final_project_419

Tyler Chun

11/24/2021

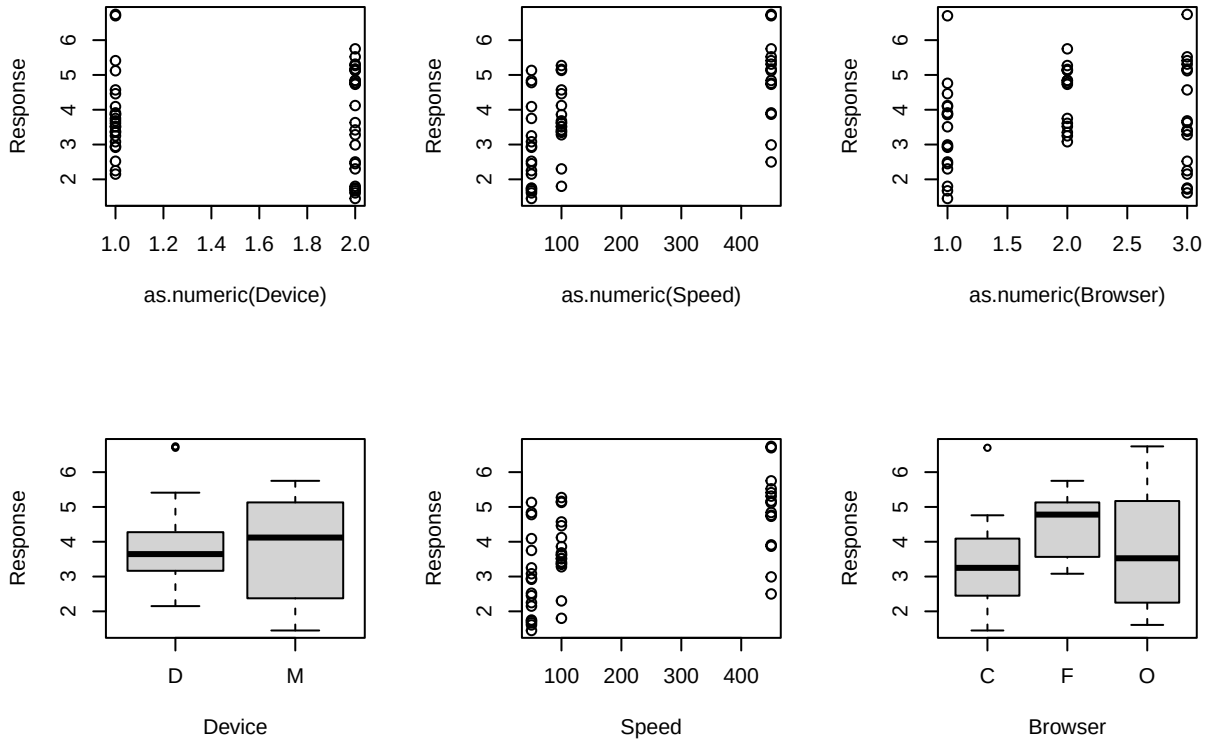
Approach: Remove point 8 and Treat Speed as numeric and Box-cox

- Treat Speed as numeric
- Transformation of response

```
df = read.csv("df.csv")
# remove point 8
df <- df[-8,]

# Do initial Analysis and fit model
Response = c(df$Replicate.1,df$Replicate.2,df$Replicate.3)

# Change Speed to original speed numbers
df$Speed_f=as.factor(df$Speed)

df$Speed[df$Speed_f==1]=50
df$Speed[df$Speed_f==2]=100
df$Speed[df$Speed_f==3]=450

df
```

```
##     X Run Device Speed Browser Replicate.1 Replicate.2 Replicate.3 Speed_f
## 1   1   1      D    50       C        2.95        4.09        2.92       1
## 2   2   2      D    50       F        3.75        3.08        3.25       1
## 3   3   3      D    50       O        2.25        2.15        2.52       1
## 4   4   4      D   100       C        4.46        3.51        3.86       2
## 5   5   5      D   100       F        3.35        3.61        3.52       2
## 6   6   6      D   100       O        3.68        3.38        4.57       2
## 7   7   7      D   450       C        6.70        3.87        3.91       3
## 9   9   9      D   450       O        5.12        5.41        6.74       3
## 10 10  10      M    50       C        2.45        1.67        1.45       1
## 11 11  11      M    50       F        5.13        4.84        4.78       1
## 12 12  12      M    50       O        1.61        1.75        1.72       1
## 13 13  13      M   100       C        1.80        4.12        2.30       2
## 14 14  14      M   100       F        5.27        5.13        5.16       2
## 15 15  15      M   100       O        3.42        3.63        3.28       2
## 16 16  16      M   450       C        2.99        4.76        2.50       3
## 17 17  17      M   450       F        5.75        4.84        4.73       3
## 18 18  18      M   450       O        5.31        5.52        5.17       3
```

```r
# After removing 8, only has 17 rows
df$Device = as.factor(df$Device)
df$Browser = as.factor(df$Browser)
new_df =
data.frame(Device = rep(df$Device,3),
Speed = rep(df$Speed,3),
Browser = rep(df$Browser,3),
Response,
replicate = as.factor(c(rep(1,17),rep(2,17),rep(3,17))))


par(mfrow=c(2,3))
plot(data= new_df, Response~ as.numeric(Device) + as.numeric(Speed)+as.numeric(Browser))
plot(data= new_df, Response~ (Device) + (Speed)+(Browser))
```




```r
par(mfrow=c(1,1))
m1 <-lm(data=new_df, Response~Device + Speed + Browser +
          Device:Speed + Device:Browser + Speed:Browser +
          replicate)
summary(m1)
```

