Stat5303HW8

Mingming Xu

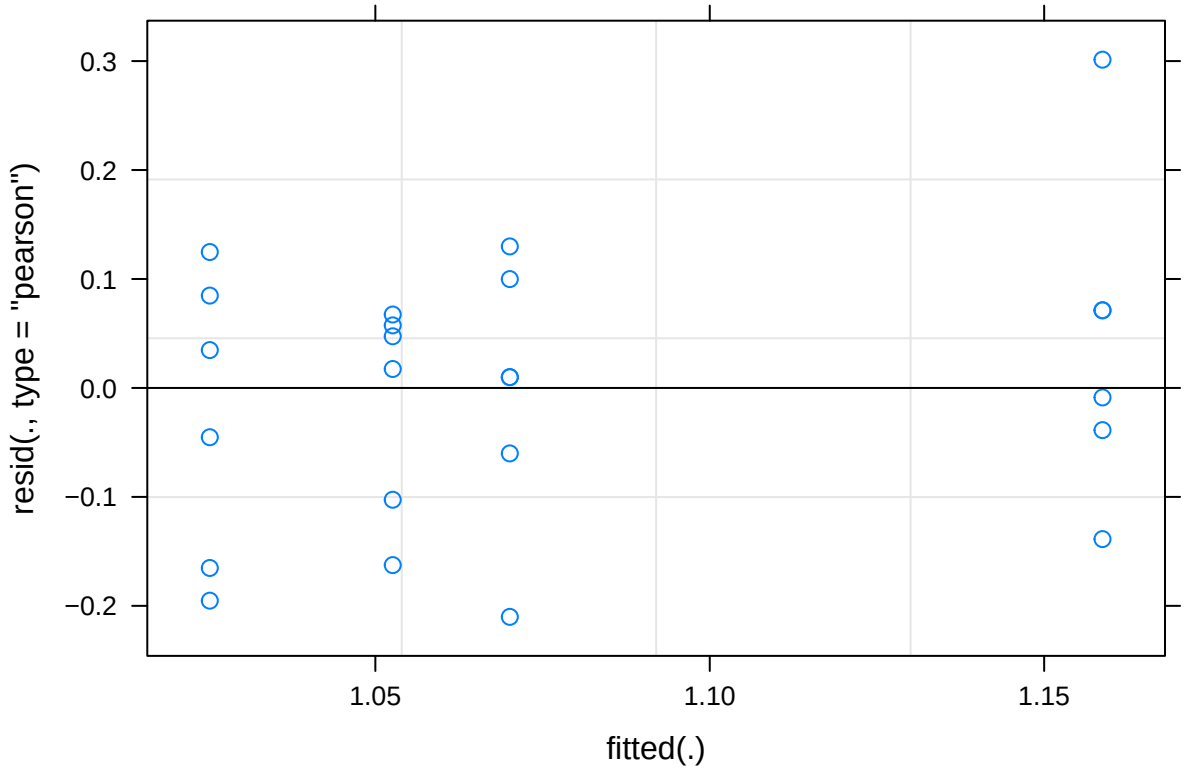
11/6/2020

E11.1

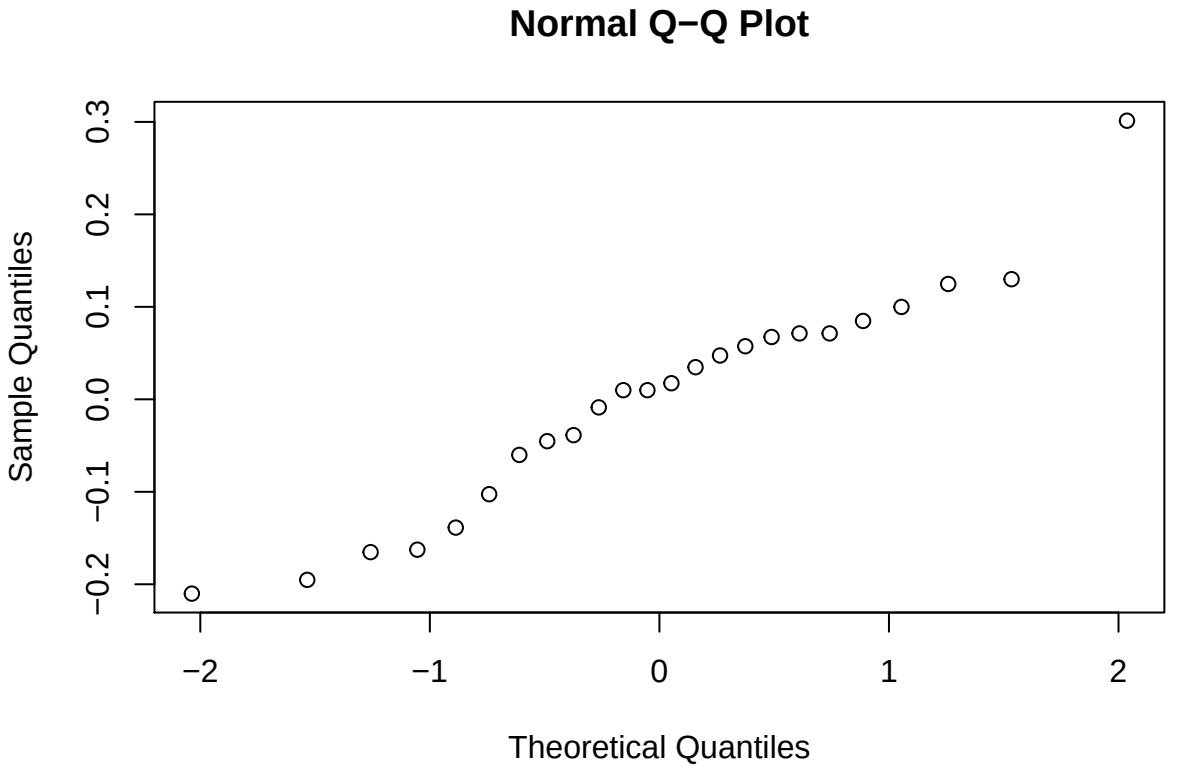
```
##lmer model for bull:
lmer.bull<-lmer(gain~1+(1|bull))
summary(lmer.bull)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: gain ~ 1 + (1 | bull)
##
## REML criterion at convergence: -23.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.6639 -0.5601  0.1081  0.5645  2.3859
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  bull     (Intercept) 0.005078 0.07126
##  Residual             0.015945 0.12627
## Number of obs: 24, groups:  bull, 4
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  1.07667    0.04397   24.48
```

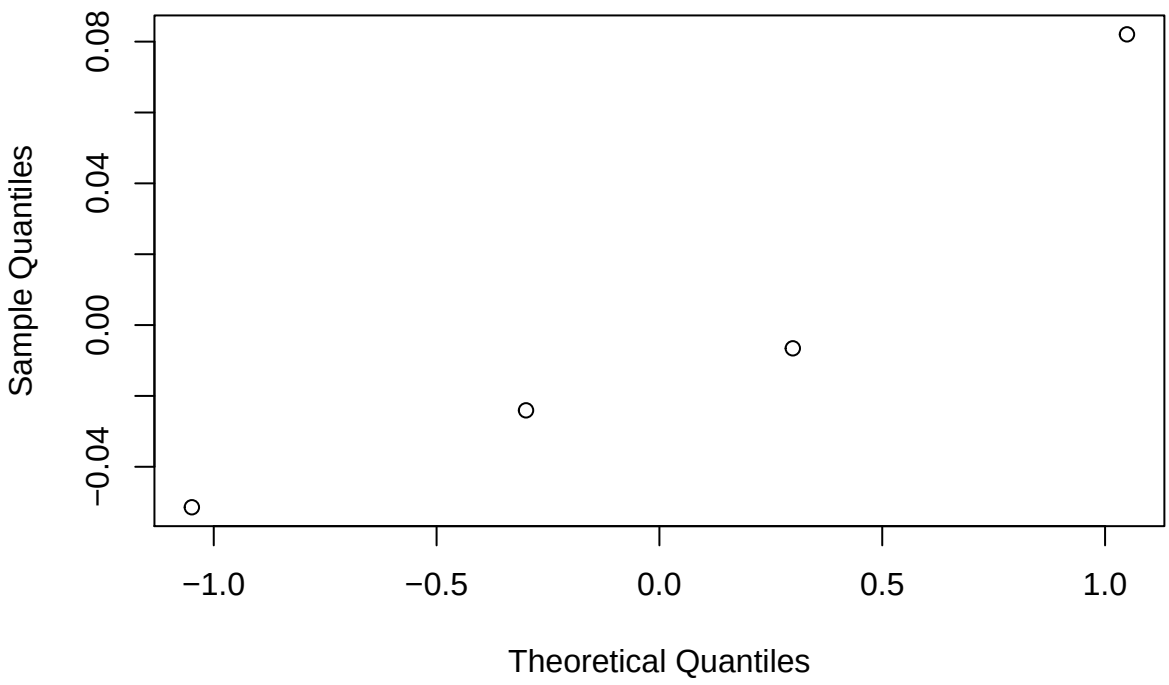
```
##Check assumptions:
plot(lmer.bull)
```




```
##Constant Variance looks like not bad
qqnorm(residuals(lmer.bull))
```




```
##Nomarlity if residuals is not too bad
qqnorm(ranef(lmer.bull)$"bull"[[1]],main="Bull effects")
```




```
##Since the factor bull only has 4 levels,
##it is difficult to say much about the normality of the random effects
##Tesing bull effects:
exactRLRT(lmer.bull)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 1.9128, p-value = 0.0594
```

(a) Looking at the result of exactRLRT(lmer.bull), because the p-value is 0.0634, larger than 0.05 we do not have enought evidence to reject the null hypothesis that there is no sire to sire variability in the response.

(b)

```
##Confidence intervals:
confint(lmer.bull,oldNames=FALSE,level = 0.9)
```

```
## Computing profile confidence intervals ...
```

```
##                             5 %      95 %
## sd_(Intercept)|bull 0.00000000 0.1477574
## sigma               0.09939488 0.1678653
## (Intercept)         1.00177802 1.1515553
```

```
##Confidence interval for Variance:
confint(lmer.bull,oldNames=FALSE,level = 0.9)^2
```

```
## Computing profile confidence intervals ...
```

```
##                             5 %       95 %
## sd_(Intercept)|bull 0.000000000 0.02183224
## sigma               0.009879343 0.02817875
## (Intercept)         1.003559203 1.32607953
```

```
##90% intervals for the error variance:
0.09939488^2; 0.1678653^2##
```

```
## [1] 0.009879342
```

```
## [1] 0.02817876
```

```
##the interval is [0.009879342, 0.02817876]
##90% intervals for the sire to sire variance:
0.1477574^2
```

```
## [1] 0.02183225
```

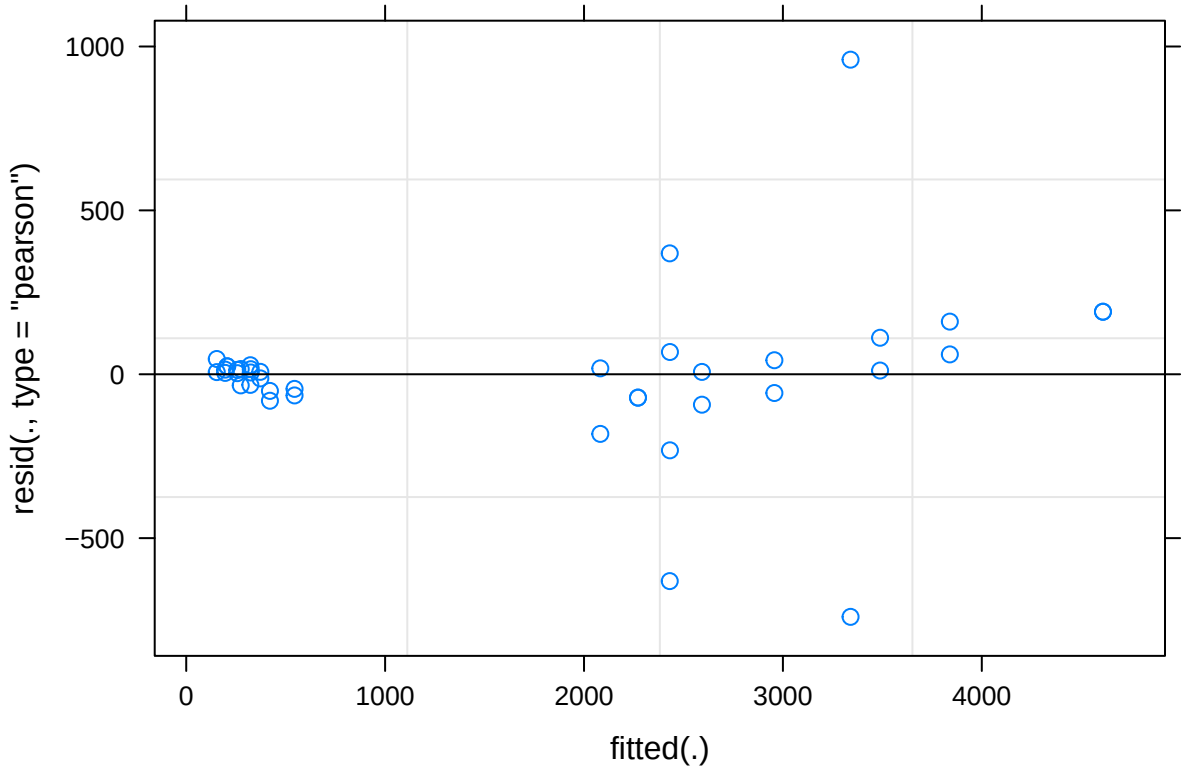
```
##the interval is [0.00000000, 0.02183225]
```

