Final Project R Code

Wenxiao Yang

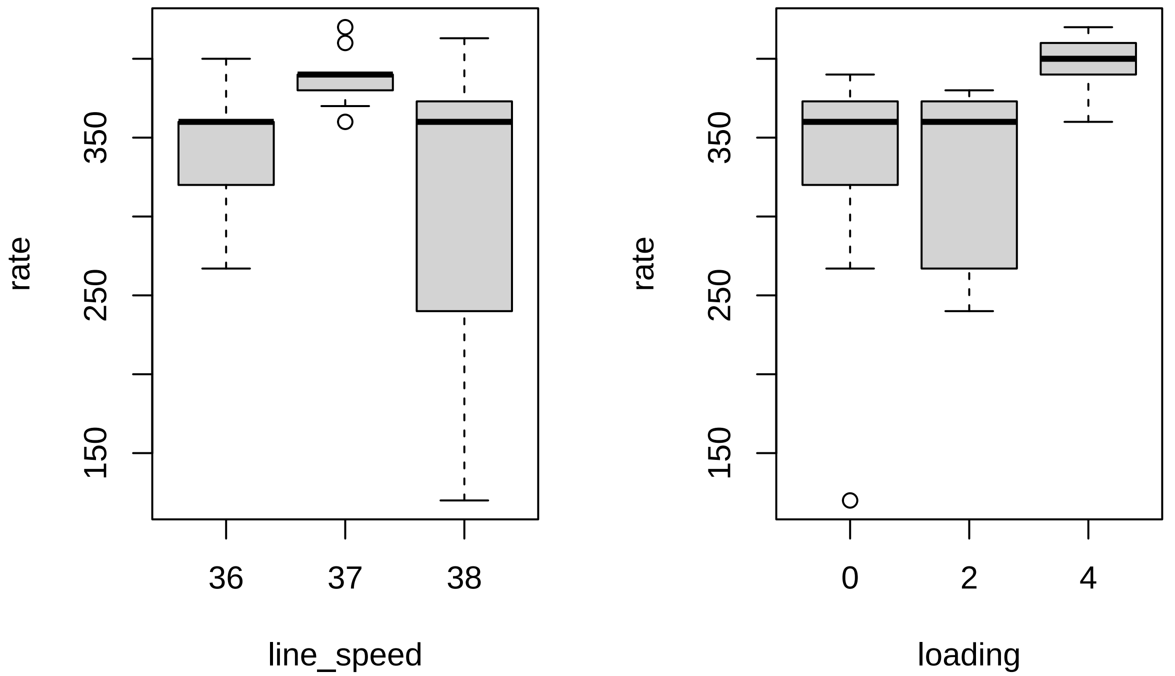
12/7/2021

R Code

```
bubble.data=read.csv("bubblewrap.csv",header=TRUE)
new.data=transform(bubble.data,line_speed=as.factor(line_speed),loading=as.factor(loading))
new.data
```

```
##    replication run_order line_speed loading rate
## 1            1         6         38       2  240
## 2            1         8         37       4  390
## 3            1         1         36       0  360
## 4            1         9         38       4  400
## 5            1         3         38       0  320
## 6            1         7         36       4  400
## 7            1         2         37       0  360
## 8            1         4         36       2  320
## 9            1         5         37       2  380
## 10           2         6         38       2  240
## 11           2         5         37       2  380
## 12           2         3         38       0  120
## 13           2         7         36       4  360
## 14           2         1         36       0  360
## 15           2         4         36       2  360
## 16           2         9         38       4  360
## 17           2         2         37       0  390
## 18           2         8         37       4  420
## 19           3         2         37       0  390
## 20           3         1         36       0  267
## 21           3         3         38       0  373
## 22           3         4         36       2  267
## 23           3         8         37       4  410
## 24           3         7         36       4  397
## 25           3         5         37       2  370
## 26           3         6         38       2  373
## 27           3         9         38       4  413
```