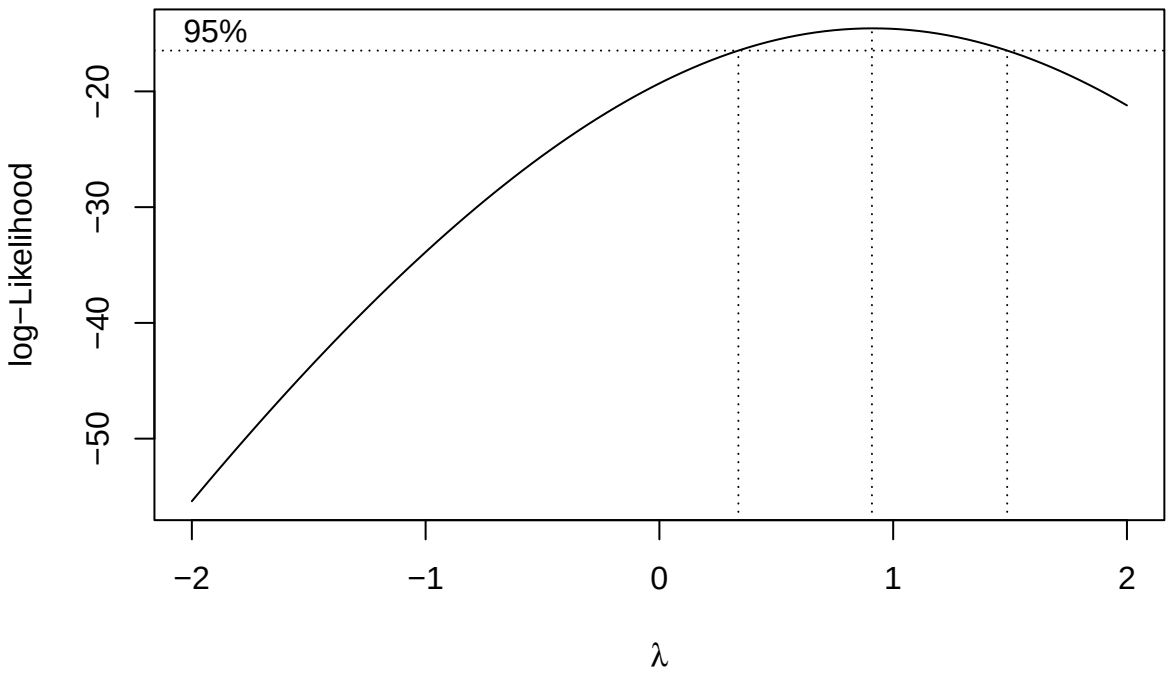
```
##
## Call:
## lm(formula = Response ~ Device + Speed + Browser + Device:Speed +
##     Device:Browser + Speed:Browser + replicate, data = new_df)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.00960 -0.53057 -0.09318  0.26281  1.78334
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.470e+00  3.816e-01   9.095 3.50e-11 ***
## DeviceM          -1.370e+00  4.537e-01  -3.020  0.00445 **
## Speed             3.214e-03  1.223e-03   2.628  0.01221 *
## BrowserF          2.506e-02  4.594e-01   0.055  0.95677
## BrowserO         -9.179e-01  4.549e-01  -2.018  0.05052 .
## replicate2       -3.706e-02  2.591e-01  -0.143  0.88701
## replicate3       -2.124e-01  2.591e-01  -0.820  0.41746
## DeviceM:Speed     5.604e-05  1.406e-03   0.040  0.96841
## DeviceM:BrowserF  2.978e+00  5.885e-01   5.061 1.03e-05 ***
## DeviceM:BrowserO  8.689e-01  5.036e-01   1.725  0.09239 .
## Speed:BrowserF   -3.022e-03  1.856e-03  -1.628  0.11164
## Speed:BrowserO    4.339e-03  1.415e-03   3.067  0.00392 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7554 on 39 degrees of freedom
## Multiple R-squared:  0.7525, Adjusted R-squared:  0.6827
## F-statistic: 10.78 on 11 and 39 DF,  p-value: 9.821e-09
```

