5303hw09

Jin Yao

2019/11/4

5303 hw09

P11.5

```
library(cfcdae)
data("Fillings")
summary(Fillings)
```

```
##      alloy    method dentist    hardness     
##  1      :15   1:40   1:24    Min.   : 245.0  
##  2      :15   2:40   2:24    1st Qu.: 696.0  
##  3      :15   3:40   3:24    Median : 752.0  
##  4      :15          4:24    Mean   : 736.6  
##  5      :15          5:24    3rd Qu.: 813.0  
##  6      :15                  Max.   :1115.0  
##  (Other):30
```

```
head(Fillings)
```

```
##   alloy method dentist hardness
## 1     1      1       1      792
## 2     2      1       1      824
## 3     3      1       1      813
## 4     4      1       1      792
## 5     5      1       1      792
## 6     6      1       1      907
```

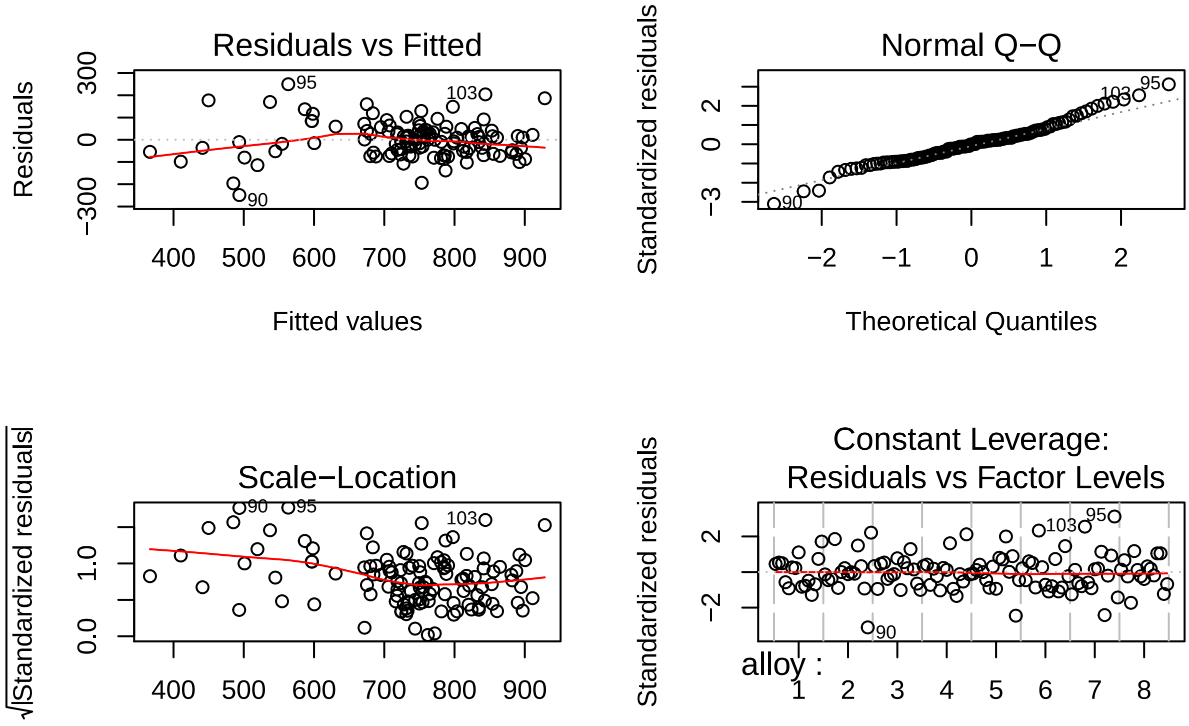
```
mod1 = lm(hardness~alloy*method+dentist+dentist:method,data=Fillings)
summary(mod1)
```

```
## 
## Call:
## lm(formula = hardness ~ alloy * method + dentist + dentist:method, 
##     data = Fillings)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -248.75  -55.59   -0.45   39.96  249.65 
## 
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)    
## (Intercept)       736.650      8.723  84.445  < 2e-16 ***
## alloy1             -9.183     23.080  -0.398 0.691720    
## alloy2            -21.583     23.080  -0.935 0.352391    
## alloy3            -11.717     23.080  -0.508 0.613027    
## alloy4            -27.383     23.080  -1.186 0.238789    
## alloy5            -48.383     23.080  -2.096 0.039062 *  
## alloy6             83.950     23.080   3.637 0.000474 ***
## alloy7             57.617     23.080   2.496 0.014501 *  
## method1            49.500     12.337   4.012 0.000130 ***
## method2            50.300     12.337   4.077 0.000103 ***
## dentist1           48.350     17.447   2.771 0.006874 ** 
## dentist2           43.017     17.447   2.466 0.015714 *  
## dentist3            4.558     17.447   0.261 0.794524    
## dentist4          -36.650     17.447  -2.101 0.038669 *  
## alloy1:method1    -53.567     32.640  -1.641 0.104510    
## alloy2:method1     37.833     32.640   1.159 0.249698    
## alloy3:method1    -20.433     32.640  -0.626 0.533000    
## alloy4:method1    -26.167     32.640  -0.802 0.425005    
## alloy5:method1     29.033     32.640   0.889 0.376276    
## alloy6:method1    -12.300     32.640  -0.377 0.707245    
## alloy7:method1     16.433     32.640   0.503 0.615951    
## alloy1:method2    -53.367     32.640  -1.635 0.105791    
## alloy2:method2    -20.167     32.640  -0.618 0.538346    
## alloy3:method2     -3.233     32.640  -0.099 0.921327    
## alloy4:method2     -5.567     32.640  -0.171 0.864990    
## alloy5:method2    -29.567     32.640  -0.906 0.367613    
## alloy6:method2     30.900     32.640   0.947 0.346513    
## alloy7:method2     10.833     32.640   0.332 0.740790    
## method1:dentist1  -16.125     24.674  -0.654 0.515198    
## method2:dentist1  -21.675     24.674  -0.878 0.382195    
## method1:dentist2   -6.167     24.674  -0.250 0.803252    
## method2:dentist2  -33.717     24.674  -1.367 0.175427    
## method1:dentist3  -52.583     24.674  -2.131 0.035999 *  
## method2:dentist3  -25.383     24.674  -1.029 0.306544    
## method1:dentist4   31.875     24.674   1.292 0.199947    
## method2:dentist4   35.325     24.674   1.432 0.155943    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 95.56 on 84 degrees of freedom
## Multiple R-squared:  0.6629, Adjusted R-squared:  0.5225 
## F-statistic: 4.721 on 35 and 84 DF,  p-value: 3.058e-09
```

```
anova(mod1)
```

```
## Analysis of Variance Table
## 
## Response: hardness
##                 Df Sum Sq Mean Sq F value    Pr(>F)    
## alloy            7 220338   31477  3.4469 0.0027252 ** 
## method           2 597615  298808 32.7216 3.103e-11 ***
## dentist          4 217576   54394  5.9566 0.0002866 ***
## alloy:method    14 209773   14984  1.6408 0.0848487 .  
## method:dentist   8 263441   32930  3.6061 0.0012030 ** 
## Residuals       84 767072    9132                      
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

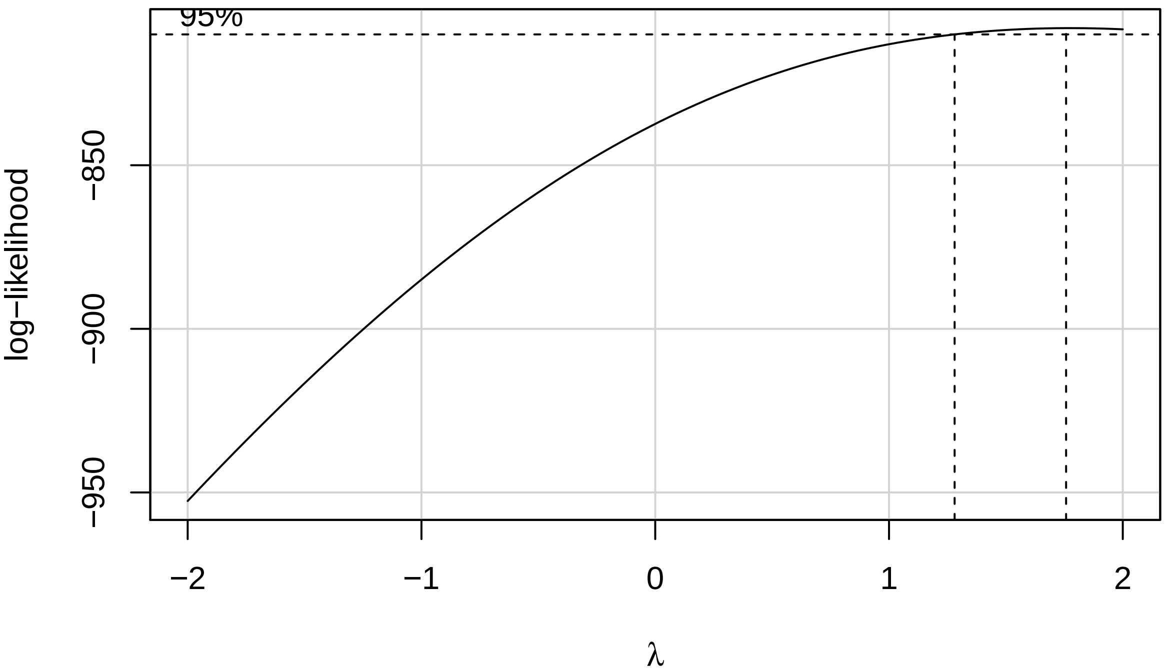
```
par(mfrow=c(2,2))
plot(mod1)
```



