2136 on 22 degrees of freedom
## Multiple R-squared:  0.2385, Adjusted R-squared:  0.2038
## F-statistic: 6.889 on 1 and 22 DF,  p-value: 0.01548
```

```
#both B0 and B1 have p values less than 0.05 thus they are both significant
#R squared value is low thus this linear model doesnt have much precision
#F statistic has value 6.889 with degrees of freedom 1, 22
6.889 > df(0.05, 1, 22)
```

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

```
#Thus at alpha = 0.05 this regression model is significant
#Seems like a non linear model would fit better from graph

#d
anova(fit2)
```

```
## Analysis of Variance Table
##
## Response: sweet.ind
##             Df  Sum Sq  Mean Sq F value  Pr(>F)
## log(pectin)  1 0.31436 0.314361  6.8886 0.01548 *
## Residuals   22 1.00397 0.045635
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#as p value is less than 0.05 this model is significant.

#e
predict(fit2, newdata = data.frame(pectin=300))
```

```
##        1
## 5.541073
```

Q4

```
df2 <- cbind(Mechanical=c(50,40), Electrical=c(30,30), Other=c(60,40))
rownames(df2) <- c("Design1", "Design2")
df2 <- as.table(df2)
chisq.test(df2)
```

```
##
##  Pearson's Chi-squared test
##
## data:  df2
## X-squared = 1.5332, df = 2, p-value = 0.4646
```

```
#As p value is higher than 0.05 we can conclude that rows are independent
```

Q5

```
df3 <- cbind(Case=c(64,230-64), Control=c(270-134,134))
rownames(df3) <- c("Exposed", "Unexposed")
df3 <- as.table(df3)

caseinex <- df3[1,1]/sum(df3[1,])
caseinex
```

```
## [1] 0.32
```

```
odds <- (df3[1,1]/sum(df3[1,]))/(df3[2,1]/sum(df3[2,]))
odds
```

```
## [1] 0.5783133
```

```
#odds ratio here means that being a case while exposed to caffeine is 0.5783..
#times more likely than being a case when not exposed. Meaning caffeine lowers
#the odds of having parkinsons. If we are talking just from this table then this is
#indeed the case but to have a solid idea more testing and research is required.
```