P11.2

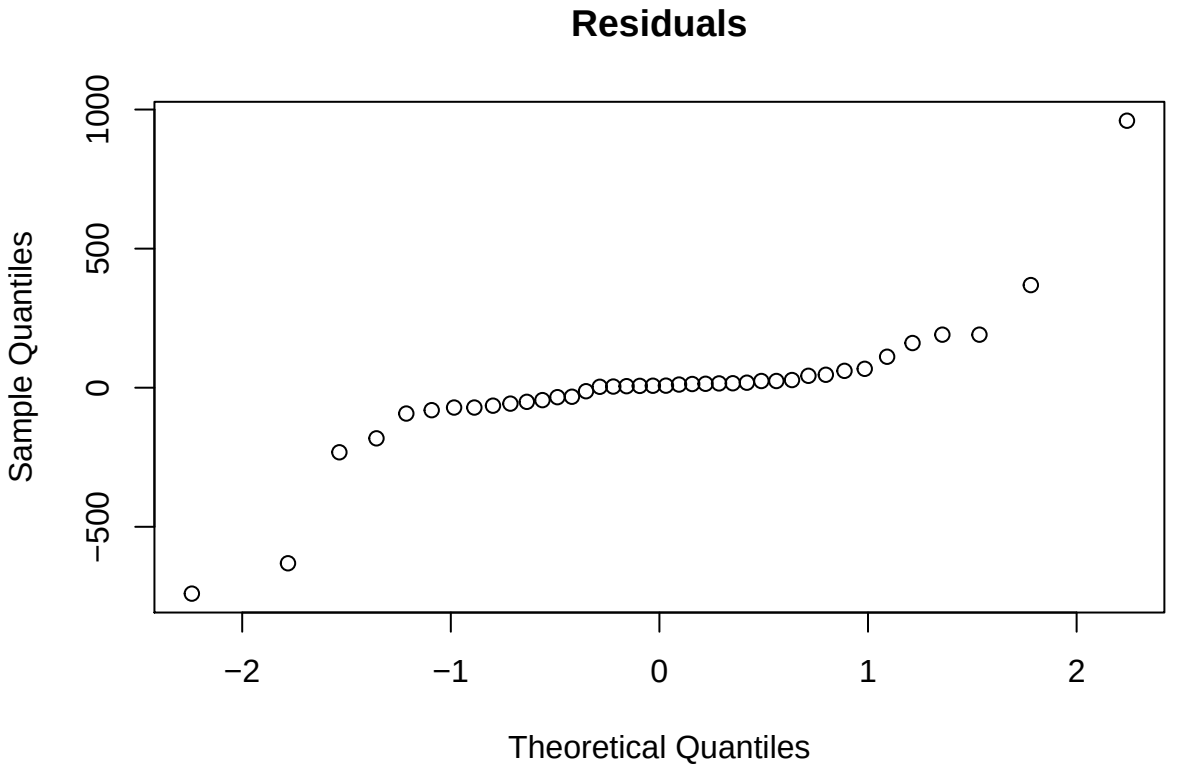
```
int.lmer<-lmer(count~1+(1|sample)+(1|lab)+(1|sample:lab))
summary(int.lmer)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: count ~ 1 + (1 | sample) + (1 | lab) + (1 | sample:lab)
##
## REML criterion at convergence: 613.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.32058 -0.16427  0.02271  0.09838  3.00904
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  sample:lab (Intercept)  266029   515.8
##  lab        (Intercept)  129667   360.1
##  sample     (Intercept) 2404245  1550.6
##  Residual                101743   319.0
## Number of obs: 40, groups:  sample:lab, 20; lab, 5; sample, 4
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   1655.7      801.8   2.065
```

```
##Check assumptions:
plot(int.lmer)
```

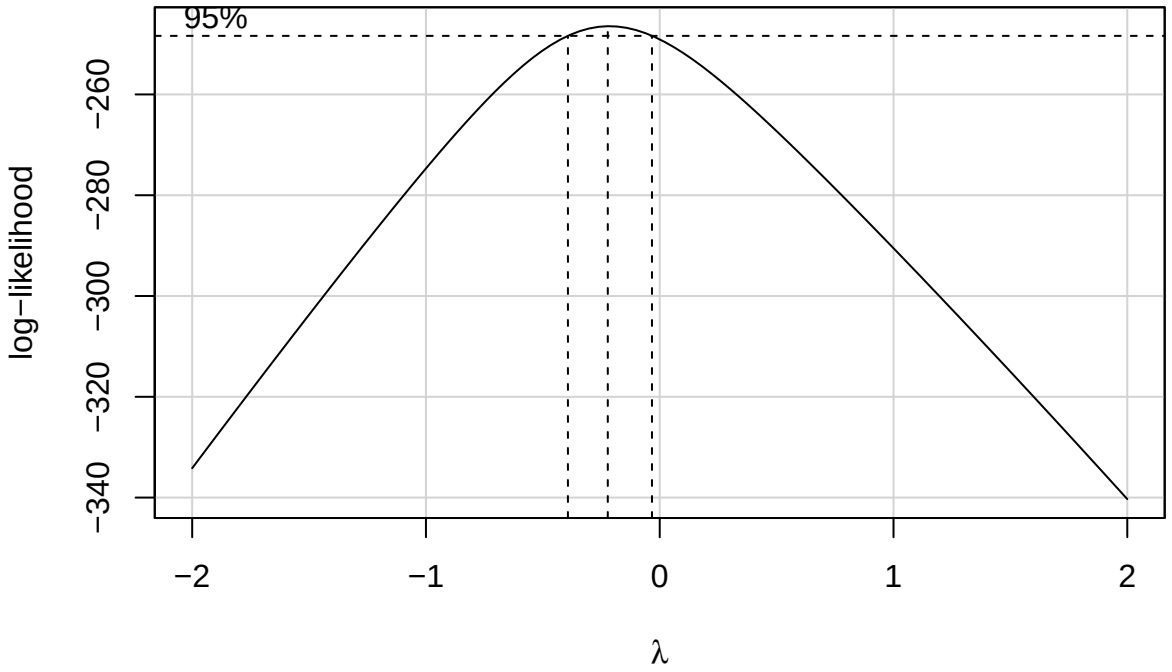



```
##Variance might be a little bit more gradually dispersed but it seems like to be good.
qqnorm(residuals(int.lmer),main = "Residuals")
```

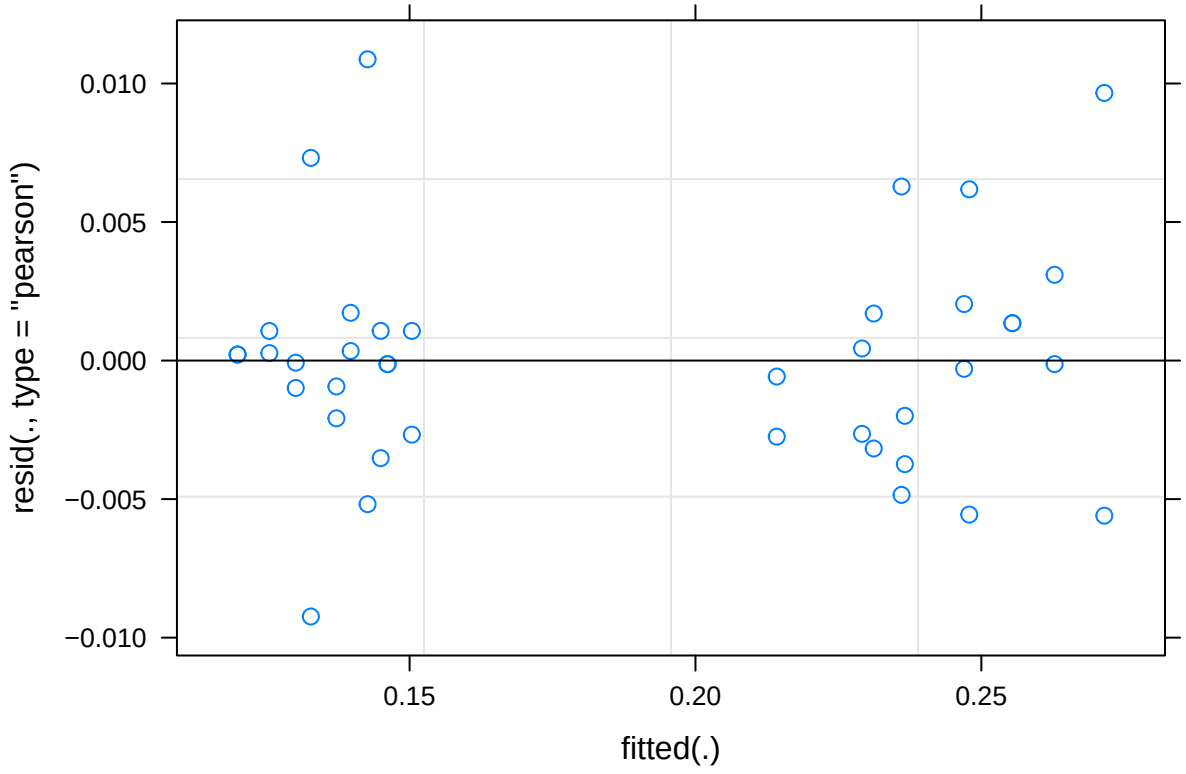



```
##The normarlity is not good
##Use transformation
```

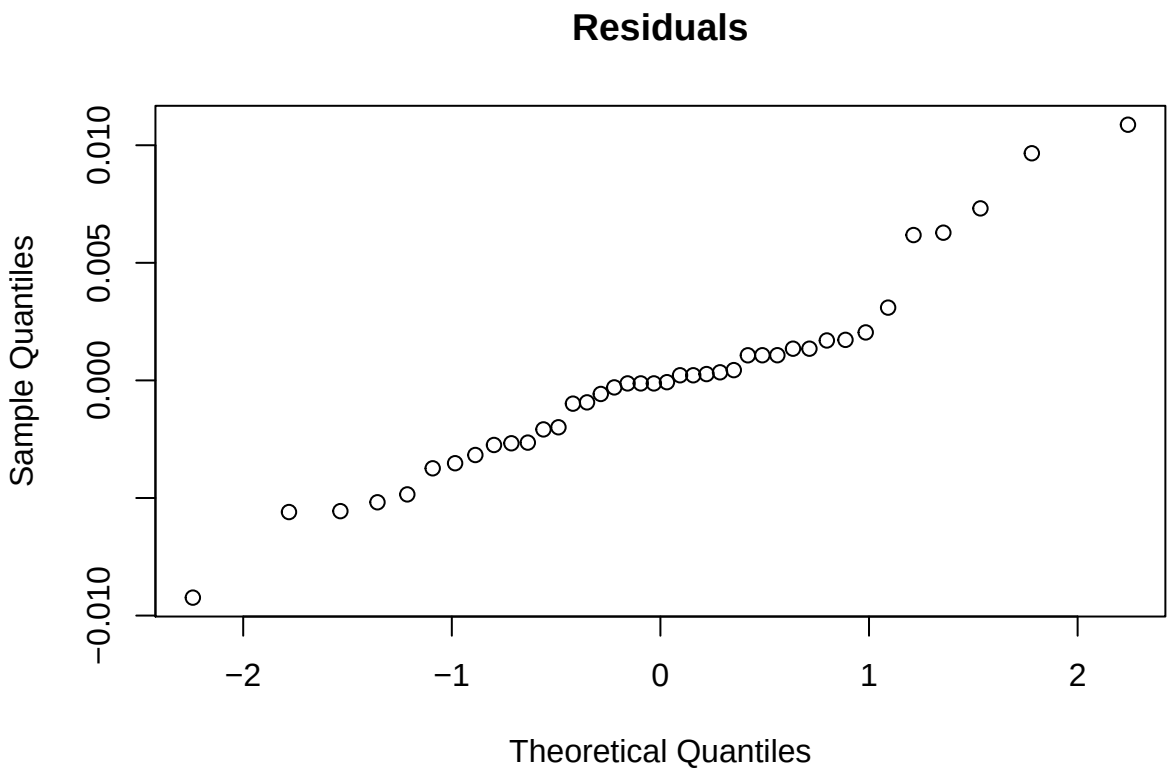
```r
boxCox(lm(count~sample:lab,data=Interlaboratory))
```