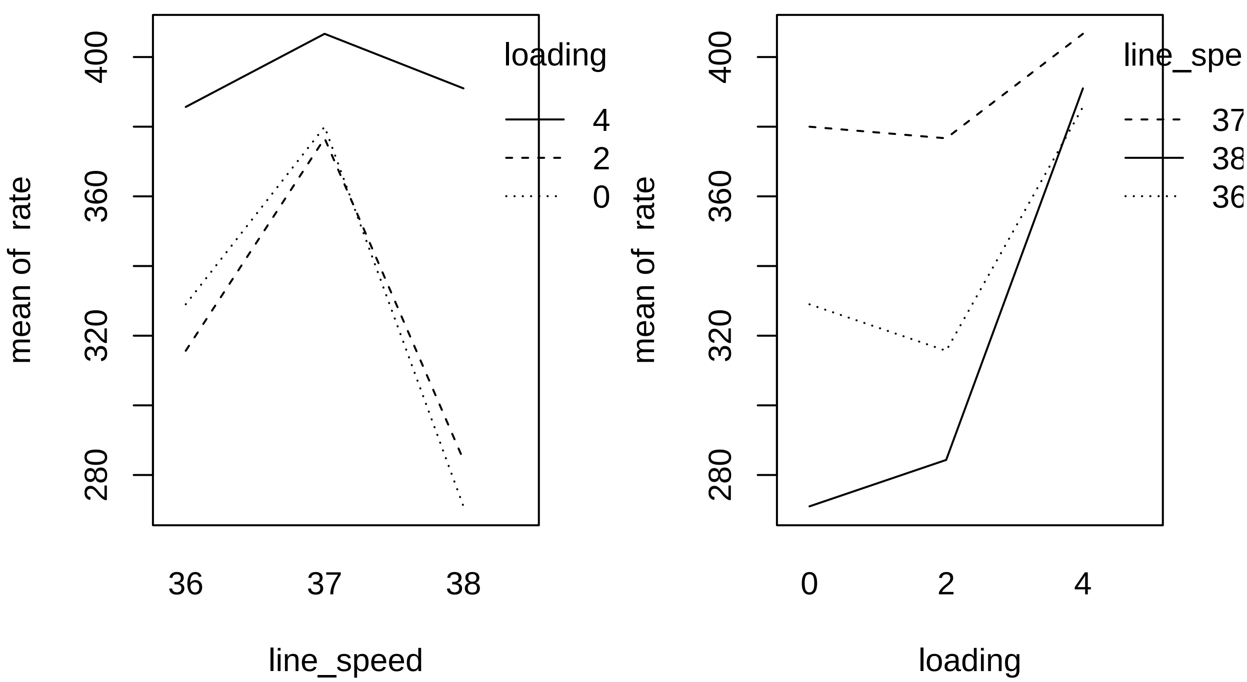
```
par(mfrow=c(1,2))
plot(rate~line_speed, new.data)
plot(rate~loading, new.data)
```



The factor effects model is as follows:

$$Y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \varepsilon_{ijk}$$

```
par(mfrow=c(1,2))
with(new.data, interaction.plot(line_speed,loading,rate))
with(new.data, interaction.plot(loading,line_speed,rate))
```



Interactions are present.

F-test

```
model1=lm(rate~line_speed*loading,new.data)
summary(model1)
```

```
##
## Call:
## lm(formula = rate ~ line_speed * loading, data = new.data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -151.000  -22.833    4.333   18.167  102.000
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)             329.00      33.66   9.775 1.27e-08 ***
## line_speed37             51.00      47.60   1.071    0.298
## line_speed38            -58.00      47.60  -1.219    0.239
## loading2                -13.33      47.60  -0.280    0.783
## loading4                 56.67      47.60   1.191    0.249
## line_speed37:loading2    10.00      67.31   0.149    0.884
## line_speed38:loading2    26.67      67.31   0.396    0.697
## line_speed37:loading4   -30.00      67.31  -0.446    0.661
## line_speed38:loading4    63.33      67.31   0.941    0.359
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 58.29 on 18 degrees of freedom
## Multiple R-squared:  0.4944, Adjusted R-squared:  0.2697
## F-statistic:   2.2 on 8 and 18 DF,  p-value: 0.07858
```

```
anova(model1)
```

```
## Analysis of Variance Table
##
## Response: rate
##                    Df Sum Sq Mean Sq F value  Pr(>F)
## line_speed          2  23945 11972.3  3.5230 0.05114 .
## loading             2  28022 14011.1  4.1230 0.03357 *
## line_speed:loading  4   7844  1961.1  0.5771 0.68293
## Residuals          18  61169  3398.3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since the p-values are large, we conclude that the interaction terms are not statistically significant. So, we can remove it from the model.

