```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.1.3
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.3
```

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.1.3
```

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

```
## Warning: package 'carData' was built under R version 4.1.3
```

```
library(lme4)
```

```
## Warning: package 'lme4' was built under R version 4.1.3
```

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

```
library(emmeans)
```

```
## Warning: package 'emmeans' was built under R version 4.1.3
```

```
library(magrittr)
```

```
## Warning: package 'magrittr' was built under R version 4.1.3
```

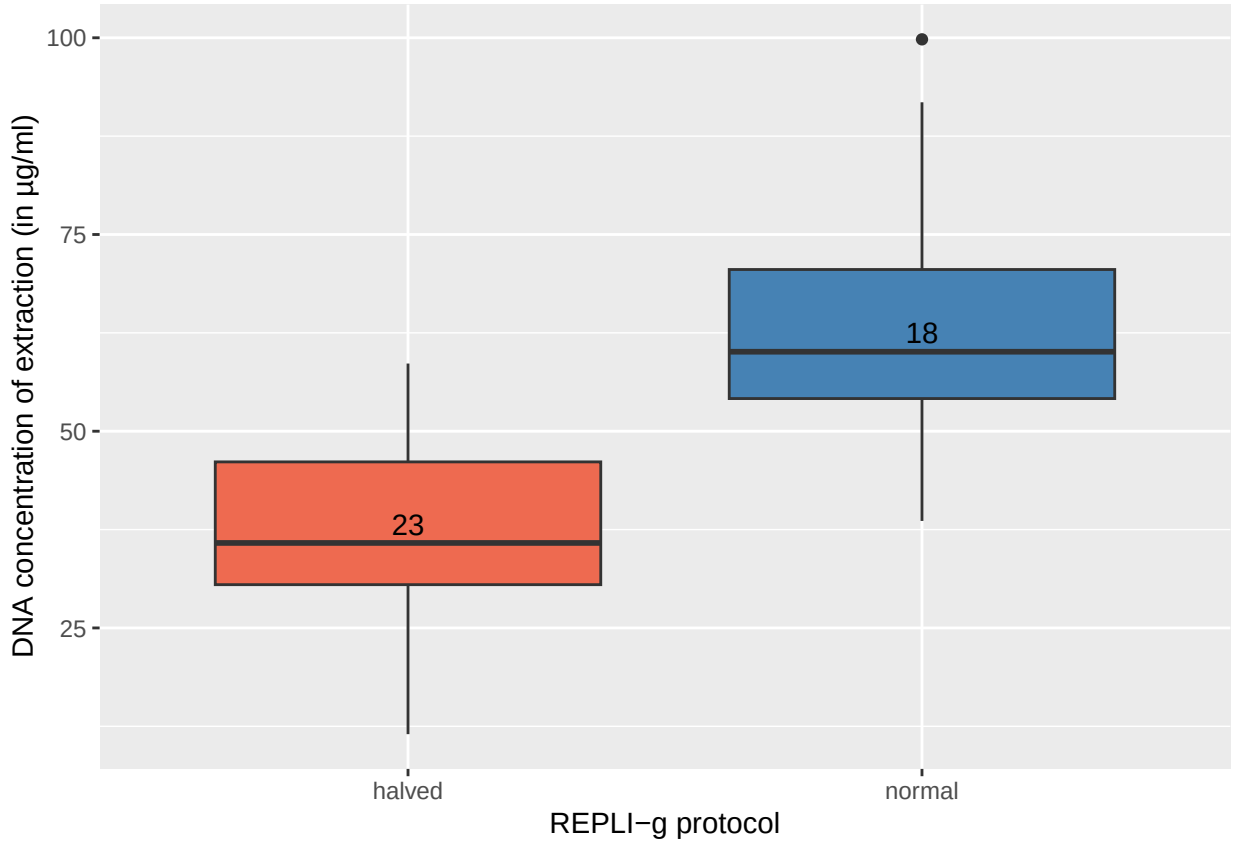
```
library(DHARMa)
```