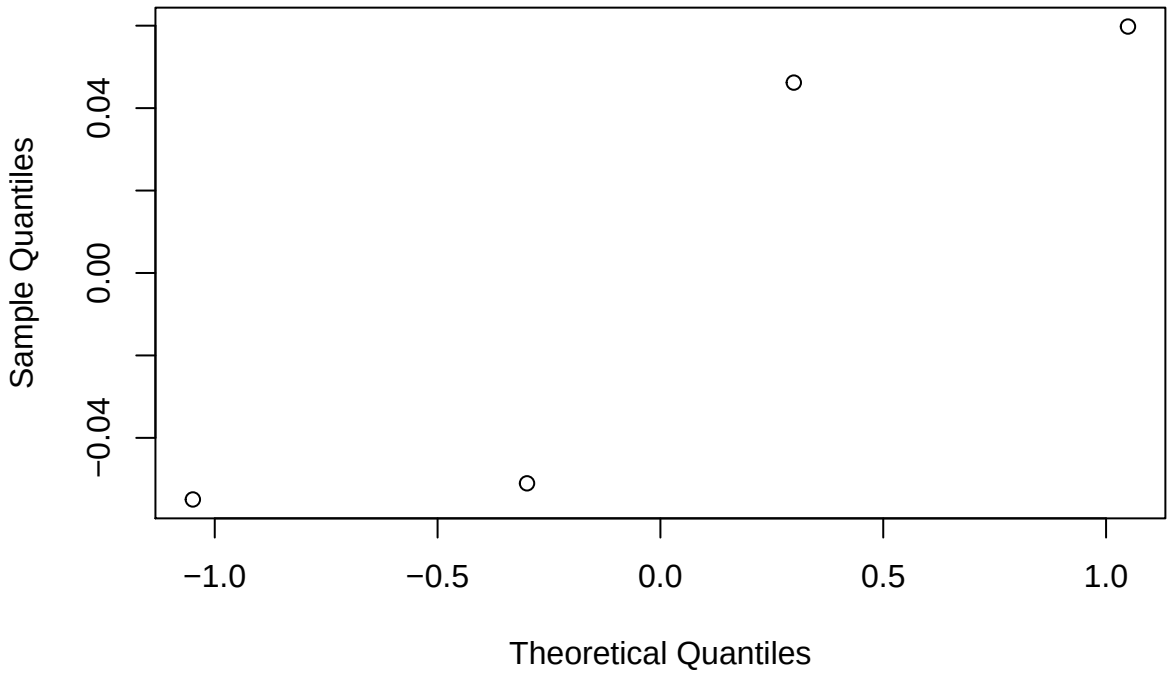

```r
##Take lambda=-0.25
int.lmer_2<-lmer((count)^(-0.25)~1+(1|sample)+(1|lab)+(1|sample:lab))
##Check assumptions:
plot(int.lmer_2)
```



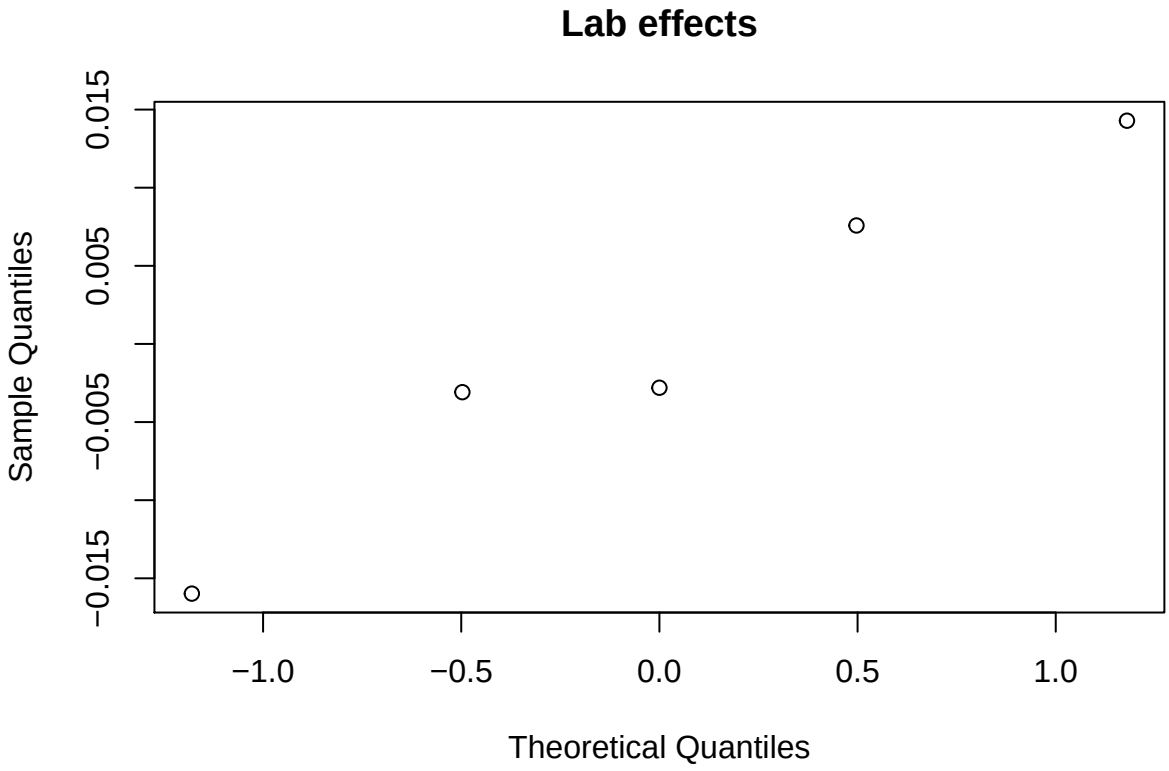
```r
qqnorm(residuals(int.lmer_2),main = "Residuals")
```



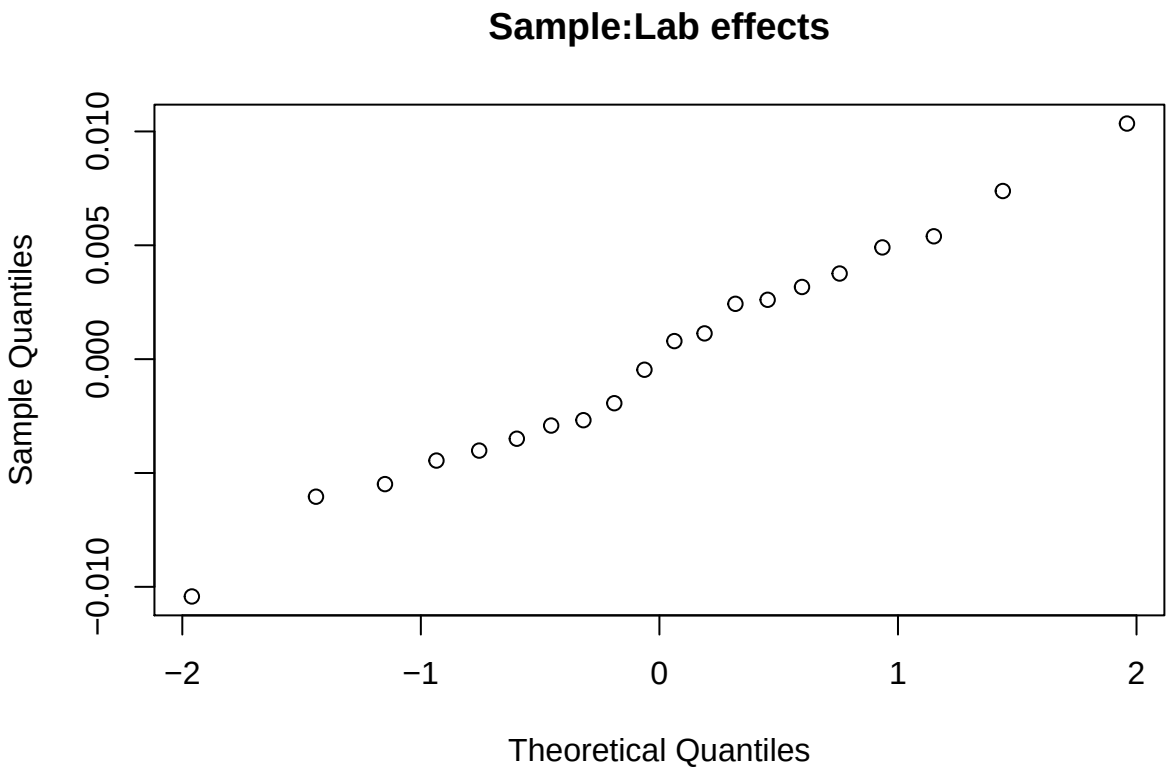

```r
qqnorm(ranef(int.lmer_2)$"sample"[[1]],main="Sample effects")
```



```
qqnorm(ranef(int.lmer_2)$"lab"[[1]],main="Lab effects")
```




```
qqnorm(ranef(int.lmer_2)$"sample:lab"[[1]],main="Sample:Lab effects")
```



```
##After transformation. the canstant variance and normarlity have been significantly improved
summary(int.lmer_2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: (count)^(-0.25) ~ 1 + (1 | sample) + (1 | lab) + (1 | sample:lab)
##
## REML criterion at convergence: -239
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.74341 -0.50068 -0.01925  0.25465  2.05220
##
## Random effects:
##  Groups     Name        Variance  Std.Dev.
##  sample:lab (Intercept) 5.104e-05 0.007144
##  lab        (Intercept) 1.482e-04 0.012176
##  sample     (Intercept) 3.790e-03 0.061562
##  Residual               2.807e-05 0.005298
## Number of obs: 40, groups:  sample:lab, 20; lab, 5; sample, 4
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.19005    0.03131    6.07
```

Test random effects:

```
int_onlysample<-lmer((count)^(-0.25)~1+(1|sample))
int_onlylab<-lmer((count)^(-0.25)~1+(1|lab))
```

```
## boundary (singular) fit: see ?isSingular
```

```
int_onlyinter<-lmer((count)^(-0.25)~1+(1|sample:lab))
int_nosample<-lmer((count)^(-0.25)~1+(1|lab)+(1|sample:lab))
```

```
## boundary (singular) fit: see ?isSingular
```

```
int_nolab<-lmer((count)^(-0.25)~1+(1|sample)+(1|sample:lab))
int_nointer<-lmer((count)^(-0.25)~1+(1|sample)+(1|lab))
##Sample effects:
exactRLRT(int_onlysample,int.lmer_2,int_nosample)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 47.618, p-value < 2.2e-16
```

```
##P-value 2.2e-16 is so smaller
##So the effect of sample is highly statistically significant ##Lab effects:
exactRLRT(int_onlylab,int.lmer_2,int_nolab,)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
```

```
##
## data:
## RLRT = 9.7415, p-value = 4e-04
```

```
##P-value 6e-04 is so smaller
##So the effect of lab also has statistically significance
##Interaction of sample and lab:
exactRLRT(int_onlyinter,int.lmer_2,int_nointer)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 9.1196, p-value = 0.0011
```

```
##P-value 4e-04 is so smaller
##So the effect of interaction term has statistically significance as well.
AIC(int_onlysample,int.lmer_2,int_nosample,int_onlylab,int_nolab,int_onlyinter,int_nointer)
```

```
##                df       AIC
## int_onlysample  3 -195.1355
## int.lmer_2      5 -228.9537
## int_nosample    4 -183.3360
## int_onlylab     3 -104.7268
## int_nolab       4 -221.2123
## int_onlyinter   3 -185.3360
## int_nointer     4 -221.8341
```

```
BIC(int_onlysample,int.lmer_2,int_nosample,int_onlylab,int_nolab,int_onlyinter,int_nointer)
```

```
##                df        BIC
## int_onlysample  3 -190.06891
## int.lmer_2      5 -220.50933
## int_nosample    4 -176.58046
## int_onlylab     3  -99.66019
## int_nolab       4 -214.45674
## int_onlyinter   3 -180.26934
## int_nointer     4 -215.07861
```

```
##The full model has the samllest AIC and BIC
```

P11.5

```
mod<-lmer(hardness~1+(1|dentist)+(1|alloy)+(1|method)+(1|dentist:method)+(1|dentist:alloy)+(1|method:al
```