```
model2=lm(rate~line_speed+loading,new.data)
summary(model2)
```

```
##
## Call:
## lm(formula = rate ~ line_speed + loading, data = new.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

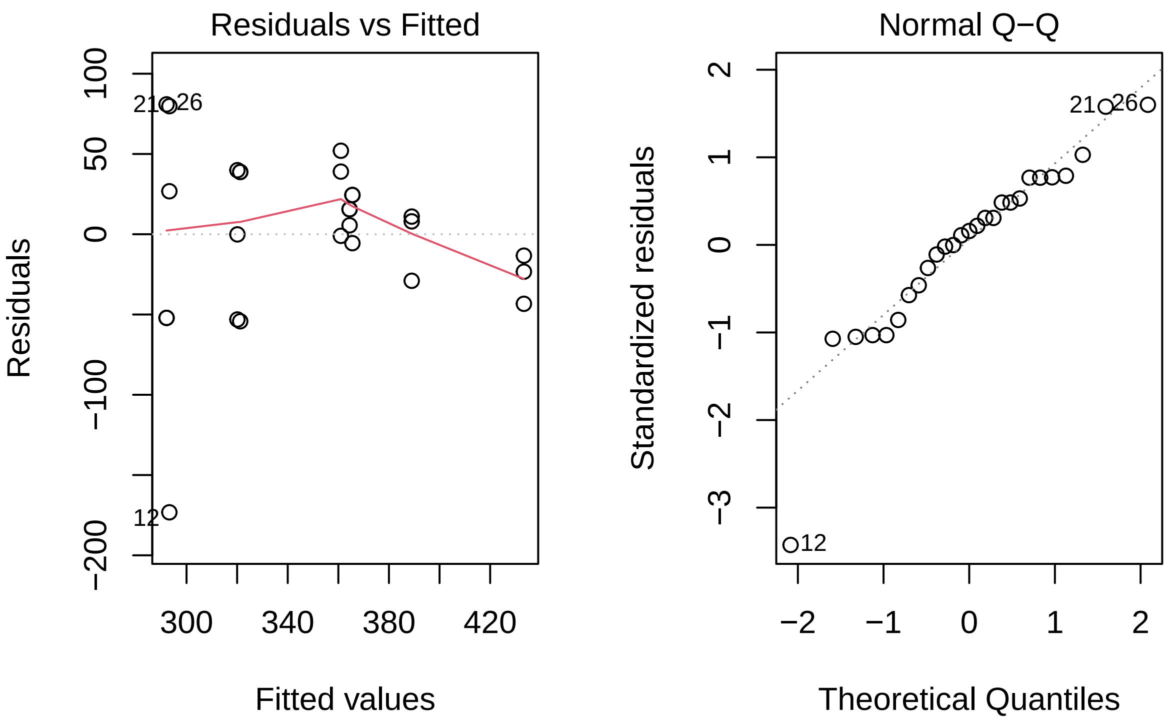
```
## -173.22  -26.17    8.00   32.78   80.89
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   321.222     24.102  13.327 5.17e-12 ***
## line_speed37   44.333     26.403   1.679   0.1073
## line_speed38  -28.000     26.403  -1.060   0.3004
## loading2       -1.111     26.403  -0.042   0.9668
## loading4       67.778     26.403   2.567   0.0176 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 56.01 on 22 degrees of freedom
## Multiple R-squared:  0.4295, Adjusted R-squared:  0.3258
## F-statistic: 4.141 on 4 and 22 DF,  p-value: 0.01192
```

```r
anova(model2)
```

```
## Analysis of Variance Table
##
## Response: rate
##            Df Sum Sq Mean Sq F value  Pr(>F)
## line_speed  2  23945   11972  3.8165 0.03777 *
## loading     2  28022   14011  4.4664 0.02355 *
## Residuals  22  69014    3137
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Check Assumptions for Model 2