```r
par(mfrow=c(1,1))
library(car)
```

```
## Loading required package: carData
```

```r
boxCox(mod1)
```



```r
# dentist is random
library(lme4)
```

```
## Loading required package: Matrix
```

```
mod1lmer = lmer(hardness^2~alloy*method+(1|dentist)+ (1|dentist:method),data=Fillings)
Anova(mod1lmer,test='F')
```

```
## Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)
##
## Response: hardness^2
##                    F Df Df.res    Pr(>F)
## alloy         4.1573  7     84 0.0005683 ***
## method       11.7347  2      8 0.0041764 **
## alloy:method  1.6020 14     84 0.0954108 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod1lmer)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## hardness^2 ~ alloy * method + (1 | dentist) + (1 | dentist:method)
##    Data: Fillings
##
## REML criterion at convergence: 2618.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.99862 -0.61799  0.01225  0.41724  2.81865
##
## Random effects:
##  Groups         Name        Variance  Std.Dev.
##  dentist:method (Intercept) 3.174e+09  56342
##  dentist        (Intercept) 1.639e+09  40482
##  Residual                   1.754e+10 132441
## Number of obs: 120, groups:  dentist:method, 15; dentist, 5
##
## Fixed effects:
##                 Estimate Std. Error t value
## (Intercept)       561618      26183  21.450
## alloy1            -29388      31987  -0.919
## alloy2            -28380      31987  -0.887
## alloy3            -23231      31987  -0.726
## alloy4            -50425      31987  -1.576
## alloy5            -66493      31987  -2.079
## alloy6            127528      31987   3.987
## alloy7             90498      31987   2.829
## method1            63818      26751   2.386
## method2            65771      26751   2.459
## alloy1:method1    -68412      45237  -1.512
## alloy2:method1     53765      45237   1.189
## alloy3:method1    -31934      45237  -0.706
## alloy4:method1    -34121      45237  -0.754
## alloy5:method1     30178      45237   0.667
## alloy6:method1    -14925      45237  -0.330
## alloy7:method1     35291      45237   0.780
## alloy1:method2    -69681      45237  -1.540
## alloy2:method2   -43125      45237  -0.953
## alloy3:method2    -6281      45237  -0.139
## alloy4:method2    -2780      45237  -0.061
## alloy5:method2   -54590      45237  -1.207
## alloy6:method2    71610      45237   1.583
## alloy7:method2    15970      45237   0.353
```