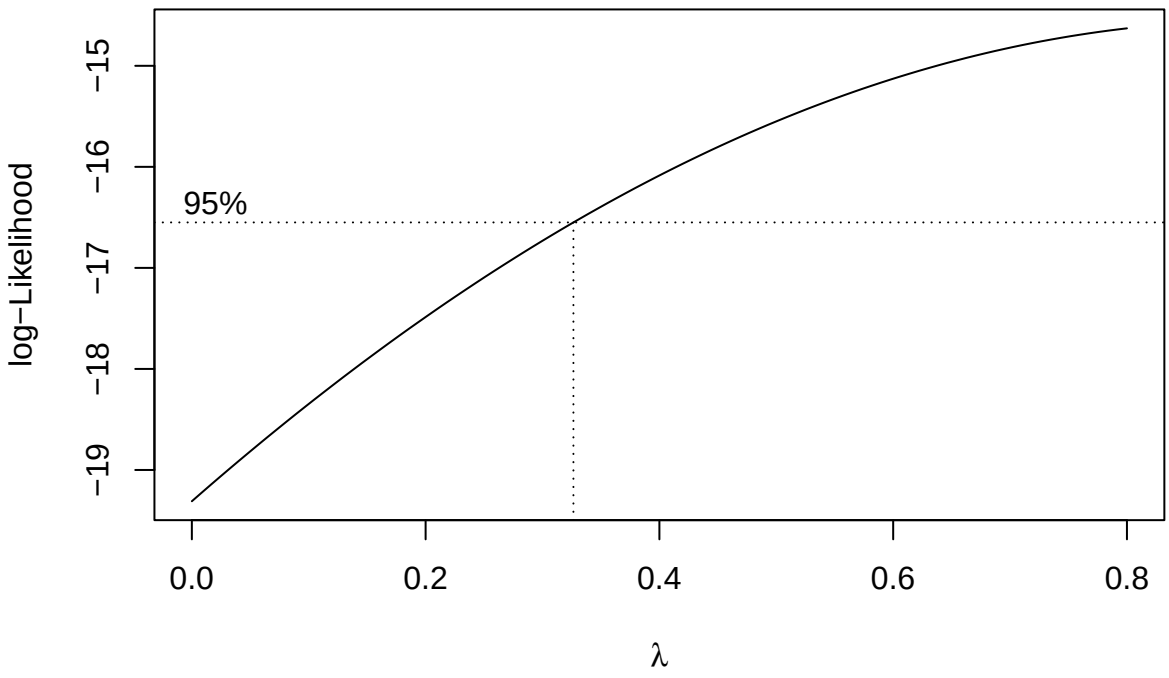
```
anova(m1)
```

```
## Analysis of Variance Table
##
## Response: Response
##                Df  Sum Sq Mean Sq F value    Pr(>F)
## Device          1  0.1731  0.1731  0.3033 0.5849720
## Speed           1 28.7623 28.7623 50.3999 1.566e-08 ***
## Browser         2 14.8588  7.4294 13.0184 4.668e-05 ***
## replicate       2  0.4374  0.2187  0.3833 0.6841741
## Device:Speed    1  2.4807  2.4807  4.3469 0.0436699 *
## Device:Browser  2 10.3653  5.1826  9.0815 0.0005781 ***
## Speed:Browser   2 10.5881  5.2940  9.2767 0.0005063 ***
## Residuals      39 22.2566  0.5707
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

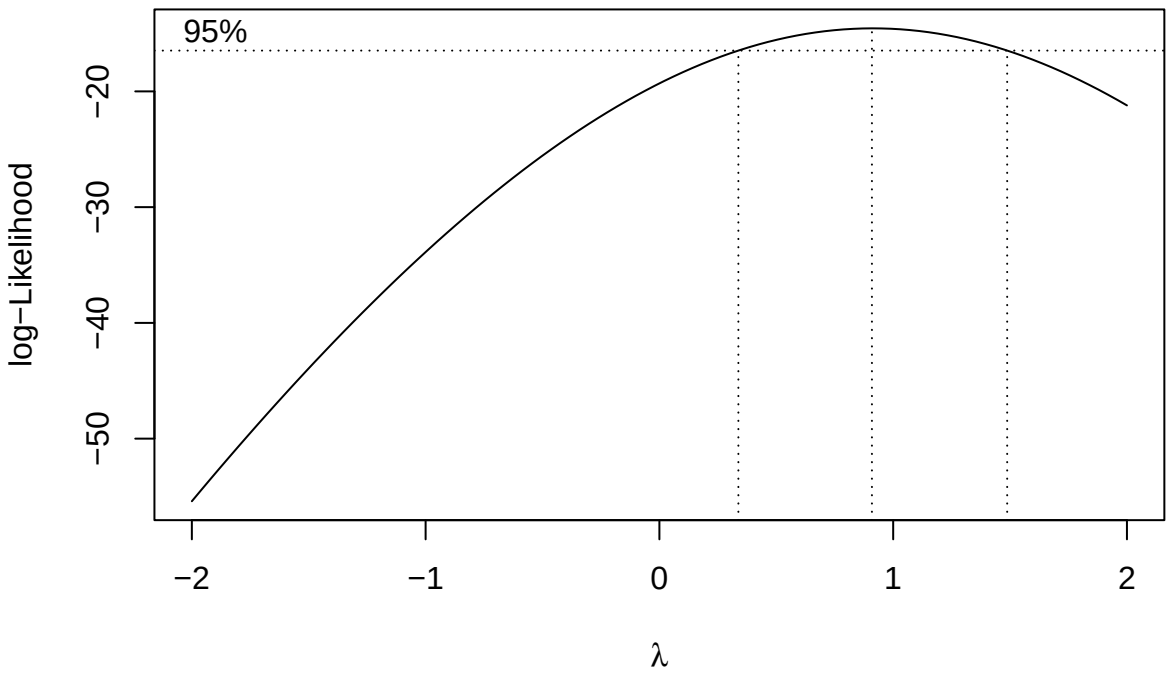
```
library(MASS)