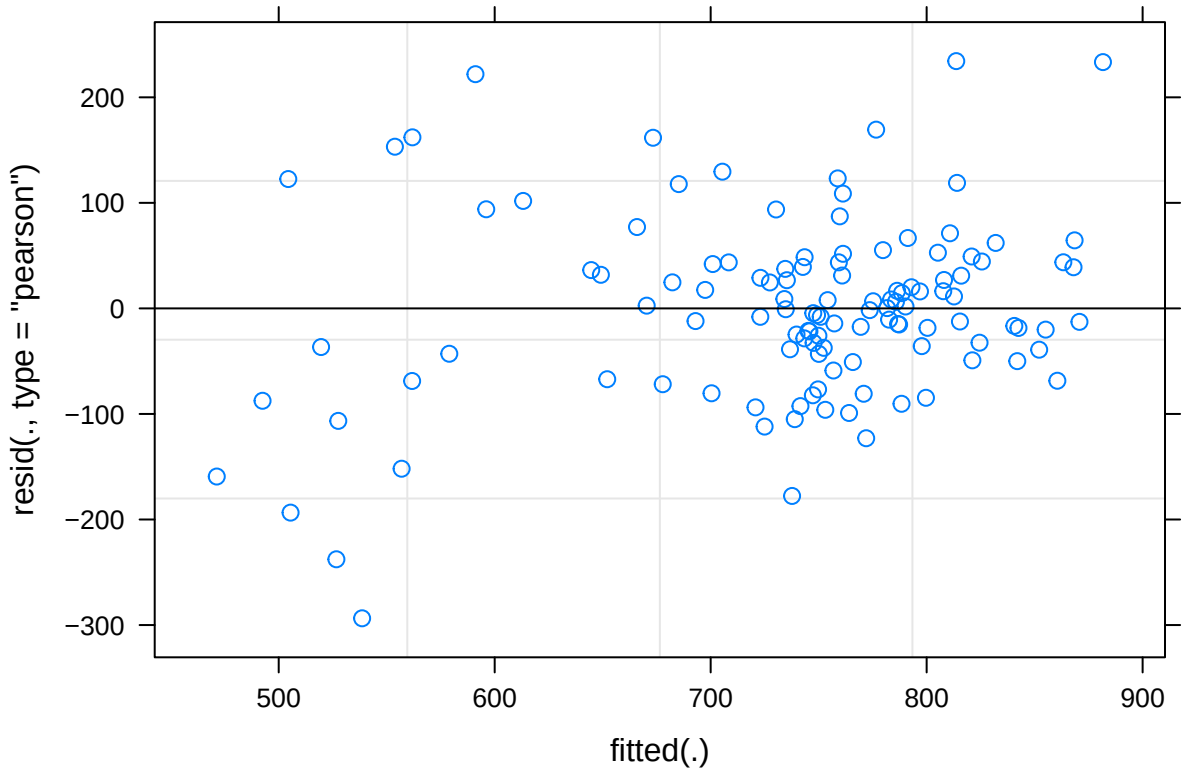
```
## boundary (singular) fit: see ?isSingular
```

```
summary(mod)
```

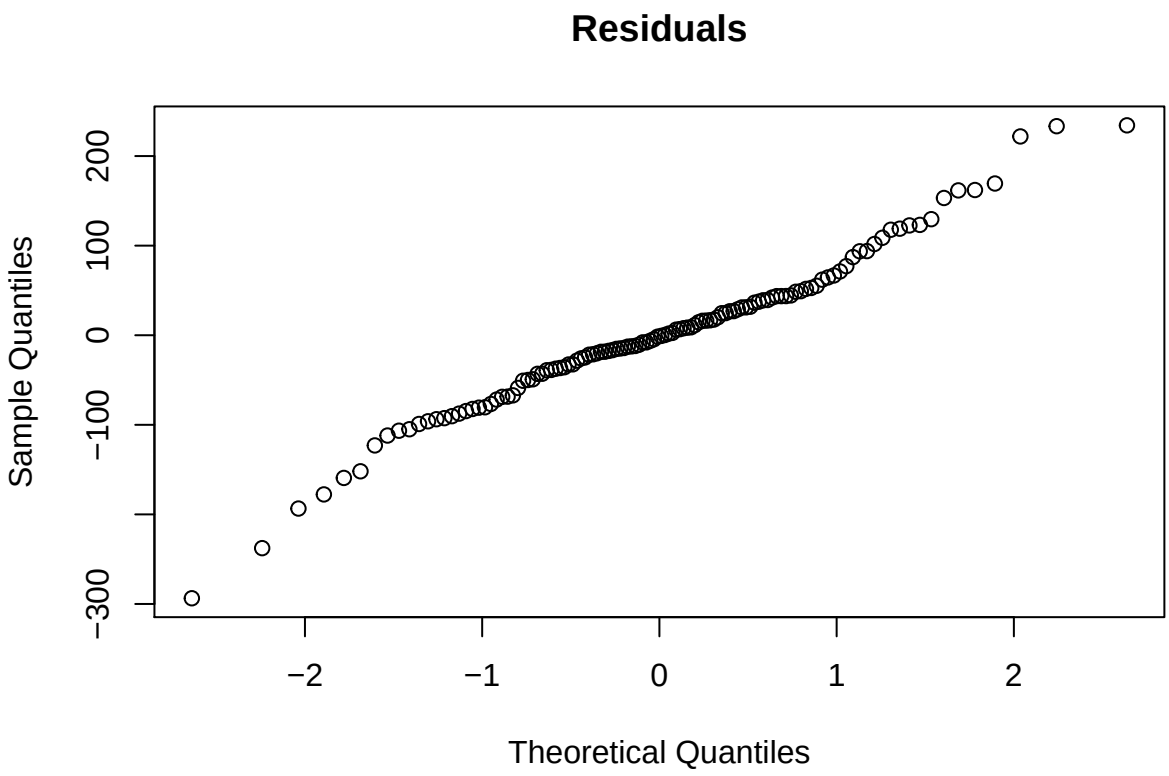
```
## Linear mixed model fit by REML ['lmerMod']
## Formula: hardness ~ 1 + (1 | dentist) + (1 | alloy) + (1 | method) + (1 |
##     dentist:method) + (1 | dentist:alloy) + (1 | method:alloy)
##
## REML criterion at convergence: 1467.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -3.07257 -0.45009 -0.01263  0.45655  2.45177
##
## Random effects:
##  Groups         Name        Variance Std.Dev.
##  dentist:alloy  (Intercept)    0.0    0.00
##  method:alloy   (Intercept) 1170.4   34.21
##  dentist:method (Intercept) 2974.7   54.54
##  alloy          (Intercept) 1099.4   33.16
##  dentist        (Intercept)  894.5   29.91
##  method         (Intercept) 6499.5   80.62
##  Residual                   9131.8   95.56
## Number of obs: 120, groups:
## dentist:alloy, 40; method:alloy, 24; dentist:method, 15; alloy, 8; dentist, 5; method, 3
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   736.65      52.97   13.91
## convergence code: 0
## boundary (singular) fit: see ?isSingular
```

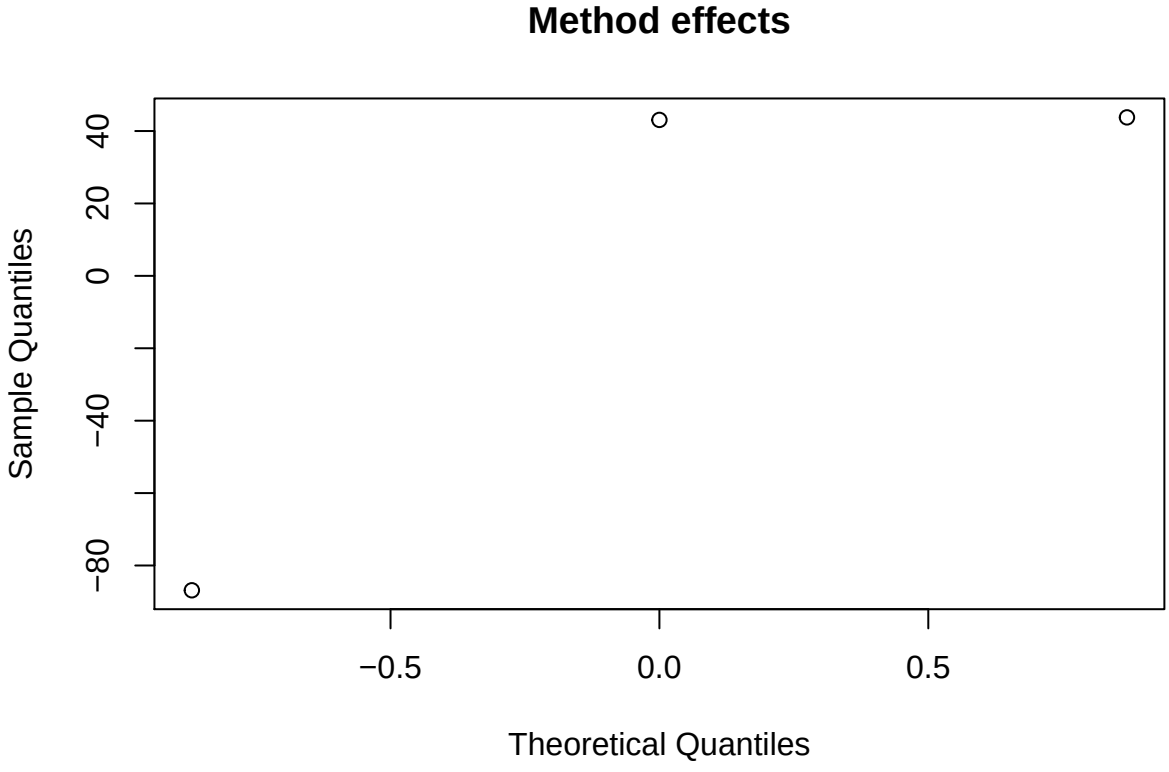
```
##Check assumptions:
plot(mod)
```



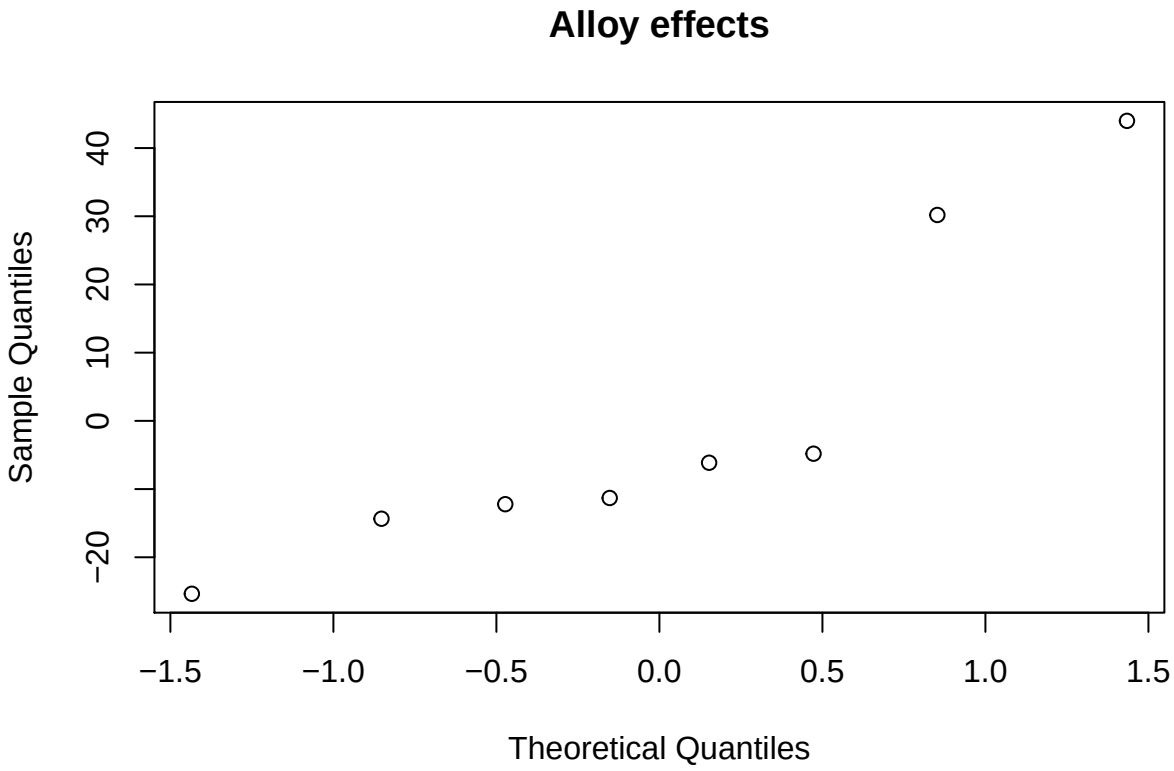

```
##Variance seems like to be good.
qqnorm(residuals(mod),main = "Residuals")
```