```
##
## Correlation matrix not shown by default, as p = 24 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```
# the alloy and method are significant
library(RLRsim)
# take a look at the dentist
mod1.denonly.lmer = lmer(hardness^2~alloy*method+(1|dentist),data=Fillings)
mod1.noden.lmer = lmer(hardness^2~alloy*method+(1|dentist:method),data=Fillings)
exactRLRT(mod1.denonly.lmer,mod1lmer,mod1.noden.lmer)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.59659, p-value = 0.1674
```

```
# the p value is big, which indicates that the dentist is not significant.
# take a look at method
mod1.methodonly.lmer =lmer(hardness^2~alloy*method+(1|dentist:method),data=Fillings)
mod1.nomethod.lmer = lmer(hardness^2~alloy*method+(1|dentist),data=Fillings)
exactRLRT(mod1.methodonly.lmer,mod1lmer,mod1.nomethod.lmer)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 3.7477, p-value = 0.0214
```

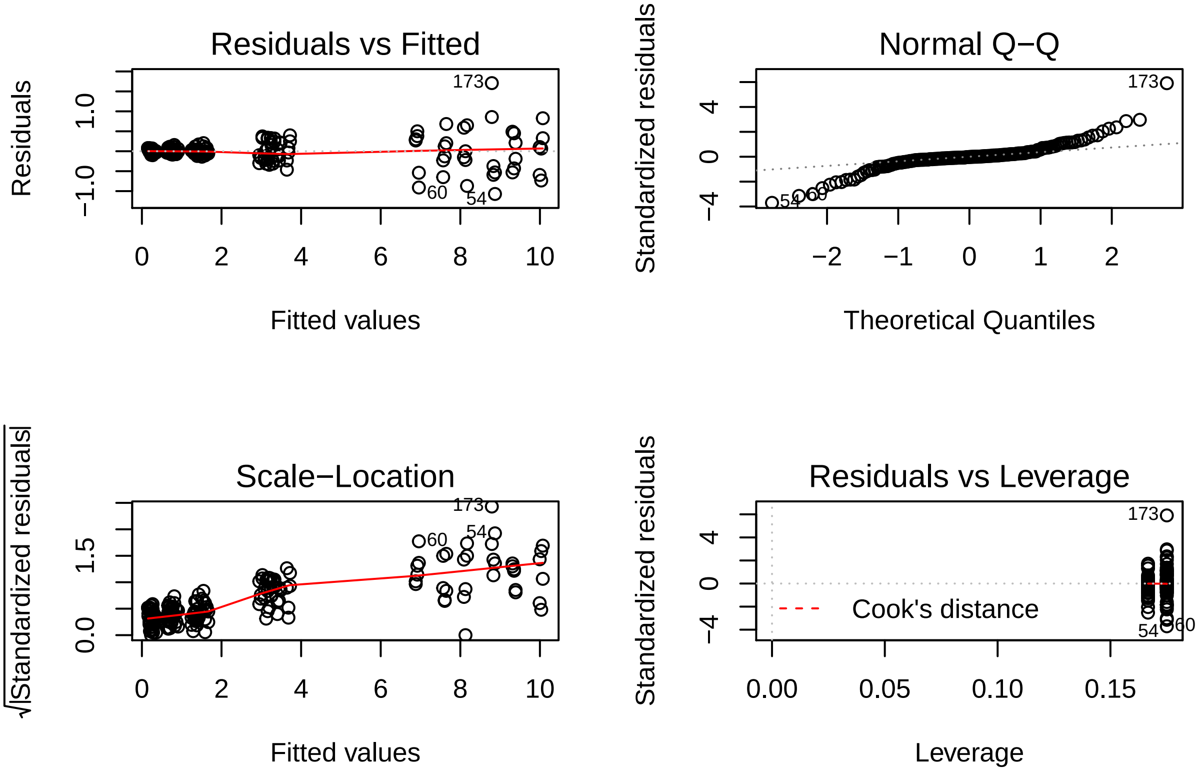
```
# the p value is tiny, which suggests that the interaction term is significant.
# so when taking the interaction into consideration, the summary table shows
# the sd of interaction is 56342.
```

P11.9

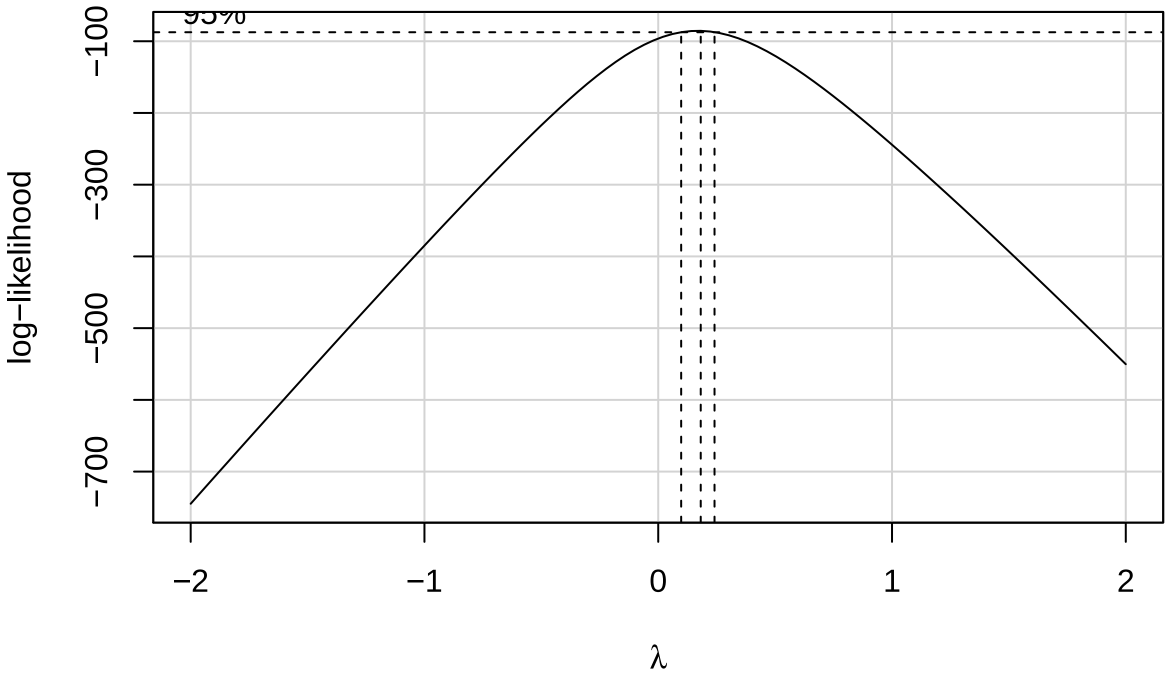
```
data(DNA)
head(DNA)
```