```
## Warning: package 'DHARMa' was built under R version 4.1.3
```

```
## This is DHARMa 0.4.6. For overview type '?DHARMa'. For recent changes, type news(package = 'DHARMa')
```

```
Metingen_DNA_conc <- read_excel("F:/UGent/2022-2023/Masterthesis/Measurements DNA concentrations.xlsx")

Metingen_DNA_conc %>% #Boxplots of the DNA concentration per extraction protocol
  ggplot() + aes(RepliG, conc_origineel_yg_ml, fill = RepliG) +
  geom_boxplot() +
  xlab('REPLI-g protocol') +
  ylab('DNA concentration of extraction (in µg/ml)') +
  theme(legend.position = "none") +
  annotate("text",
           x = 1:length(table(Metingen_DNA_conc$RepliG)),
           y = aggregate(conc_origineel_yg_ml ~ RepliG, Metingen_DNA_conc, median)[ , 2],
           label = table(Metingen_DNA_conc$RepliG),
           col = "black",
           vjust = -0.4)+
  scale_fill_manual('Storage', values=c('coral2','steelblue'))
```




```
#linear model of the DNA concentration depending on the extraction protocol
lm.model1<-lmer(conc_origineel_yg_ml~RepliG+(1|Species), data=Metingen_DNA_conc)
summary(lm.model1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: conc_origineel_yg_ml ~ RepliG + (1 | Species)
##    Data: Metingen_DNA_conc
##
## REML criterion at convergence: 322.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.6990 -0.6317 -0.1627  0.5203  2.5908
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Species  (Intercept)  15.18    3.896
##  Residual             182.05   13.493
## Number of obs: 41, groups:  Species, 29
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    37.194      2.972  12.517
## RepliGnormal   26.242      4.437   5.914
##
## Correlation of Fixed Effects:
```

```
##             (Intr)
## RepliGnorml -0.651
```

```
anova(lm.model1, ddf="Satterthwaite",type=3)
```

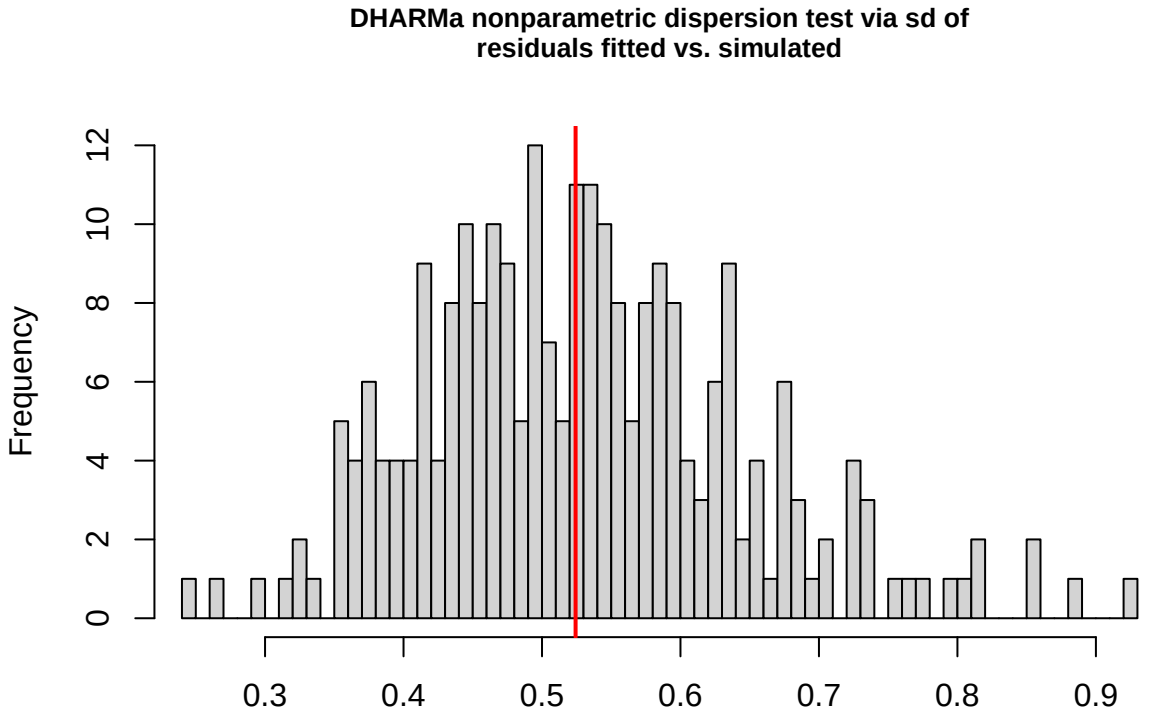
```
## Warning in anova.merMod(lm.model1, ddf = "Satterthwaite", type = 3): additional
## arguments ignored: 'ddf', 'type'
```