```r
par(mfrow=c(1,2))
plot(model2,which=1:2)
```



###

Check Homoscedasticity of Model 2

```
library(lmtest)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
bptest(model2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model2
## BP = 7.3817, df = 4, p-value = 0.117
```

Homoscedasticity holds.

Check Normality of Model 2

```
ks.test(residuals(model2), y=pnorm)
```

```
## Warning in ks.test(residuals(model2), y = pnorm): ties should not be present for
## the Kolmogorov-Smirnov test

##
##  One-sample Kolmogorov-Smirnov test
##
## data:  residuals(model2)
## D = 0.55556, p-value = 1.156e-07
```

```
## alternative hypothesis: two-sided
```

Normality doesn't hold. Try Box-cox transformation.

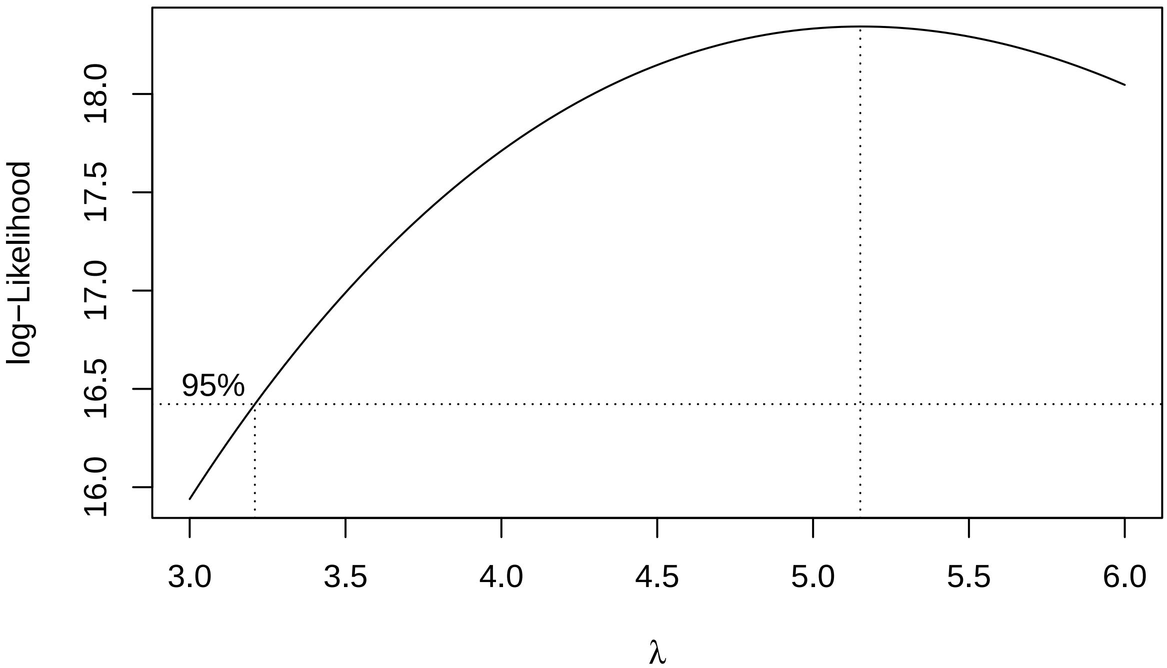
Box-cox transformation