```
##   vol.z vol conc.z conc user od260
## 1    15  15     10   10    1  0.36
## 2    15  15     10   10    1  0.33
## 3    20  20     10   10    1  0.31
## 4    20  20     10   10    1  0.30
## 5     30   30       10     10      1   0.20
## 6     30   30       10     10      1   0.17
```

```
mod2 = lm(od260~vol*conc+user,data=DNA)
par(mfrow=c(2,2))
plot(mod2)
```



```
par(mfrow=c(1,1))
boxCox(mod2)
```



```
# maybe log
# analysts are random
mod2lmer = lmer(log(od260)~vol*conc+(1|user),data=DNA)
Anova(mod2lmer,test='F')
```

```
## Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)
##
## Response: log(od260)
##                    F Df Df.res    Pr(>F)
## vol          46.4711  5    148 < 2.2e-16 ***
## conc       7423.3407  4    148 < 2.2e-16 ***
## vol:conc      2.7456 20    148 0.0002666 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# all fixed terms including interaction seems signifiant
exactRLRT(mod2lmer)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.75017, p-value = 0.1188
```

```
# the p value is large, the random user seems not significant.
# as the question describes that the samll volumn may cause error, let make a comparison:
library(cfcdae)
linear.contrast(mod2lmer,vol,c(-1/3,-1/3,-1/3,1/3,1/3,1/3))
```

```
##    estimates         se   t-value      p-value   lower-ci   upper-ci
## 1 -0.1831092 0.01415734 -12.93387 4.430873e-26 -0.2110859 -0.1551326
```

```
# the p value is tiny, so there is difference between less than 40 and higher than 40.
summary(mod2lmer)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: log(od260) ~ vol * conc + (1 | user)
##    Data: DNA
##
## REML criterion at convergence: -188.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2009 -0.5315 -0.0609  0.6407  2.2136
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  user     (Intercept) 0.0001727 0.01314
##  Residual             0.0090194 0.09497
## Number of obs: 180, groups:  user, 3
##
## Fixed effects:
##              Estimate Std. Error  t value
## (Intercept)  0.387975   0.010376   37.391
## vol1         0.173024   0.015828   10.931
## vol2         0.107005   0.015828    6.760
## vol3        -0.005366   0.015828   -0.339
## vol4        -0.083533   0.015828   -5.277
## vol5        -0.055584   0.015828   -3.512
## conc1       -1.817711   0.014157 -128.394
## conc2       -0.688528   0.014157  -48.634
## conc3       -0.029813   0.014157   -2.106
## conc4        0.796591   0.014157   56.267
## vol1:conc1   0.113837   0.031657    3.596
## vol2:conc1   0.109536   0.031657    3.460
## vol3:conc1  -0.047194   0.031657   -1.491
## vol4:conc1  -0.155646   0.031657   -4.917
## vol5:conc1   0.016704   0.031657    0.528
## vol1:conc2  -0.015671   0.031657   -0.495
## vol2:conc2  -0.020274   0.031657   -0.640
## vol3:conc2  -0.014140   0.031657   -0.447
## vol4:conc2   0.021386   0.031657    0.676
## vol5:conc2   0.003065   0.031657    0.097
## vol1:conc3  -0.044070   0.031657   -1.392
## vol2:conc3  -0.032577   0.031657   -1.029
## vol3:conc3   0.001419   0.031657    0.045
## vol4:conc3   0.025618   0.031657    0.809
## vol5:conc3   0.019029   0.031657    0.601
## vol1:conc4  -0.058031   0.031657   -1.833
## vol2:conc4  -0.056662   0.031657   -1.790
## vol3:conc4   0.009006   0.031657    0.284
## vol4:conc4   0.058958   0.031657    1.862
## vol5:conc4   0.005474   0.031657    0.173

##
## Correlation matrix not shown by default, as p = 30 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```
# we know the numberic relation between the concentration and OD260, try to fit a numeric model
mod2num = lm(od260~vol.z*conc.z,data=DNA)
summary(mod2num)
```

```
##
## Call:
## lm(formula = od260 ~ vol.z * conc.z, data = DNA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4796 -0.3230 -0.1488  0.3853  1.4295
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.256e-01  9.488e-02   4.486 1.31e-05 ***
## vol.z         4.788e-04  1.859e-03   0.258    0.797
## conc.z        2.011e-02  4.248e-04  47.328  < 2e-16 ***
## vol.z:conc.z -7.088e-05  8.325e-06  -8.514 7.24e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.549 on 176 degrees of freedom
## Multiple R-squared:  0.9684, Adjusted R-squared:  0.9679
## F-statistic:  1800 on 3 and 176 DF,  p-value: < 2.2e-16
```