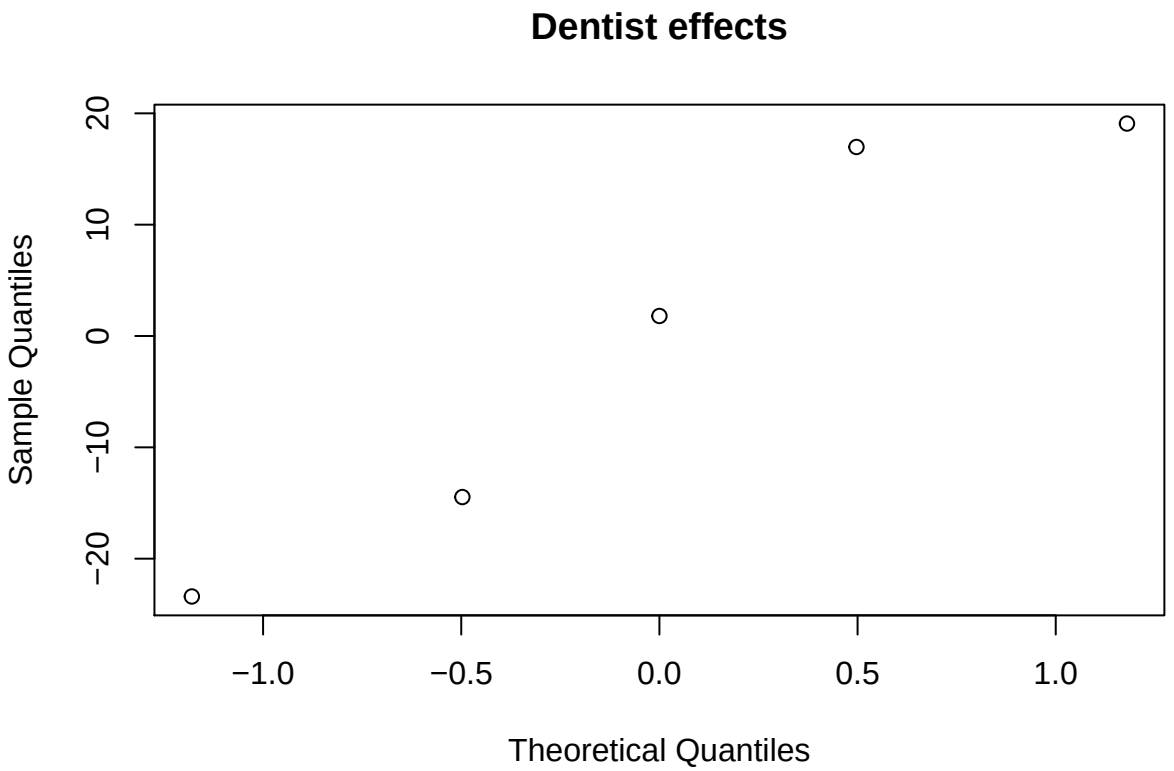

```
##The normarlity is not bad
qqnorm(ranef(mod)$"method"[[1]],main="Method effects")
```



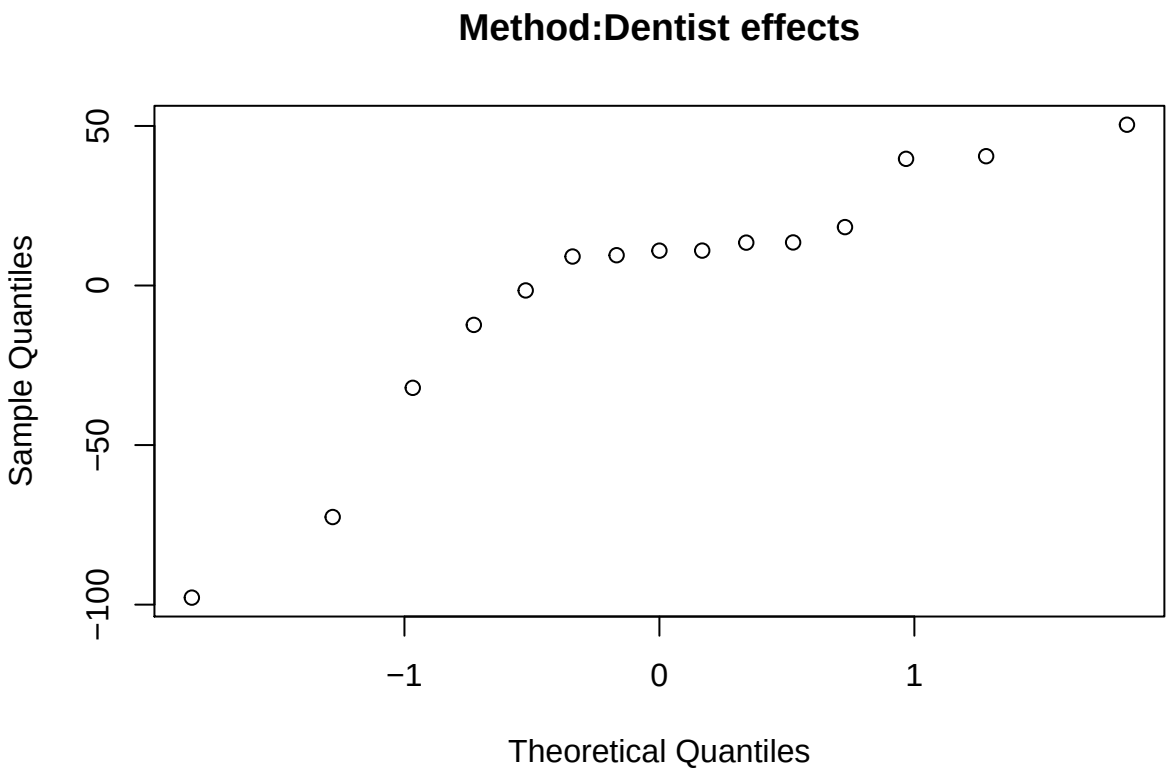
```
qqnorm(ranef(mod)$"alloy"[[1]],main="Alloy effects")
```



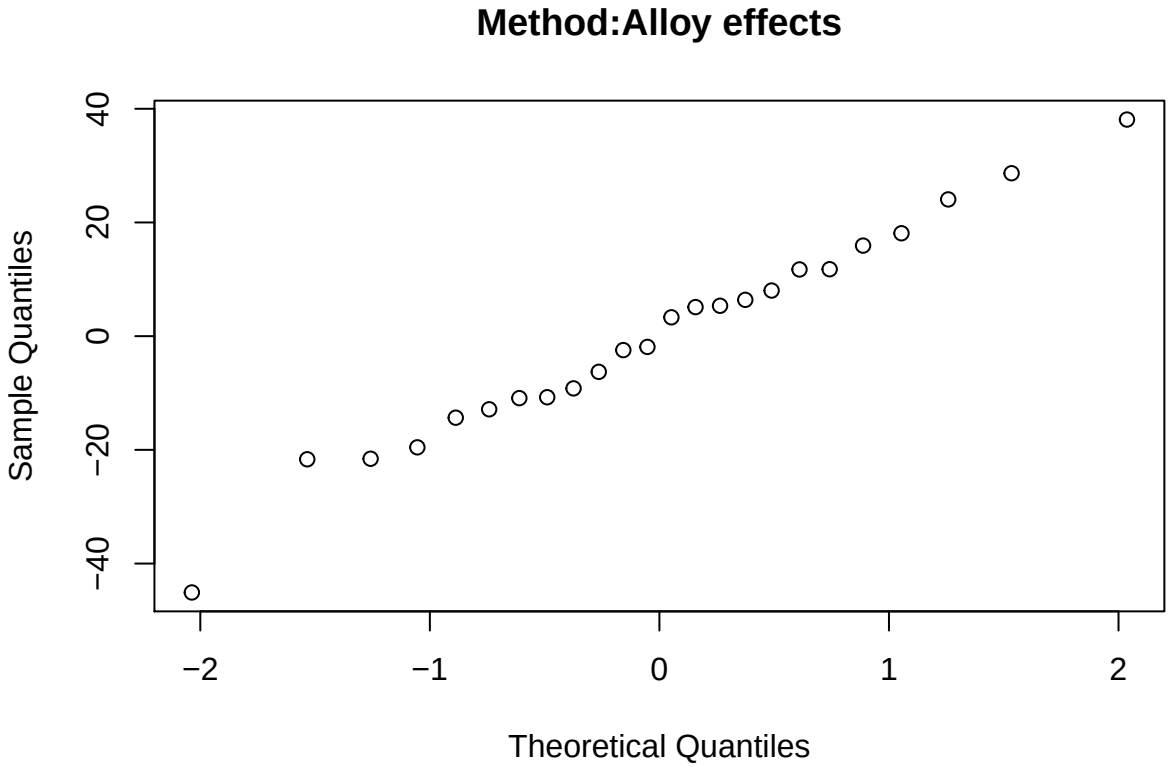

```
qqnorm(ranef(mod)$"dentist"[[1]],main="Dentist effects")
```



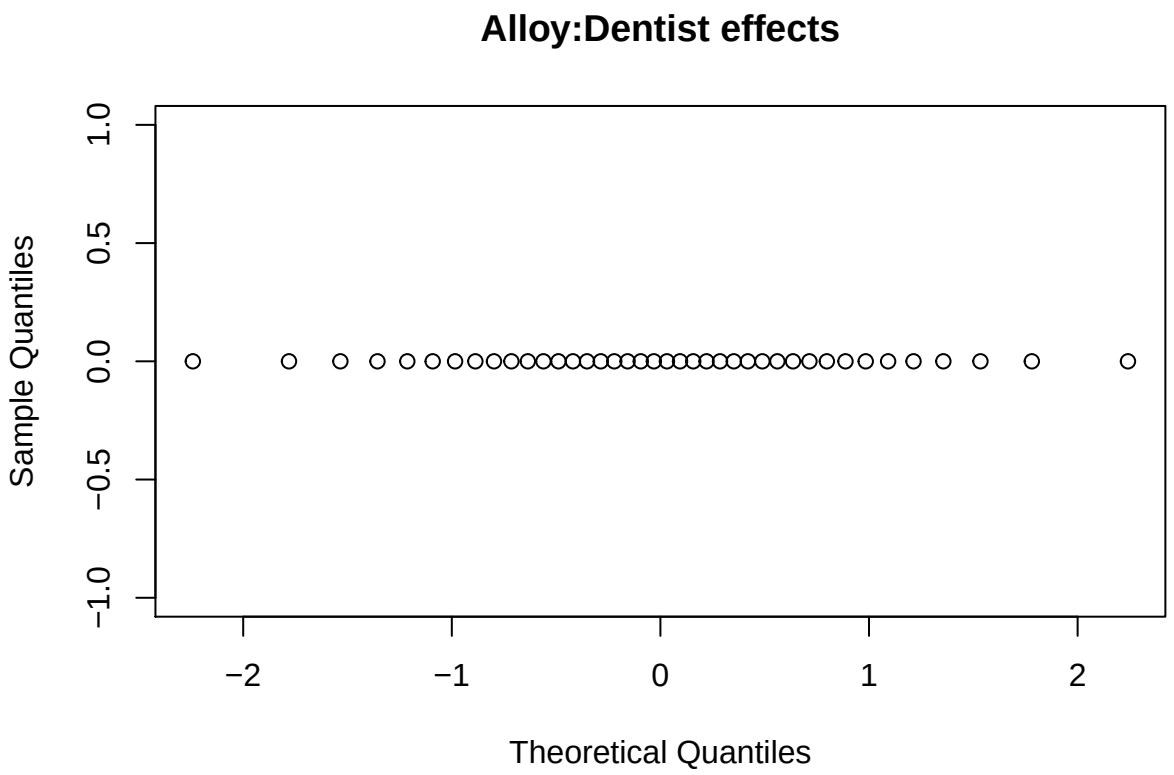
```r
qqnorm(ranef(mod)$"dentist:method"[[1]],main="Method:Dentist effects")
```




```r
qqnorm(ranef(mod)$"method:alloy"[[1]],main="Method:Alloy effects")
```



```
qqnorm(ranef(mod)$"dentist:alloy"[[1]],main="Alloy:Dentist effects")
```