# Box-Cox Transformation
boxcox(m1, plotit=T)
```




```
boxcox(m1, plotit=T, lambda = seq(0, 0.8, by=0.1))
```




```
# To see the exact best lambda:
lmod1_bc <- boxcox(m1)
```



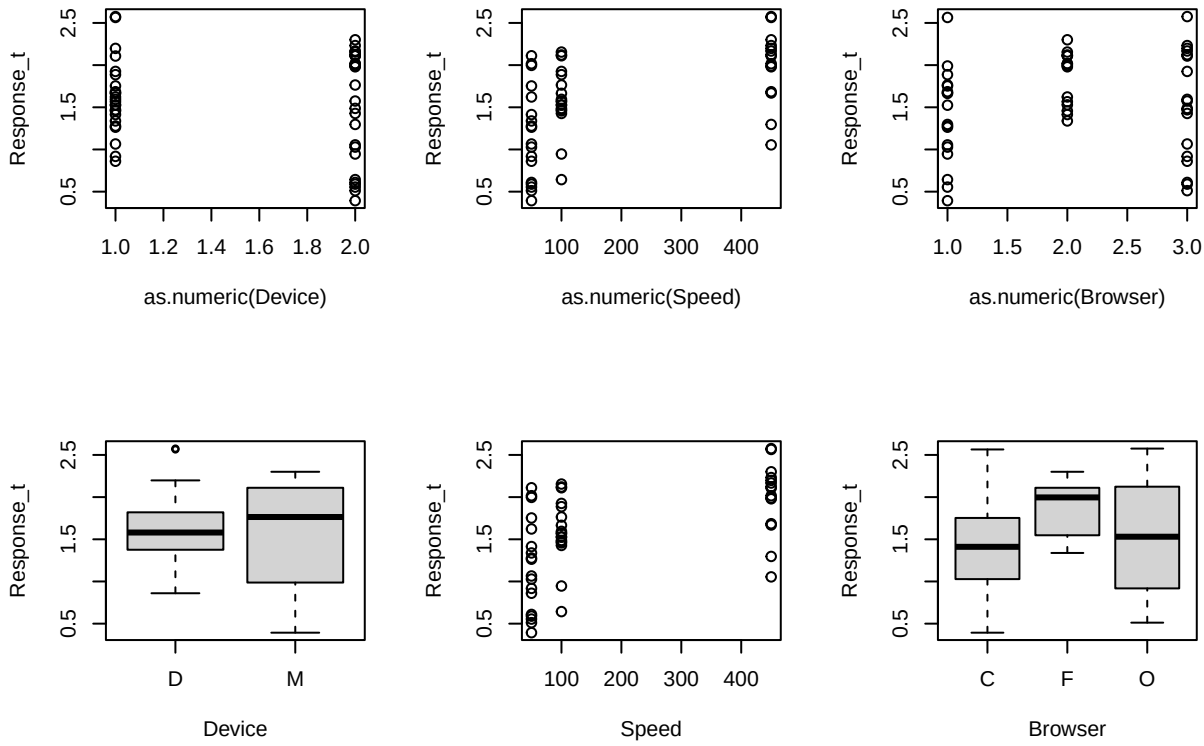

```
lambda <- lmod1_bc$x[which.max(lmod1_bc$y)]
lambda
```

```
## [1] 0.9090909
```

```
#2. We see the best lambda is 0.3030303, round to 0.3. Create a new transformed response using Box-Cox
lambda <- 0.3
Response_t <- (Responseˆlambda - 1) / lambda
```

```
# Re-do all analysis with transformed y

# After removing 8, only has 17 rows
# Set variables to factors
df$Device = as.factor(df$Device)
df$Browser = as.factor(df$Browser)

# Create a new dataframe with the variables and transformed response
new_df =
data.frame(Device = rep(df$Device,3),
Speed = rep(df$Speed,3),
Speed_f = rep(df$Speed_f,3),
Browser = rep(df$Browser,3),
Response_t,
replicate = as.factor(c(rep(1,17),rep(2,17),rep(3,17))))
```

```r
# Plot
par(mfrow=c(2,3))
plot(data= new_df, Response_t~ as.numeric(Device) + as.numeric(Speed)+as.numeric(Browser))
plot(data= new_df, Response_t~ (Device) + (Speed)+(Browser))
```




```r
par(mfrow=c(1,1))
m1 <-lm(data=new_df, Response_t~(Device + Speed + Browser)^2 + replicate)
summary(m1)
```

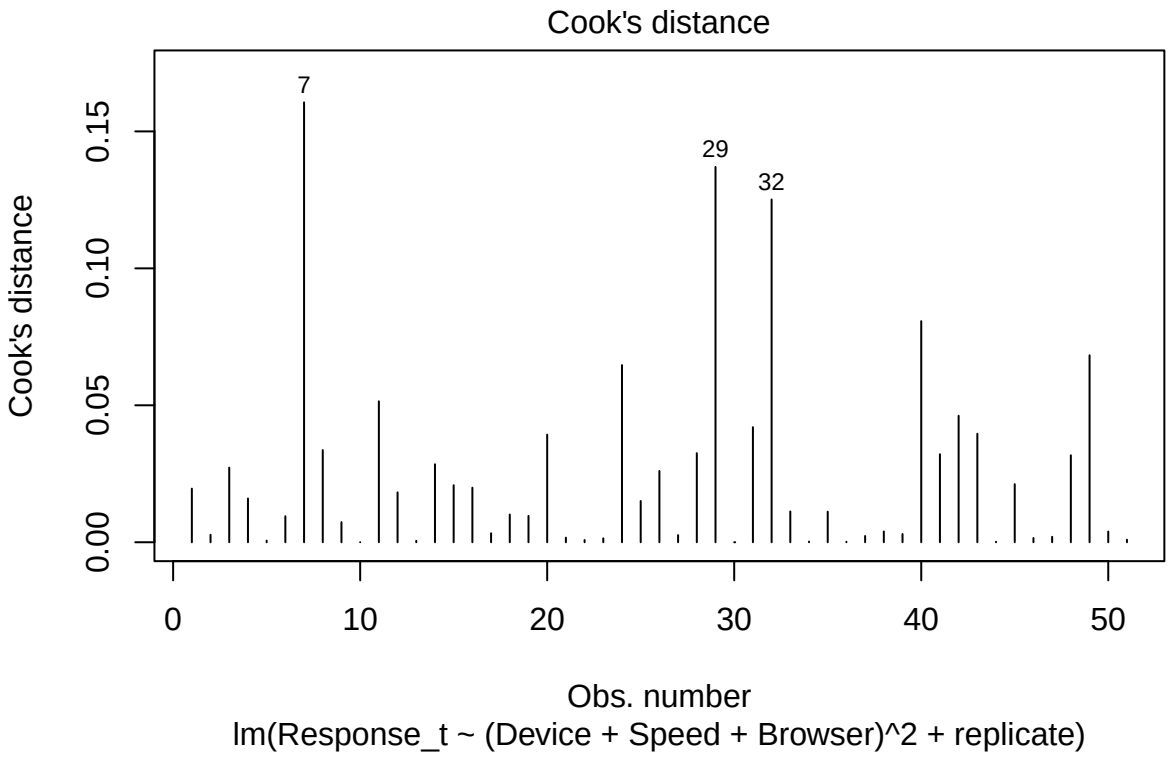
```
##
## Call:
## lm(formula = Response_t ~ (Device + Speed + Browser)^2 + replicate,
##     data = new_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.44132 -0.22614 -0.02623  0.15221  0.81659
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1.4923896  0.1641193   9.093 3.52e-11 ***
## DeviceM          -0.7104327  0.1951586  -3.640 0.000789 ***
## Speed             0.0011560  0.0005260   2.198 0.033964 *
## BrowserF          0.0462369  0.1975771   0.234 0.816194
## BrowserO         -0.3933528  0.1956447  -2.011 0.051324 .
```