```
library(MASS)
```

```
min(new.data$rate)
```

```
## [1] 120
```

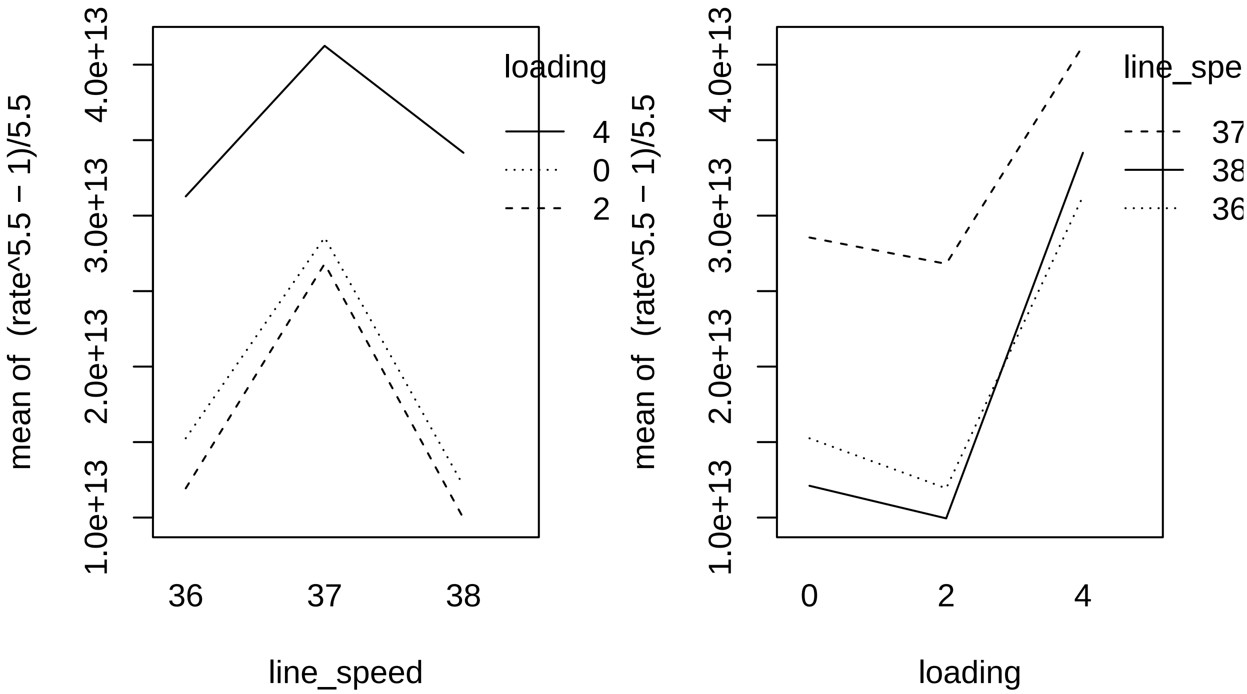
```
model.transformation=boxcox(model2,lambda=seq(3, 6, by=0.1))
```




```
model.transformation$x[model.transformation$y==max(model.transformation$y)]
```

```
## [1] 5.151515
```

```
par(mfrow=c(1,2))
with(new.data, interaction.plot(line_speed,loading,(rate^5.5-1)/5.5))
with(new.data, interaction.plot(loading,line_speed,(rate^5.5-1)/5.5))
```




```r
model.bc=lm((rate^5.5-1)/5.5~line_speed*loading,new.data)
summary(model.bc)
```

```
##
## Call:
## lm(formula = (rate^5.5 - 1)/5.5 ~ line_speed * loading, data = new.data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.331e+13 -7.704e+12  1.277e+12  5.609e+12  1.541e+13
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            1.525e+13  5.610e+12   2.718   0.0141 *
## line_speed37           1.330e+13  7.934e+12   1.676   0.1109
## line_speed38          -3.145e+12  7.934e+12  -0.396   0.6965
## loading2              -3.315e+12  7.934e+12  -0.418   0.6810
## loading4               1.602e+13  7.934e+12   2.020   0.0586 .
## line_speed37:loading2  1.571e+12  1.122e+13   0.140   0.8902
## line_speed38:loading2  1.156e+12  1.122e+13   0.103   0.9191
## line_speed37:loading4 -3.326e+12  1.122e+13  -0.296   0.7703
## line_speed38:loading4  6.035e+12  1.122e+13   0.538   0.5973
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.717e+12 on 18 degrees of freedom
## Multiple R-squared:  0.6476, Adjusted R-squared:  0.4909
## F-statistic: 4.134 on 8 and 18 DF,  p-value: 0.005915
```

```r
anova(model.bc)
```

```
## Analysis of Variance Table
##
```

```
## Response: (rate^5.5 - 1)/5.5
##                    Df     Sum Sq    Mean Sq F value    Pr(>F)
## line_speed          2 1.0305e+27 5.1525e+26  5.4567 0.0140461 *
## loading             2 1.9983e+27 9.9915e+26 10.5814 0.0009154 ***
## line_speed:loading  4 9.4140e+25 2.3535e+25  0.2492 0.9063497
## Residuals          18 1.6996e+27 9.4425e+25
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can find the interaction term is still insignificant.

```r
model3=lm((rate^5.5-1)/5.5~line_speed+loading,new.data)
summary(model3)
```