Test random effects:

```
only_dentist<-lmer(hardness~1+(1|dentist))
only_method<-lmer(hardness~1+(1|method))
only_alloy<-lmer(hardness~1+(1|alloy))
only_dentist_method<-lmer(hardness~1+(1|dentist:method))
only_method_alloy<-lmer(hardness~1+(1|method:alloy))
only_dentist_alloy<-lmer(hardness~1+(1|dentist:alloy))
```

```
## boundary (singular) fit: see ?isSingular
```

```
##
no_dentist<-lmer(hardness~1+(1|alloy)+(1|method)+(1|alloy:method)+(1|dentist:method)+(1|dentist:alloy))
```

```
## boundary (singular) fit: see ?isSingular
```

```
no_method<-lmer(hardness~1+(1|dentist)+(1|alloy)+(1|alloy:method)+(1|dentist:method)+(1|dentist:alloy))
```

```
## boundary (singular) fit: see ?isSingular
```

```
no_alloy<-lmer(hardness~1+(1|dentist)+(1|method)+(1|alloy:method)+(1|dentist:method)+(1|dentist:alloy))
```

```
## boundary (singular) fit: see ?isSingular
```

```
no_dentist_method<-lmer(hardness~1+(1|dentist)+(1|alloy)+(1|method)+(1|alloy:method) +(1|dentist:alloy))
```

```
## boundary (singular) fit: see ?isSingular
```

```
no_method_alloy<-lmer(hardness~1+(1|dentist)+(1|alloy)+(1|method)+(1|dentist:method)+(1|dentist:alloy))
```

```
## boundary (singular) fit: see ?isSingular
```

```
no_dentist_alloy<-lmer(hardness~1+(1|dentist)+(1|alloy)+(1|method)+(1|alloy:method)+(1|dentist:method))
##Tets dentist:
exactRLRT(only_dentist,mod,no_dentist)##Not significant
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.35184, p-value = 0.2086
```

```
##Tets method:
exactRLRT(only_method,mod,no_method)##Significant
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 4.9954, p-value = 0.0063
```

```
##Tets alloy:
exactRLRT(only_alloy,mod,no_alloy)##Not significant
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 1.3676, p-value = 0.0949
```

```
##Tets dentist:method:
exactRLRT(only_dentist_method,mod,no_dentist_method)##Not significant
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 8.5309, p-value = 0.0012
```

```
##Tets dentist:alloy:
exactRLRT(only_dentist_alloy,mod,no_dentist_alloy)##Significant
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 1.4006e-08, p-value = 0.4789
```

```
##Tets method:alloy:
exactRLRT(only_method_alloy,mod,no_method_alloy)##Not significant
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 1.6516, p-value = 0.0968
```

P11.9

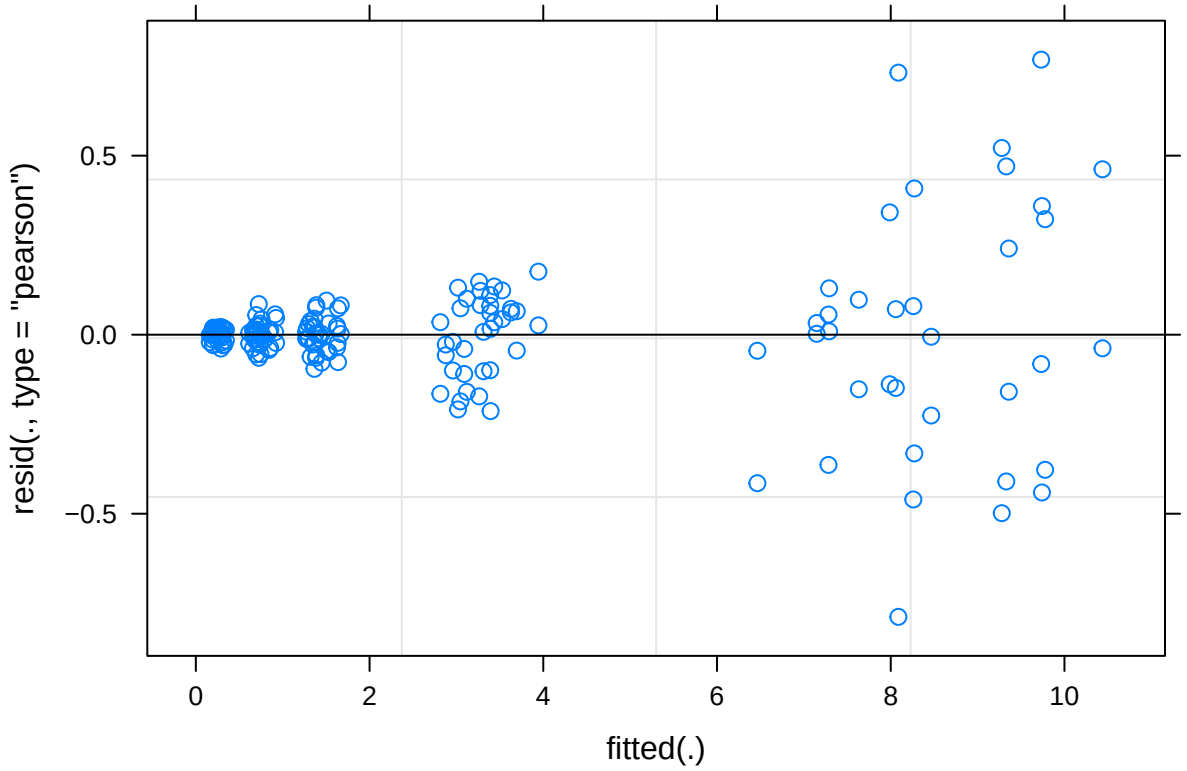
```
mod_dna<-lmer(od260~1+(1|vol)+(1|conc)+(1|user)+(1|vol:conc)+(1|vol:conc:user))
```

```
## boundary (singular) fit: see ?isSingular
```

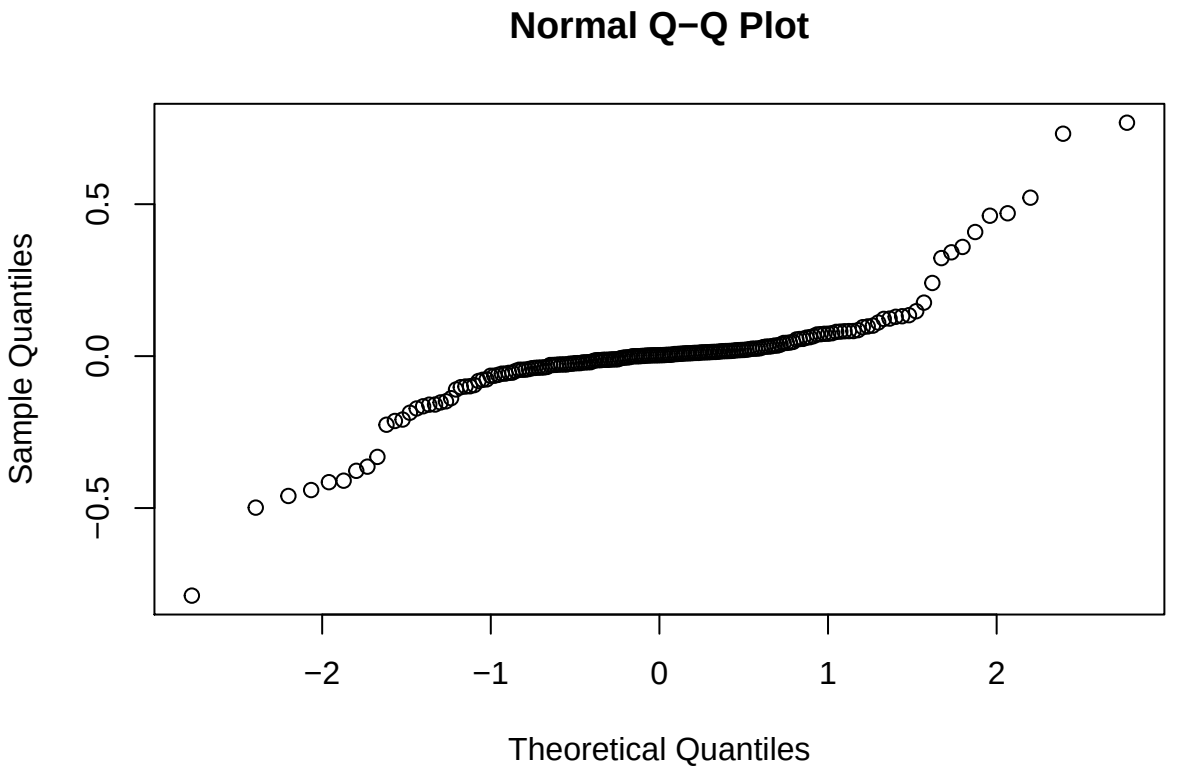
```
summary(mod_dna)##Check assumptions:
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: od260 ~ 1 + (1 | vol) + (1 | conc) + (1 | user) + (1 | vol:conc) +
##     (1 | vol:conc:user)
##
## REML criterion at convergence: 173.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5820 -0.1634  0.0139  0.1502  3.4908
##
## Random effects:
##  Groups        Name        Variance Std.Dev.
##  vol:conc:user (Intercept)  0.06680 0.2585
##  vol:conc      (Intercept)  0.18776 0.4333
##  vol           (Intercept)  0.06346 0.2519
##  conc          (Intercept) 11.23324 3.3516
##  user          (Intercept)  0.00000 0.0000
##  Residual                   0.04841 0.2200
## Number of obs: 180, groups:
## vol:conc:user, 90; vol:conc, 30; vol, 6; conc, 5; user, 3
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)    2.839      1.505   1.887
## convergence code: 0
## boundary (singular) fit: see ?isSingular
```

```
plot(mod_dna)
```

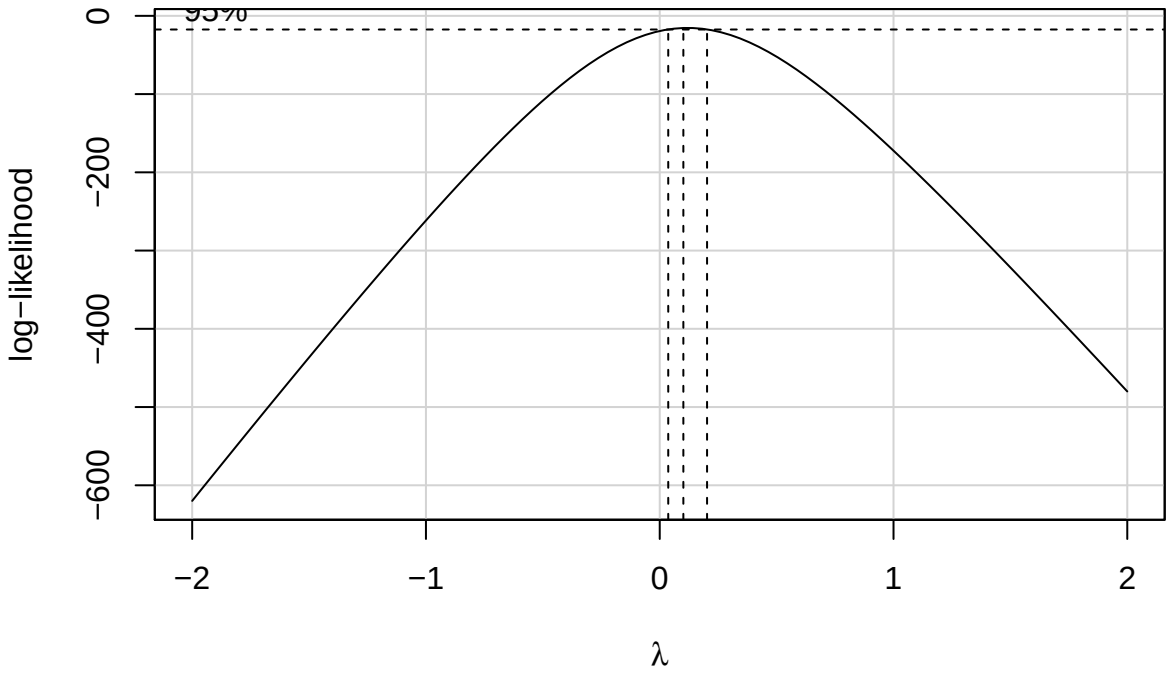



```
##Constant Variance looks like not good
qqnorm(residuals(mod_dna))
```