```
# while it seems that the volumn is not significant, but the interaction is still significant.
mod2numpol = lm(od260~(I(vol.z)+I(vol.z^2)+I(vol.z^3))*conc.z,data=DNA)
summary(mod2numpol)
```

```
##
## Call:
## lm(formula = od260 ~ (I(vol.z) + I(vol.z^2) + I(vol.z^3)) * conc.z,
##     data = DNA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1100 -0.3267 -0.1261  0.1707  1.5900
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         7.542e-01  5.701e-01   1.323    0.188
## I(vol.z)          -2.677e-02  4.766e-02  -0.562    0.575
## I(vol.z^2)         6.051e-04  1.091e-03   0.554    0.580
## I(vol.z^3)        -3.649e-06  6.747e-06  -0.541    0.589
## conc.z             2.275e-02  2.552e-03   8.914    7e-16 ***
## I(vol.z):conc.z   -1.880e-04  2.134e-04  -0.881    0.380
## I(vol.z^2):conc.z  5.680e-07  4.887e-06   0.116    0.908
## I(vol.z^3):conc.z  4.177e-09  3.021e-08   0.138    0.890
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5163 on 172 degrees of freedom
## Multiple R-squared:  0.9727, Adjusted R-squared:  0.9716
## F-statistic:   876 on 7 and 172 DF,  p-value: < 2.2e-16
```

```
# when I try a model of power to the 3, only concentration significant. So I don't think volume signifi
confint(mod2num)
```

```
##                     2.5 %        97.5 %
## (Intercept)   2.383813e-01  6.128852e-01
## vol.z        -3.190582e-03  4.148166e-03
## conc.z        1.926799e-02  2.094483e-02
## vol.z:conc.z -8.730578e-05 -5.444645e-05
```

```
# from the output, I can find that the ci for the concentraion is 1.926799e-02  2.094483e-02.
```

P11.10

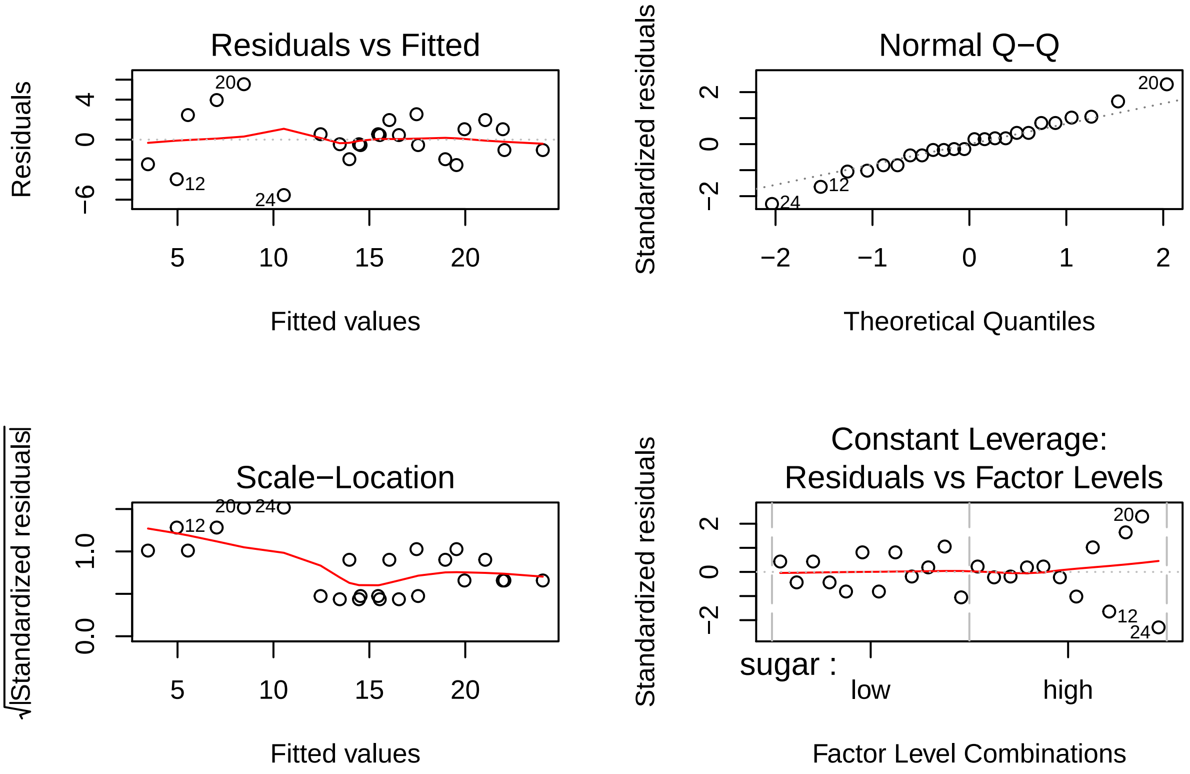
To make the power of testing A to be high, it is to we are more easy to reject the null hypothesis, which means the F value is larger. $F = MS(A)/MS(AB) = (sigma^2 + bn(sigmaA^2) + n(sigmaAB^2))/(sigma^2 + n(sigmaAB^2))$, so we need to increase the b.

E12.2

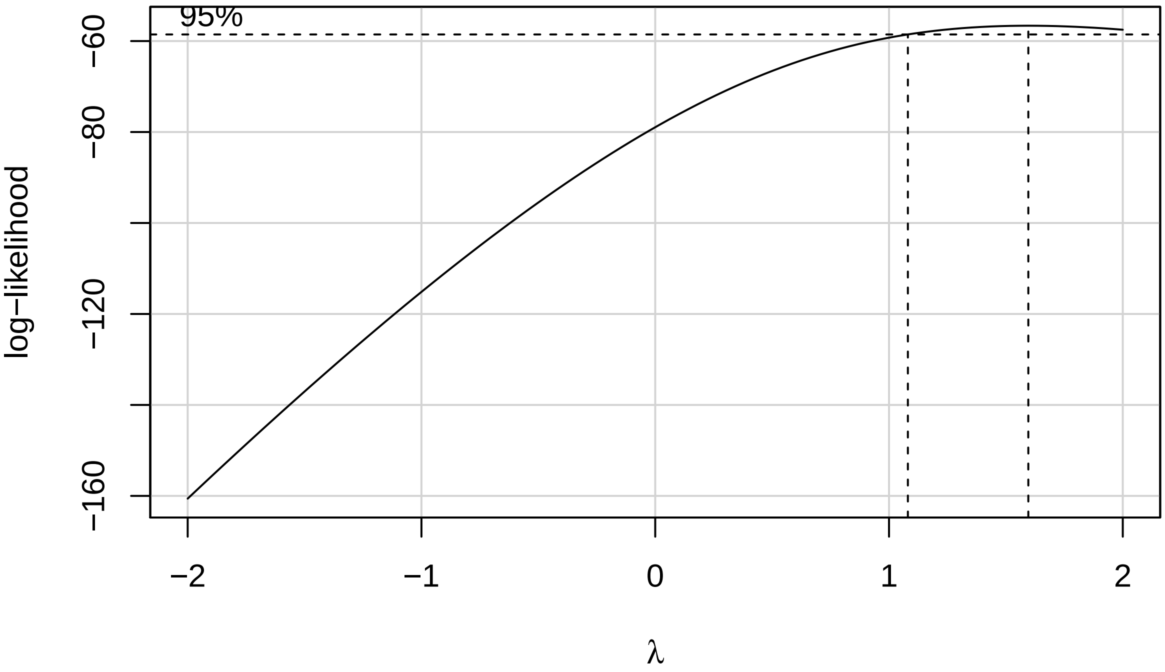
```
data("Graininess")
head(Graininess)
```