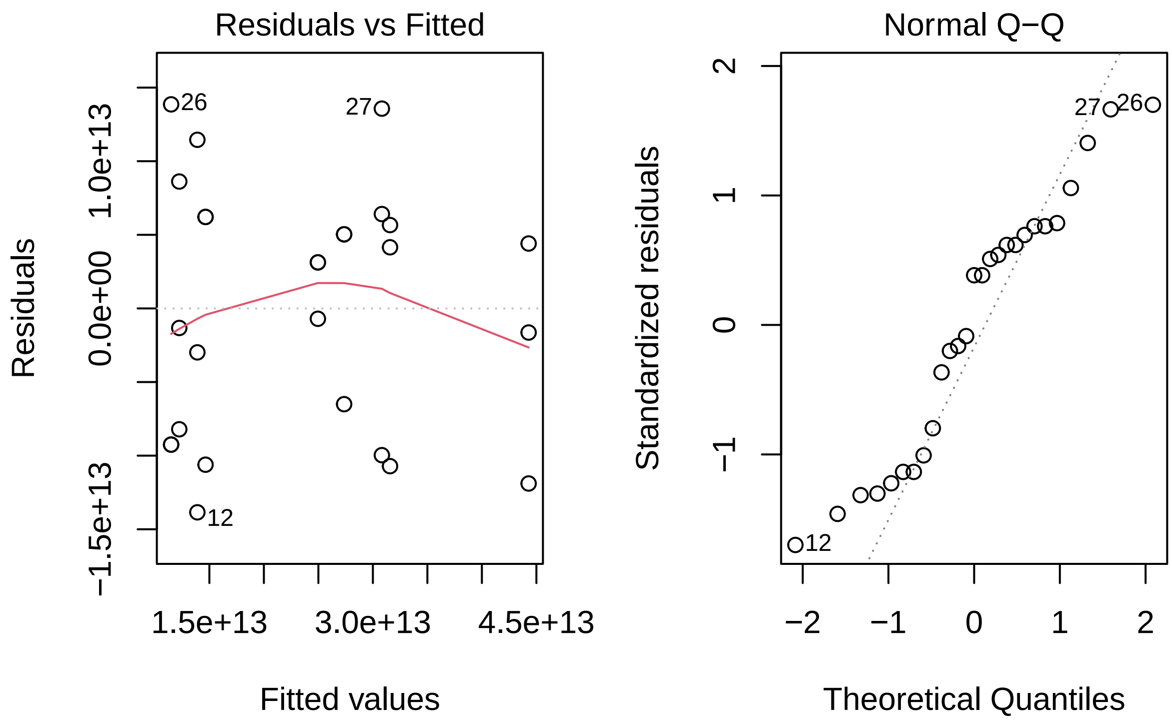
```
##
## Call:
## lm(formula = (rate^5.5 - 1)/5.5 ~ line_speed + loading, data = new.data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.385e+13 -8.729e+12  3.128e+12  5.938e+12  1.386e+13
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.465e+13  3.886e+12   3.769 0.001057 **
## line_speed37   1.272e+13  4.257e+12   2.987 0.006792 **
## line_speed38  -7.474e+11  4.257e+12  -0.176 0.862221
## loading2      -2.406e+12  4.257e+12  -0.565 0.577593
## loading4       1.693e+13  4.257e+12   3.977 0.000638 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.03e+12 on 22 degrees of freedom
## Multiple R-squared:  0.628,  Adjusted R-squared:  0.5604
## F-statistic: 9.287 on 4 and 22 DF,  p-value: 0.0001491
```

```r
anova(model3)
```

```
## Analysis of Variance Table
##
## Response: (rate^5.5 - 1)/5.5
##            Df     Sum Sq    Mean Sq F value    Pr(>F)
## line_speed  2 1.0305e+27 5.1525e+26  6.3193 0.0067840 **
## loading     2 1.9983e+27 9.9915e+26 12.2541 0.0002654 ***
## Residuals  22 1.7938e+27 8.1536e+25
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Check Assumptions for Model 3