```
## Analysis of Variance Table
##        npar Sum Sq Mean Sq F value
## RepliG    1   6367    6367  34.973
```

```
emmeans(lm.model1, specs = pairwise ~ RepliG)
```

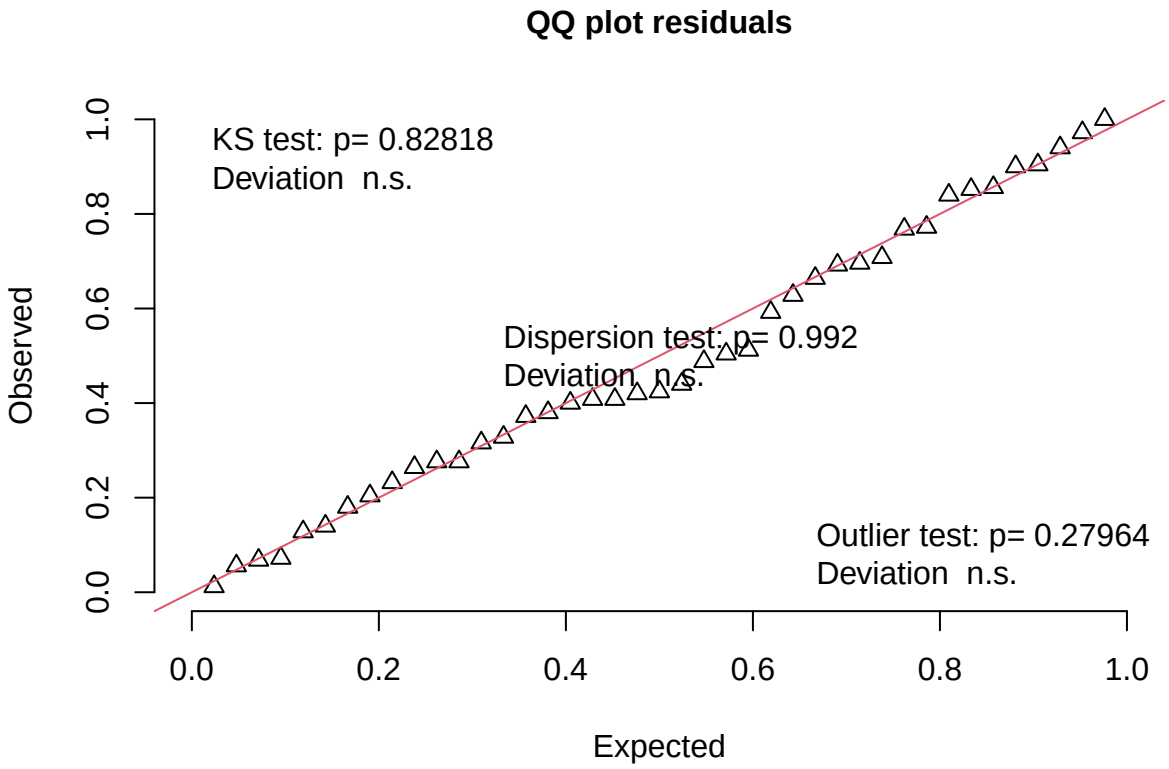
```
## $emmeans
##  RepliG emmean   SE   df lower.CL upper.CL
##  halved   37.2 3.09 31.3     30.9     43.5
##  normal   63.4 3.57 29.4     56.1     70.7
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
##  contrast        estimate   SE df t.ratio p.value
##  halved - normal    -26.2 4.65 38  -5.638  <.0001
##
## Degrees-of-freedom method: kenward-roger
```