```
##   batch temp   pH sugar graininess
## 1     1    1  low   low         21
## 2     1    1 high   low         12
## 3     1    1  low  high         13
## 4     1    1 high  high          1
## 5     2    1  low   low         21
## 6     2    1 high   low         18
```

```
mod12.2 =lm(graininess~sugar*pH*temp+batch, data = Graininess)
par(mfrow=c(2,2))
plot(mod12.2)
```



```r
par(mfrow=c(1,1))
boxCox(mod12.2)
```



```r
# from the plot, I don't think we need a transformation.
mod12.2lmer = lmer(graininess~sugar*pH*temp+(1|batch), data = Graininess)
Anova(mod12.2lmer,test='F')
```

```
## Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)
##
```

```
## Response: graininess
##                      F Df Df.res    Pr(>F)
## sugar         30.9229  1     11 0.0001700 ***
## pH            20.5112  1     11 0.0008593 ***
## temp           1.0878  2     11 0.3706020
## sugar:pH       1.1864  1     11 0.2993369
## sugar:temp     0.3977  2     11 0.6811649
## pH:temp        0.6540  2     11 0.5390402
## sugar:pH:temp  0.2005  2     11 0.8212638
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# the table shows the fixed terms sugar and pH are significant
exactRLRT(mod12.2lmer)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.2905, p-value = 0.1751
```

```
# the p is very large, so the block seems not significant.
Graininess$comb = interaction(Graininess$pH,Graininess$sugar)
Graininess$comb = as.factor(Graininess$comb)
mod12.2lmer = lm(graininess~comb,data=Graininess)
compare.to.best(mod12.2lmer,comb,lowisbest=T)
```