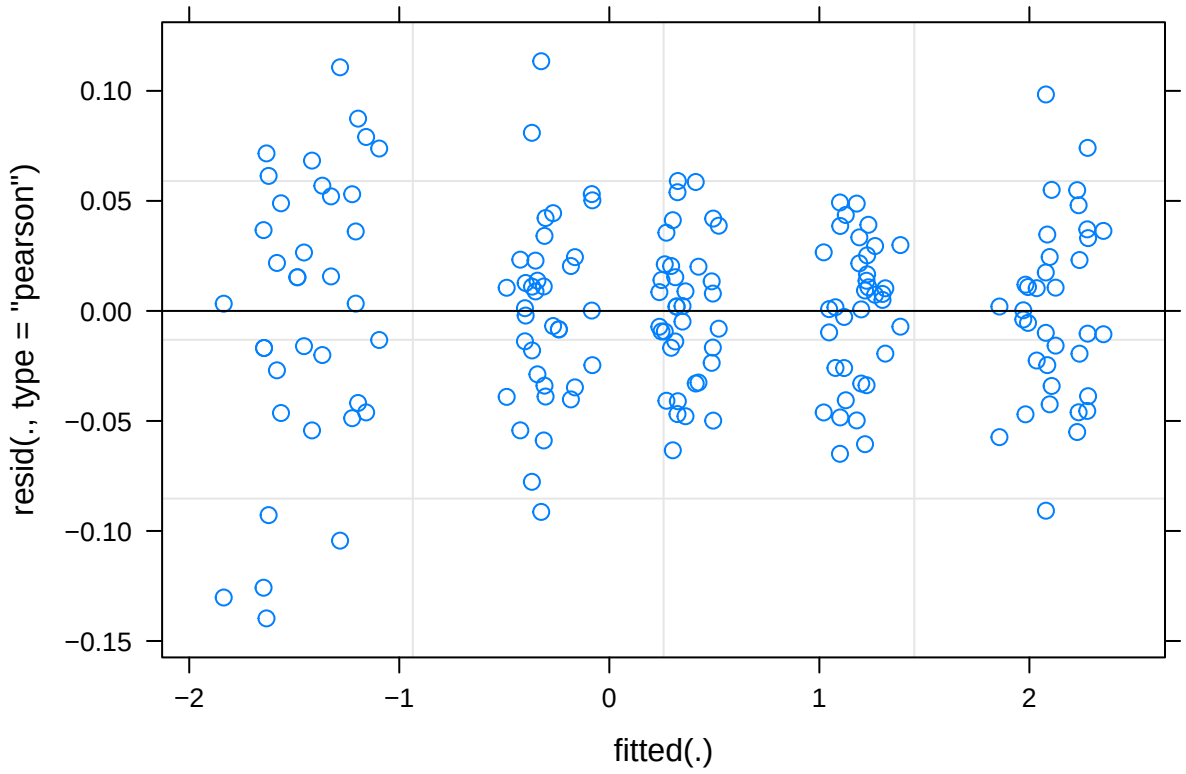



```
##Nomarlity if residuals is not too bad
##Use transformation
```

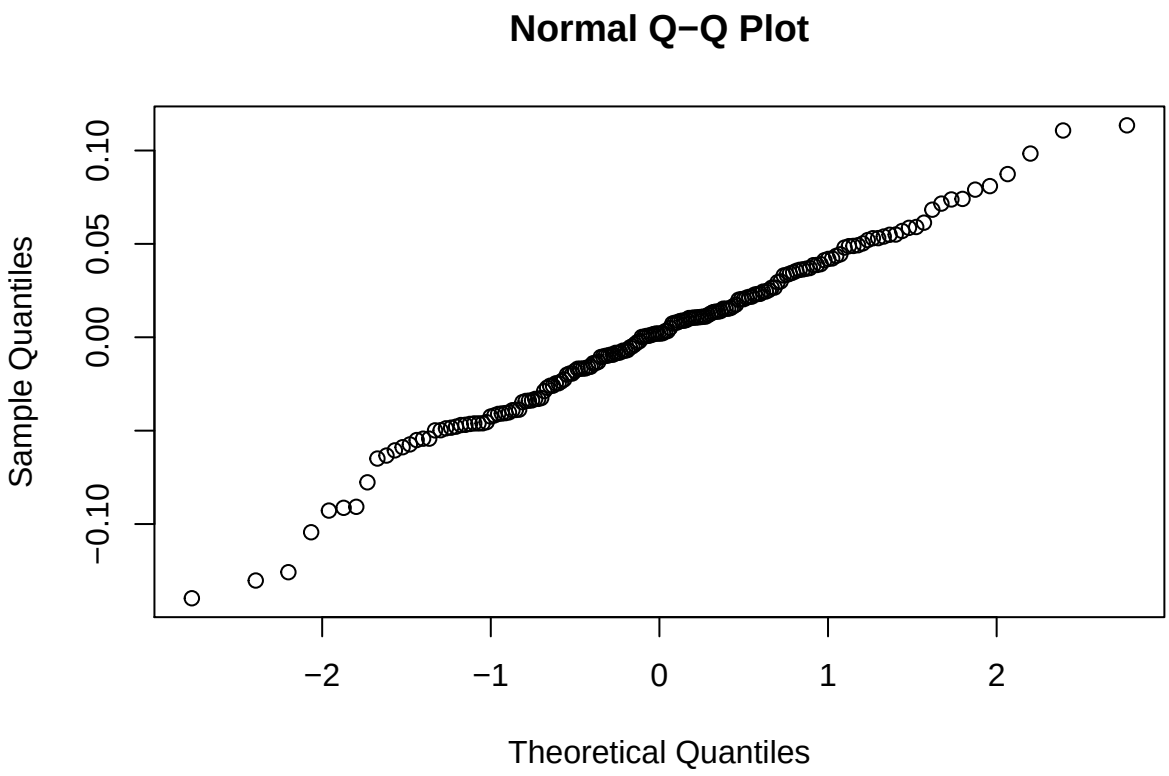
```
boxCox(lm(od260~1+vol+conc+user+vol:conc+vol:conc:user),data=DNA)
```



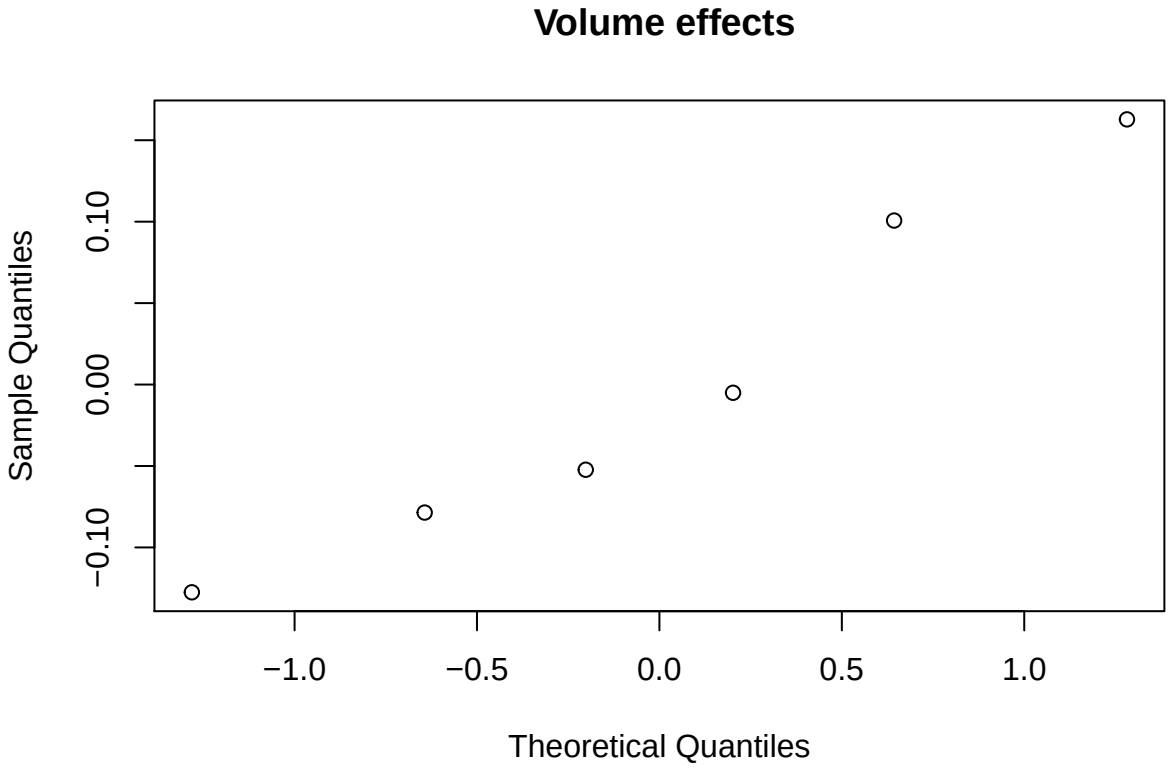

```
##Take Log transfromation:
mod_dna_1<-lmer(log(od260)~1+(1|vol)+(1|conc)+(1|user)+(1|vol:conc)+(1|vol:conc:user))
plot(mod_dna_1)
```



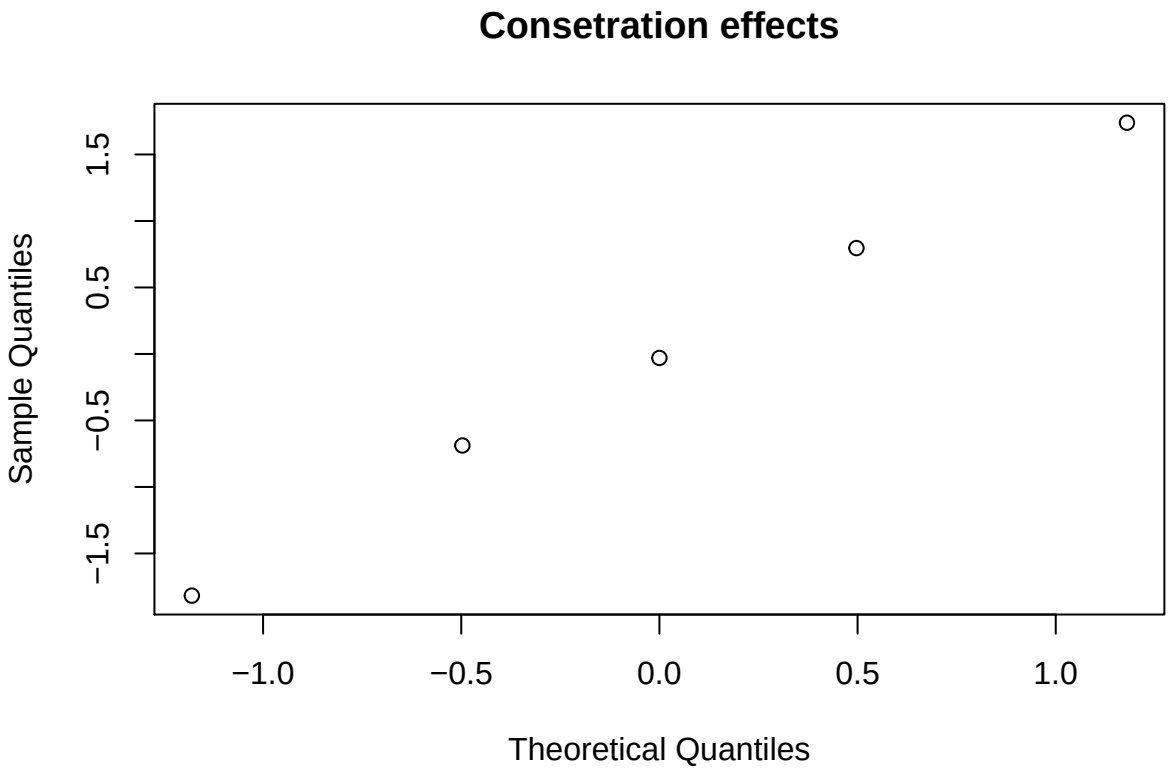

```
##Constant Variance looks like good
qqnorm(residuals(mod_dna_1))
```



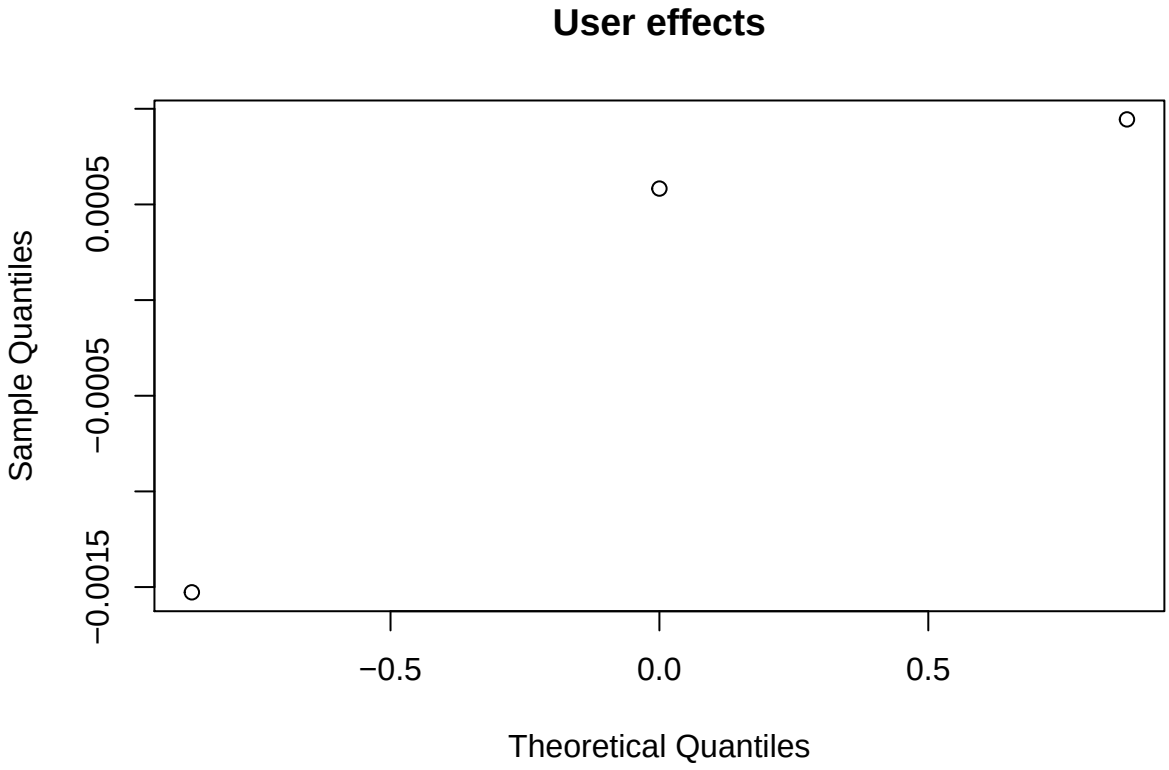

```
##Nomarlity of residuals is also bad
qqnorm(ranef(mod_dna_1)$"vol"[[1]],main=" Volume effects")
```