```
## replicate2        0.0069542  0.1114451   0.062 0.950563
## replicate3       -0.0809691  0.1114451  -0.727 0.471847
## DeviceM:Speed     0.0004296  0.0006047   0.710 0.481673
## DeviceM:BrowserF  1.2699784  0.2531084   5.018 1.19e-05 ***
## DeviceM:BrowserO  0.4030419  0.2166103   1.861 0.070342 .
## Speed:BrowserF   -0.0015057  0.0007985  -1.886 0.066787 .
## Speed:BrowserO    0.0016465  0.0006086   2.705 0.010065 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3249 on 39 degrees of freedom
## Multiple R-squared:  0.7361, Adjusted R-squared:  0.6617
## F-statistic:  9.89 on 11 and 39 DF,  p-value: 3.126e-08
```

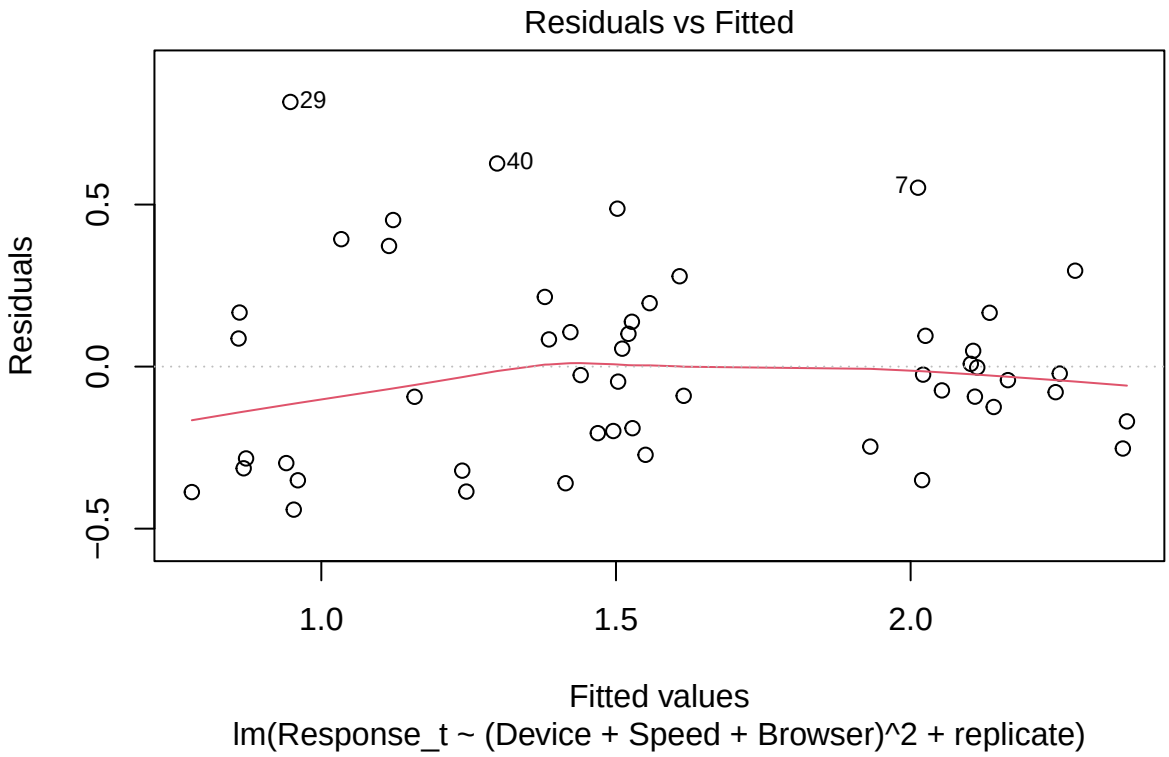
```
anova(m1)
```

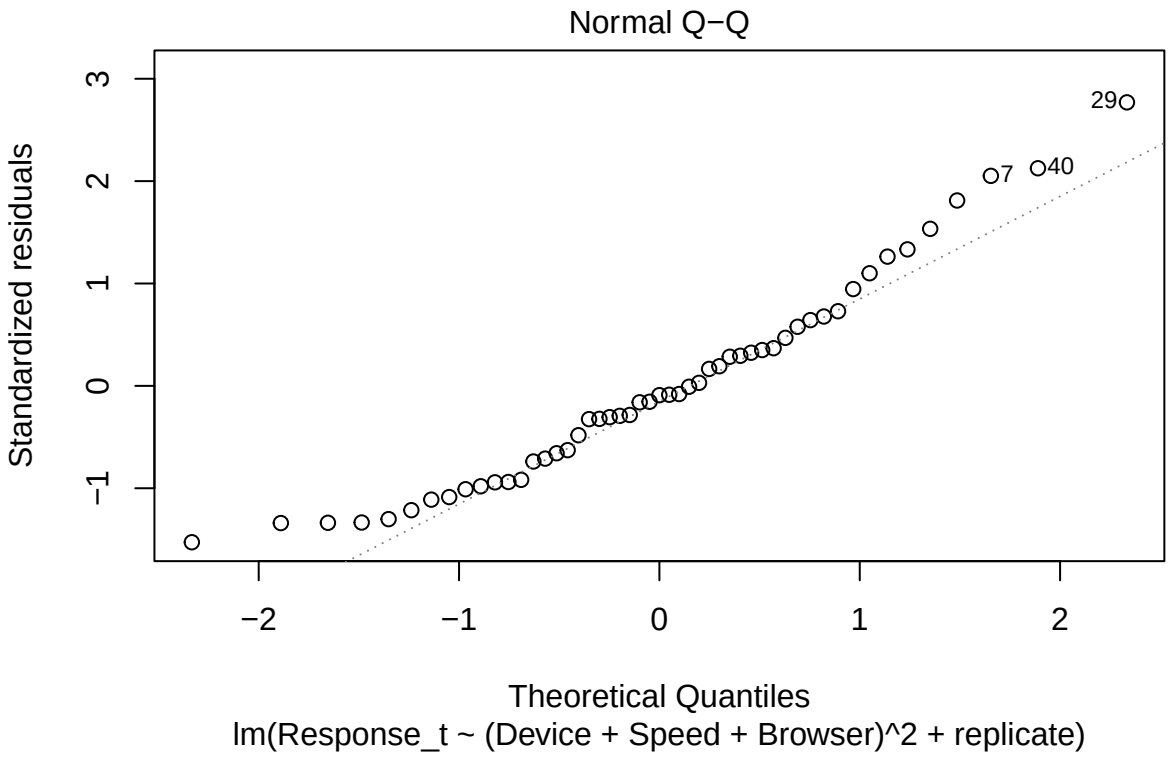
```
## Analysis of Variance Table
##
## Response: Response_t
##                Df Sum Sq Mean Sq F value    Pr(>F)
## Device          1 0.1179  0.1179  1.1172 0.2970302
## Speed           1 4.5911  4.5911 43.4885 7.757e-08 ***
## Browser         2 2.8505  1.4253 13.5007 3.503e-05 ***
## replicate       2 0.0812  0.0406  0.3847 0.6831919
## Device:Speed    1 0.1954  0.1954  1.8510 0.1814746
## Device:Browser  2 1.8247  0.9123  8.6421 0.0007820 ***
## Speed:Browser   2 1.8235  0.9117  8.6364 0.0007851 ***
## Residuals      39 4.1172  0.1056
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