```
##                         difference allowance
## * low.low - high.high   14.666667  4.241294
## * high.low - high.high   9.666667  4.241294
## * low.high - high.high   8.166667  4.241294
## best is high.high        0.000000        NA
```

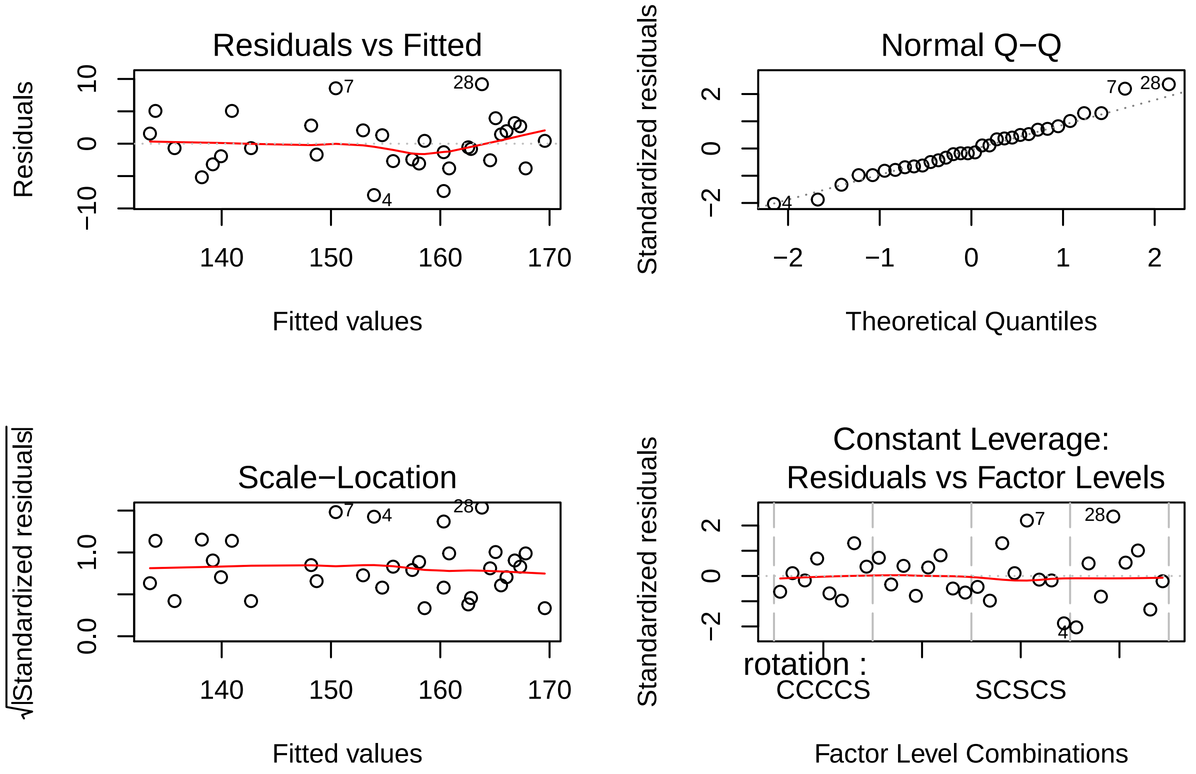
```
# the combination of high high leads to the least graininess.
```

P12.5

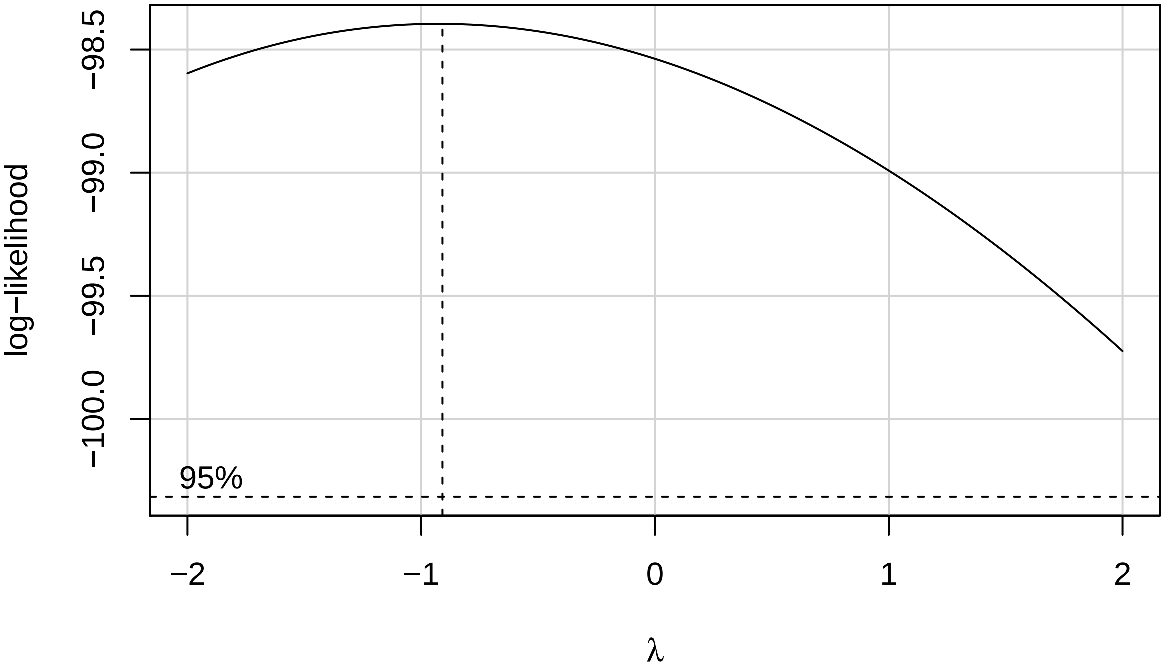
```
data("Rotations")
head(Rotations)
```

```
##   weight locYr rotation variety
## 1    155   W87    CCCCS     HOD
## 2    151   W87    CCCSS     HOD
## 3    147   W87    SCSCS     HOD
## 4    146   W87    SSSSS     HOD
## 5    153   W87    CCCCS     BSR
## 6    156   W87    CCCSS     BSR
```

```
mod12.5 =lm(weight~rotation*variety+locYr, data = Rotations)
par(mfrow=c(2,2))
plot(mod12.5)
```



```
# it looks pretty good
par(mfrow=c(1,1))
boxCox(mod12.5)
```



```
# use -1
mod12.5lmer = lmer(1/weight~rotation*variety+(1|locYr), data = Rotations)
Anova(mod12.5lmer,test='F')
```