```r
par(mfrow=c(1,2))
plot(model3,which=1:2)
```



Check Homoscedasticity of Model 3

```
bptest(model3)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model3
## BP = 9.3792, df = 4, p-value = 0.05229
```

Homoscedasticity holds.

Check Normality of Model 3

```
ks.test(residuals(model3), y=pnorm)
```

```
## Warning in ks.test(residuals(model3), y = pnorm): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  residuals(model3)
## D = 0.51852, p-value = 9.902e-07
## alternative hypothesis: two-sided
```

The Box-Cox transformation can't fix Normality problem. However, the model3's R^2 is higher, so we use model3.

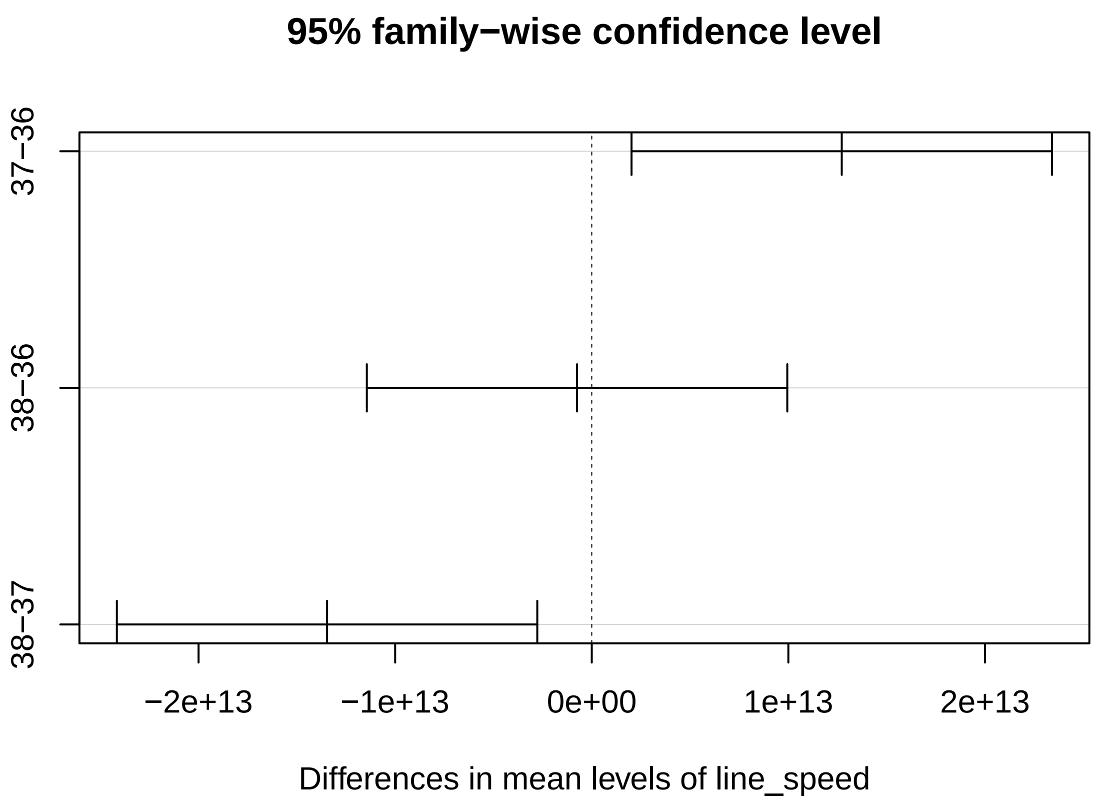
Tukey's Paired Comparison

```
TukeyHSD(aov((rate^5.5-1)/5.5~line_speed+loading,new.data), "line_speed")
```