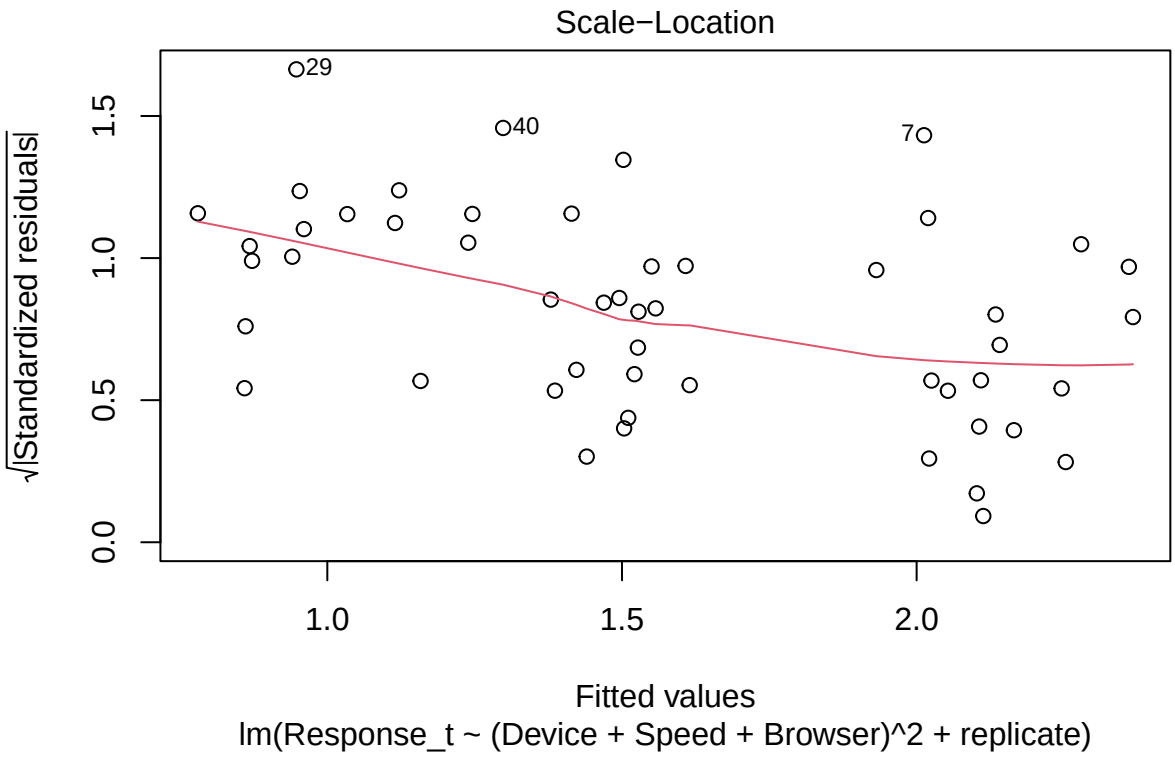
```
# Cook's distance
plot(m1,4)
```

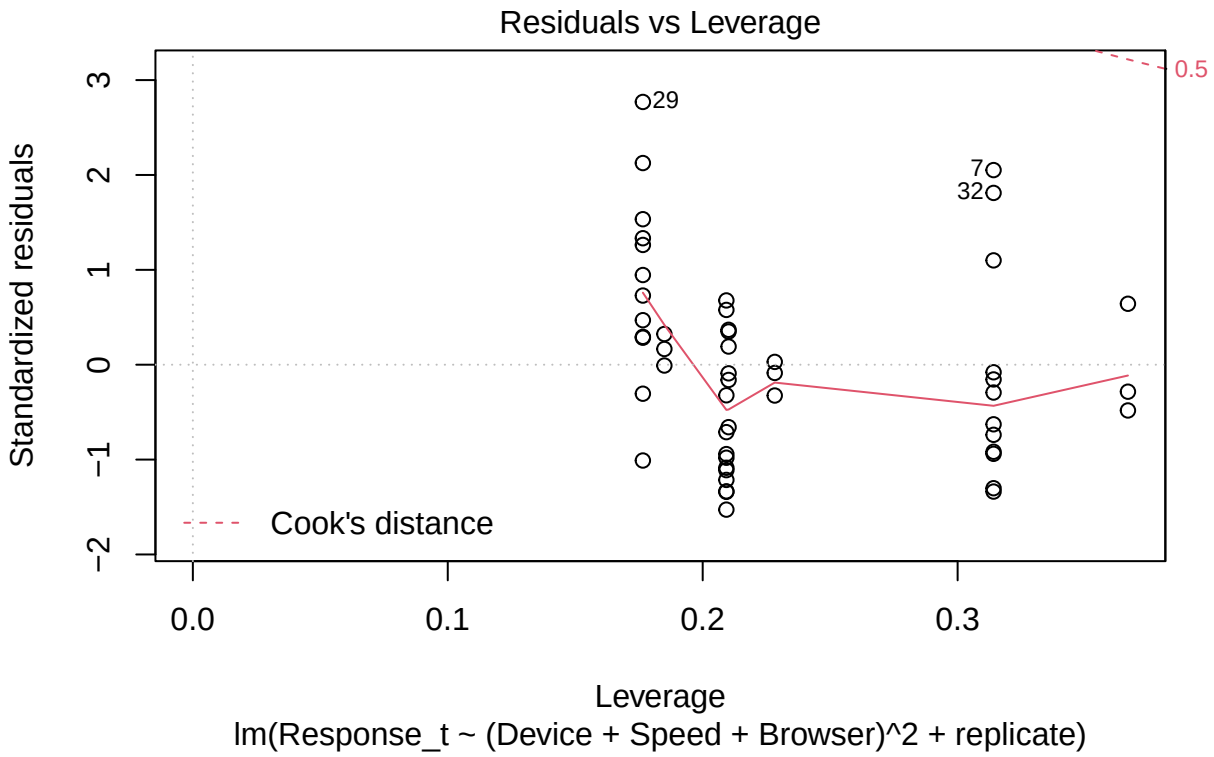



```
# Residual vs Fitted
plot(m1)
```

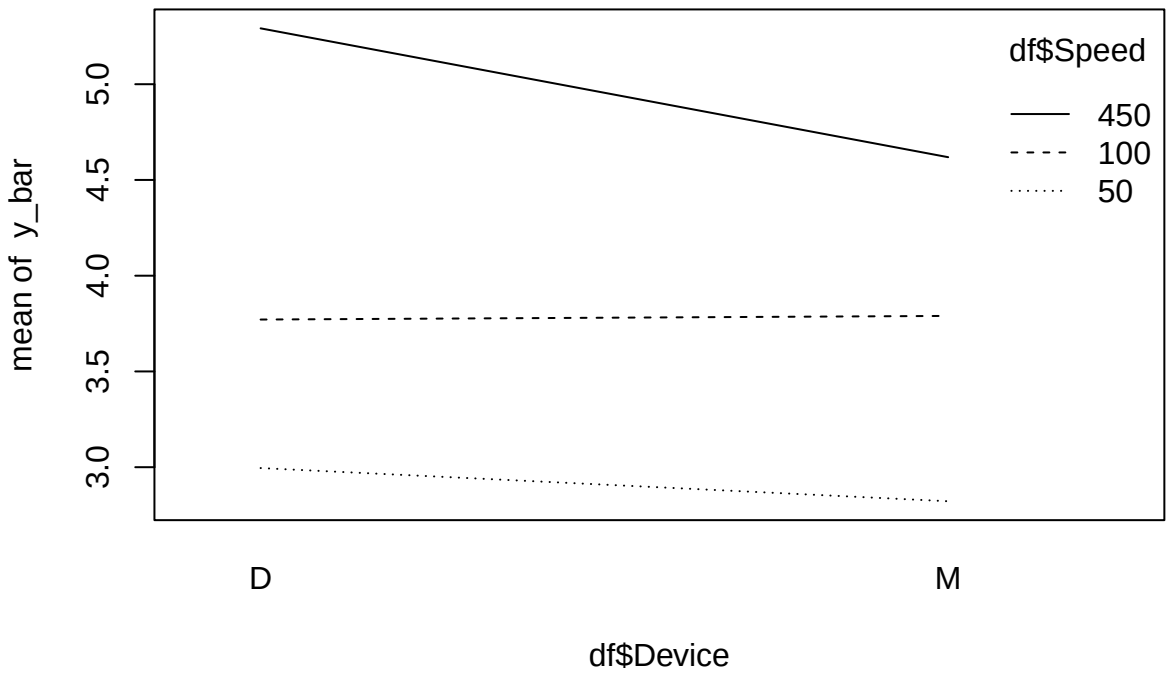

Residuals vs Fitted
29
40
7
Residuals
0.5
0.0
−0.5
1.0
1.5
2.0
Fitted values
lm(Response_t ~ (Device + Speed + Browser)^2 + replicate)


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Response_t ~ (Device + Speed + Browser)^2 + replicate)
29
7
40
−2
−1
0
1
2
−1
0
1
2
3