```
## Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)
```

```
##
## Response: 1/weight
##                         F Df Df.res  Pr(>F)
## rotation          3.1127  3     21 0.04812 *
## variety           0.7455  1     21 0.39765
## rotation:variety 1.0626  3     21 0.38605
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# the fixed term rotation seems significant
exactRLRT(mod12.5lmer)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 38.281, p-value < 2.2e-16
```

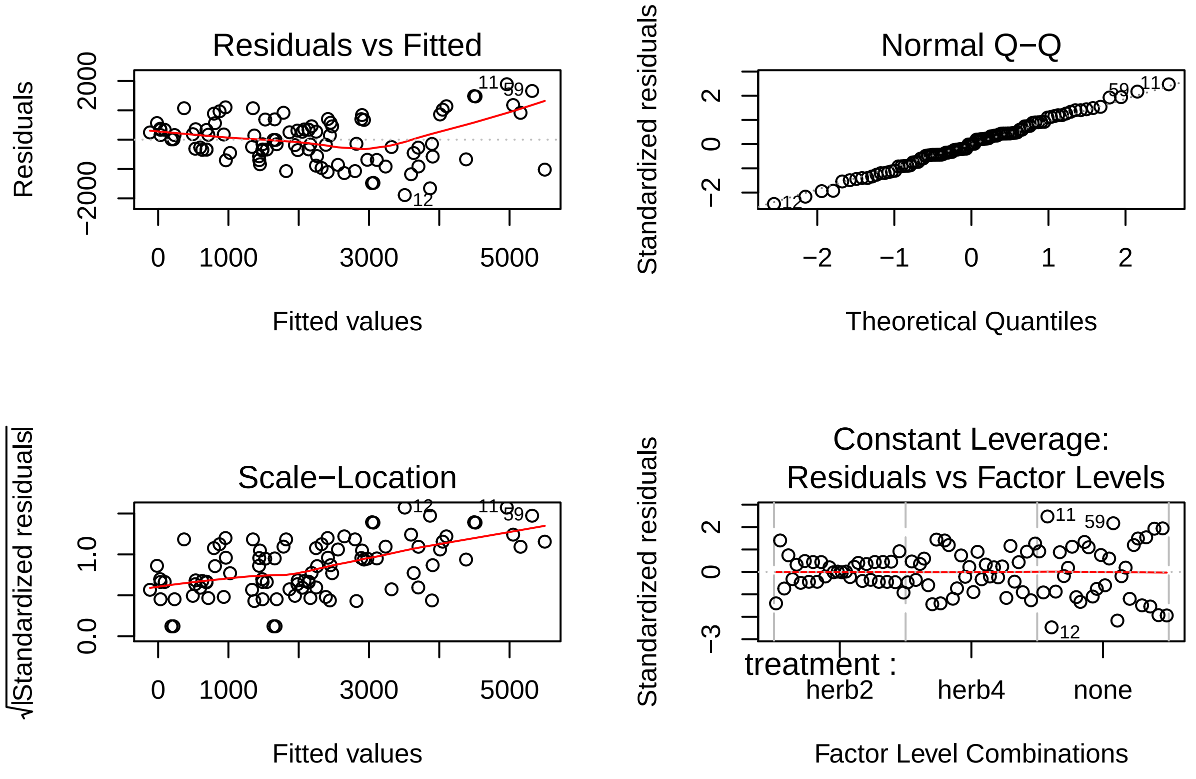
```
# the p value is small, so the random of locYr is significant
```

P12.6

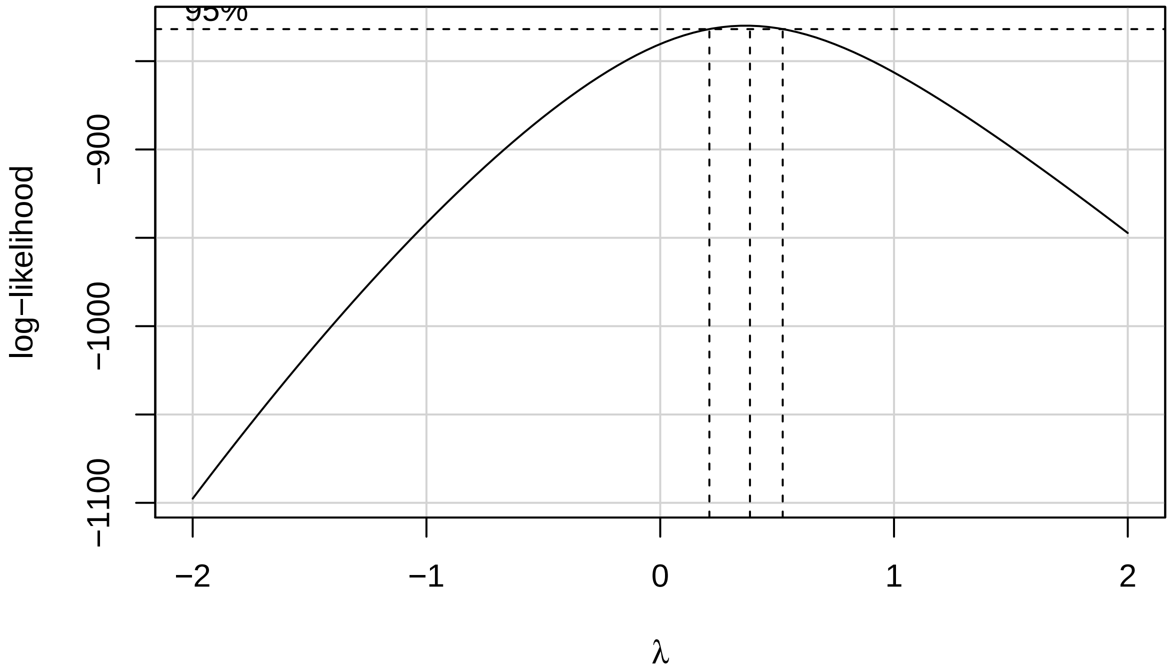
```
data("Herbicides" )
head(Herbicides)
```

```
##   location treatment variety biomass
## 1        R     herb2  Parker     750
## 2      StP     herb2  Parker    1440
## 3        R     herb4  Parker    1630
## 4      StP     herb4  Parker     890
## 5        R      none  Parker    3590
## 6      StP      none  Parker     740
```

```
mod12.6 = lm(biomass~treatment*variety+location, data = Herbicides)
par(mfrow=c(2,2))
plot(mod12.6)
```



```
par(mfrow=c(1,1))
boxCox(mod12.6)
```



```
# try 0.5
mod12.6lmer = lmer(biomass^(0.5)~treatment*variety+(1|location), data = Herbicides)
Anova(mod12.6lmer,test='F')
```

```
## Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)
##
```

```
## Response: biomass^(0.5)
##                         F Df Df.res    Pr(>F)
## treatment         46.2889  2     47 7.782e-12 ***
## variety            1.4284 15     47    0.1737
## treatment:variety  0.6861 30     47    0.8621
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# the fixed term treatment is significant
exactRLRT(mod12.6lmer)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 34.063, p-value < 2.2e-16
```

```
# the p value is tiny, the random location is also significant.
```