```
##   Tukey multiple comparisons of means
```

```
##     95% family-wise confidence level
##
## Fit: aov(formula = (rate^5.5 - 1)/5.5 ~ line_speed + loading, data = new.data)
##
## $line_speed
##                diff           lwr           upr     p adj
## 37-36  1.271563e+13  2.022656e+12  2.340860e+13 0.0179125
## 38-36 -7.474332e+11 -1.144041e+13  9.945539e+12 0.9831589
## 38-37 -1.346306e+13 -2.415603e+13 -2.770089e+12 0.0120449
```

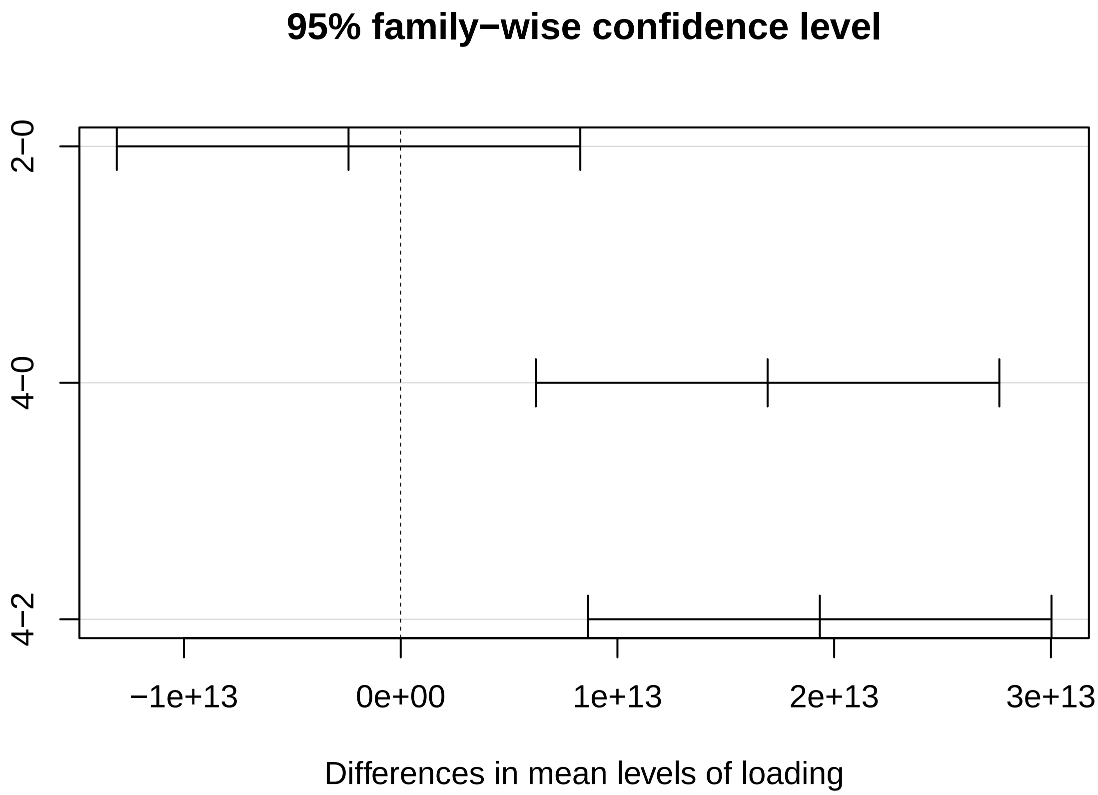
```
myCIs.line_speed = TukeyHSD(aov((rate^5.5-1)/5.5~line_speed+loading,new.data), "line_speed")
plot(myCIs.line_speed)
```




```
TukeyHSD(aov((rate^5.5-1)/5.5~line_speed+loading,new.data), "loading")
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = (rate^5.5 - 1)/5.5 ~ line_speed + loading, data = new.data)
##
## $loading
##              diff           lwr          upr     p adj
## 2-0 -2.406263e+12 -1.309924e+13 8.286709e+12 0.8397419
## 4-0  1.692712e+13  6.234150e+12 2.762009e+13 0.0017759
## 4-2  1.933339e+13  8.640413e+12 3.002636e+13 0.0004541
```

```
myCIs.loading = TukeyHSD(aov((rate^5.5-1)/5.5~line_speed+loading,new.data), "loading")
plot(myCIs.loading)
```


95% family−wise confidence level
2−0
4−0
4−2
−1e+13
0e+00
1e+13
2e+13
3e+13
Differences in mean levels of loading