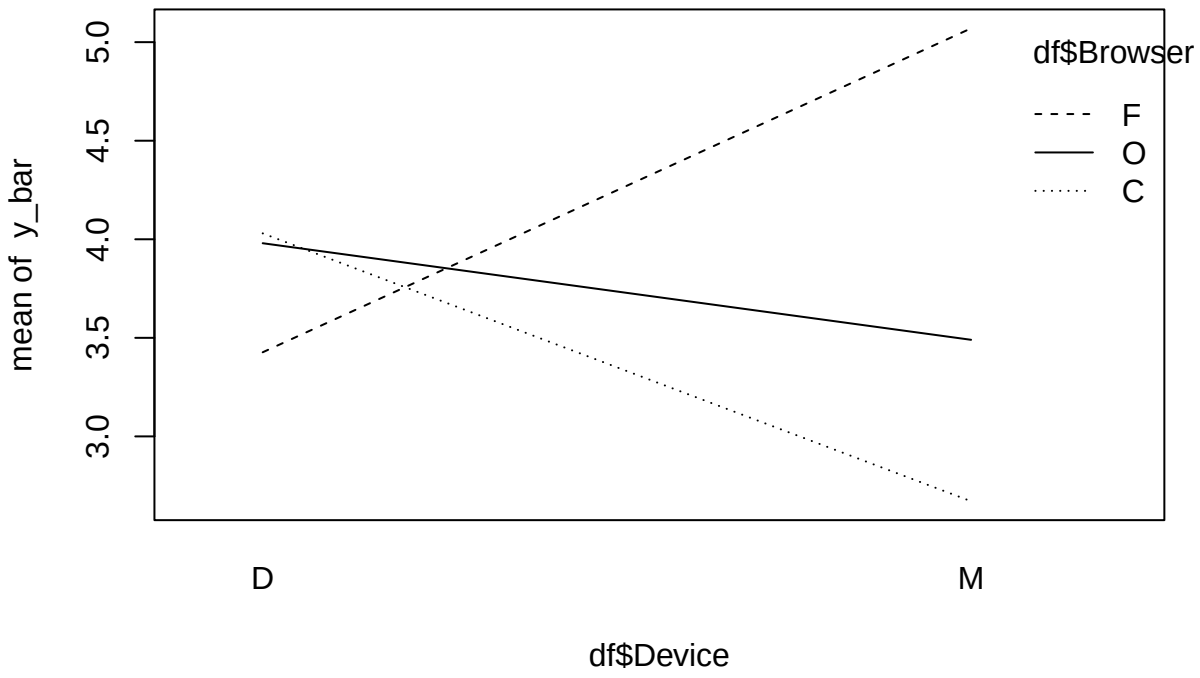

Scale−Location
29
40
7
√|Standardized residuals|
0.0
0.5
1.0
1.5
1.0
1.5
2.0
Fitted values
lm(Response_t ~ (Device + Speed + Browser)^2 + replicate)



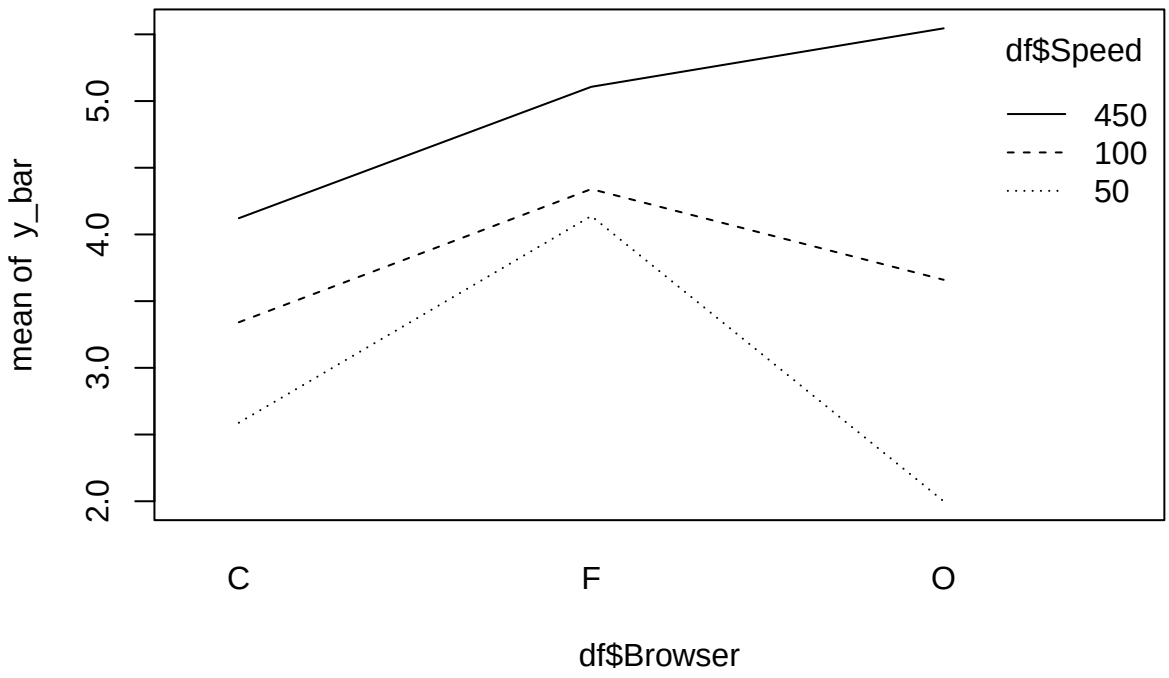

```
# Interaction plots
y_bar = (df$Replicate.1 + df$Replicate.2 + df$Replicate.3)/3
interaction.plot(df$Device, df$Speed, y_bar)
```




```
interaction.plot(df$Device, df$Browser, y_bar)
```




```
interaction.plot(df$Browser, df$Speed, y_bar)
```




```
# Tukey doesn't work with numeric factor
#plot(TukeyHSD(aov(m1)))

#TukeyHSD(aov(m1))
```