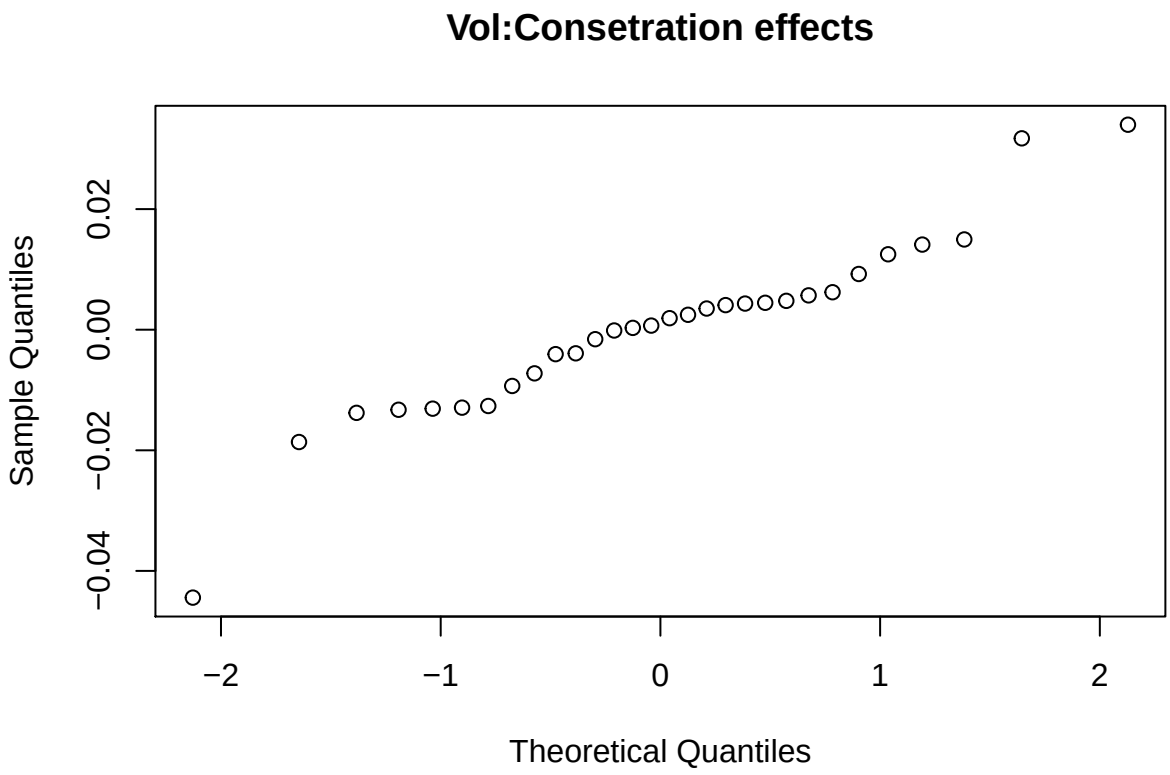
```
qqnorm(ranef(mod_dna_1)$"conc"[[1]],main="Consetration effects")
```



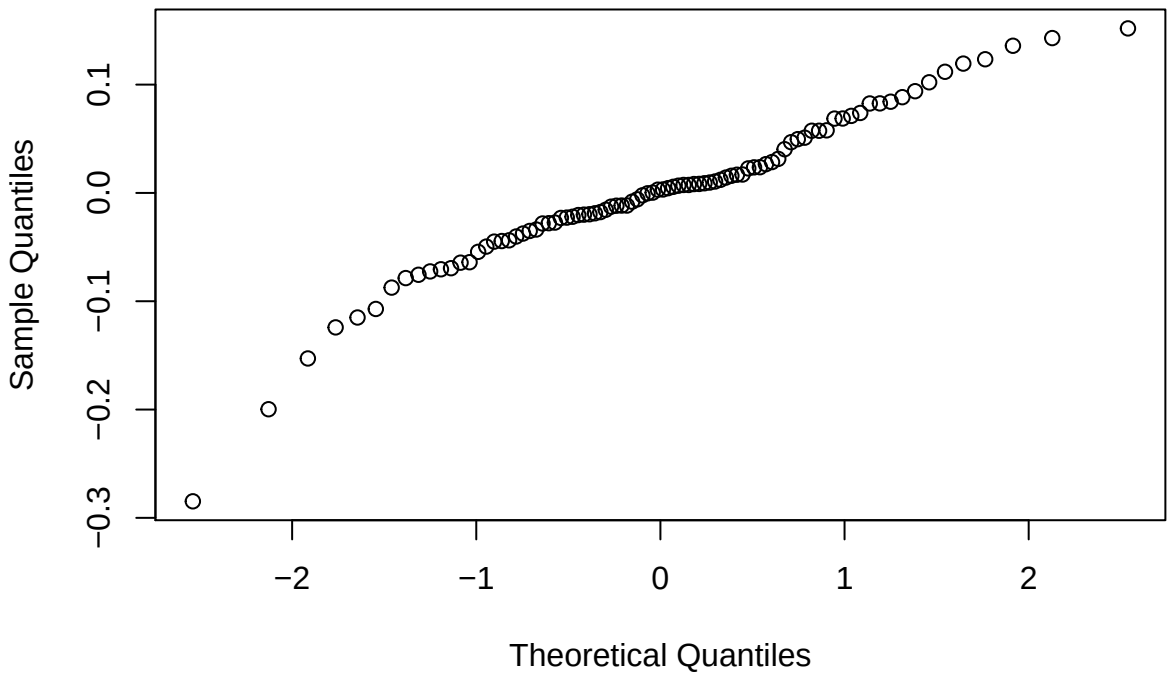

```
qqnorm(ranef(mod_dna_1)$"user"[[1]],main="User effects")
```



```
qqnorm(ranef(mod_dna_1)$"vol:conc"[[1]],main="Vol:Consetration effects")
```




```
qqnorm(ranef(mod_dna_1)$"vol:conc:user"[[1]],main="Vol:Consetration:User effects")
```



```
summary(mod_dna_1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## log(od260) ~ 1 + (1 | vol) + (1 | conc) + (1 | user) + (1 | vol:conc) +
##     (1 | vol:conc:user)
##
## REML criterion at convergence: -304.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.4433 -0.4800  0.0349  0.4645  1.9841
##
## Random effects:
##  Groups        Name        Variance  Std.Dev.
##  vol:conc:user (Intercept) 7.335e-03 0.085642
##  vol:conc      (Intercept) 1.137e-03 0.033721
##  vol           (Intercept) 1.315e-02 0.114654
##  conc          (Intercept) 1.859e+00 1.363569
##  user          (Intercept) 2.399e-05 0.004898
##  Residual                  3.271e-03 0.057189
## Number of obs: 180, groups:
## vol:conc:user, 90; vol:conc, 30; vol, 6; conc, 5; user, 3
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   0.3880     0.6117   0.634
```

Test random effects:

```
dna_onlyvol<-lmer(od260~1+(1|vol))
```

```
## boundary (singular) fit: see ?isSingular
```

```
dna_onlycon<-lmer(od260~1+(1|conc))
dna_onlyuser<-lmer(od260~1+(1|user))
```

```
## boundary (singular) fit: see ?isSingular
```

```
dna_onlyvolCon<-lmer(od260~1+(1|vol:conc))
dna_onlyvolConUser<-lmer(od260~1+(1|vol:conc:user))
dna_novol<-lmer(log(od260)~1+(1|conc)+(1|user)+(1|vol:conc)+(1|vol:conc:user))
dna_nocon<-lmer(log(od260)~1+(1|vol)+(1|user)+(1|vol:conc)+(1|vol:conc:user))
dna_nouser<-lmer(log(od260)~1+(1|vol)+(1|conc)+(1|vol:conc)+(1|vol:conc:user))
dna_novolCon<-lmer(log(od260)~1+(1|vol)+(1|conc)+(1|user)+(1|vol:conc:user))
```

```
## boundary (singular) fit: see ?isSingular
```

```
dna_novolConUser<-lmer(log(od260)~1+(1|vol)+(1|conc)+(1|user)+(1|vol:conc))
##Test Vol:
exactRLRT(dna_onlyvol,mod_dna_1,dna_novol)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
```

```
## data:
## RLRT = 21.644, p-value < 2.2e-16
```

```
##P-value is  so smaller
##So the effect of Volume has statistically significance
##Test Conc:
exactRLRT(dna_onlycon,mod_dna_1,dna_nocon)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 126.25, p-value < 2.2e-16
```

```
##P-value is  so smaller
##So the effect of Concentration has statistically significance
##Test User:
exactRLRT(dna_onlyuser,mod_dna_1,dna_nouser)##Not significant
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.0058972, p-value = 0.3462
```

```
##Test Vol:Conc:
exactRLRT(dna_onlyvolCon,mod_dna_1,dna_novolCon)##Not significant
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.81213, p-value = 0.1713
```

```
##Test Vol:Conc:User
exactRLRT(dna_onlyvolConUser,mod_dna_1,dna_novolConUser)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 51.413, p-value < 2.2e-16
```

```
##P-value is  so smaller
##So the effect of the interaction of Volume:Concetration:User has statistically significance
confint(mod_dna_1,oldNames=FALSE,level = 0.9)
```

```
## Computing profile confidence intervals ...
```

```
##                                      5 %       95 %
```

```
## sd_(Intercept)|vol:conc:user  0.07089729 0.10351701
## sd_(Intercept)|vol:conc       0.00000000 0.06625154
## sd_(Intercept)|vol            0.07001279 0.21907337
## sd_(Intercept)|conc           0.78290452 2.28772300
## sd_(Intercept)|user           0.00000000 0.05244695
## sigma                         0.05083432 0.06498665
## (Intercept)                  -0.64848277 1.42443187
```

P11.10

I think we should increase a to make the power for testing A to be very high.

```
mixed.power(~A*B,c(2,4,3),list(A=0.01,"A:B"=0.02,Error=0.05),resrict=FALSE)
```

```
##           num.ev den.ev num.df den.df power
## Intercept   0.05   0.05      1     16  0.05
## A           0.17   0.05      1     16  0.31
## B           0.05   0.05      3     16  0.05
## A:B         0.11   0.05      3     16  0.26
```

```
##Increse a:
mixed.power(~A*B,c(8,4,3),list(A=0.01,"A:B"=0.02,Error=0.05),resrict=FALSE)
```

```
##           num.ev den.ev num.df den.df power
## Intercept   0.05   0.05      1     64  0.05
## A           0.17   0.05      7     64  0.82
## B           0.05   0.05      3     64  0.05
## A:B         0.11   0.05     21     64  0.77
```

```
##increase n:
mixed.power(~A*B,c(2,4,10),list(A=0.01,"A:B"=0.02,Error=0.05),resrict=FALSE)
```

```
##           num.ev den.ev num.df den.df power
## Intercept   0.05   0.05      1     72  0.05
## A           0.45   0.05      1     72  0.80
## B           0.05   0.05      3     72  0.05
## A:B         0.25   0.05      3     72  0.82
```

```
##increase b:
mixed.power(~A*B,c(2,8,3),list(A=0.01,"A:B"=0.02,Error=0.05),resrict=FALSE)
```

```
##           num.ev den.ev num.df den.df power
## Intercept   0.05   0.05      1     32  0.05
## A           0.29   0.05      1     32  0.57
## B           0.05   0.05      7     32  0.05
## A:B         0.11   0.05      7     32  0.43
```