```
#test assumptions for lmer
testDispersion(lm.model1)
```



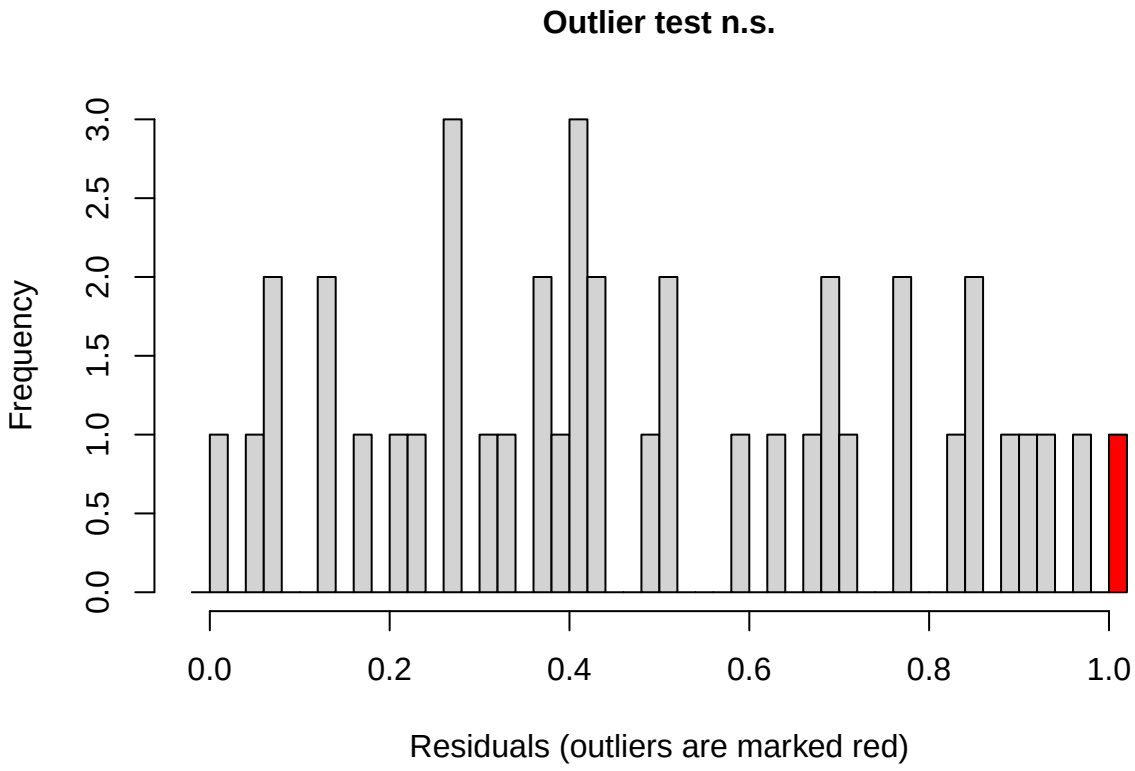

```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 0.98989, p-value = 0.992
## alternative hypothesis: two.sided
```

```
testUniformity(lm.model1)
```




```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  simulationOutput$scaledResiduals
## D = 0.097756, p-value = 0.8282
## alternative hypothesis: two-sided
```

```
testOutliers(lm.model1)
```




```
##
##  DHARMa outlier test based on exact binomial test with approximate
##  expectations
##
## data:  lm.model1
## outliers at both margin(s) = 1, observations = 41, p-value = 0.2796
## alternative hypothesis: true probability of success is not equal to 0.007968127
## 95 percent confidence interval:
##  0.0006173169 0.1285540204
## sample estimates:
## frequency of outliers (expected: 0.00796812749003984 )
##                                            0.02439024
```

```
simulationOutput <- simulateResiduals(fittedModel = lm.model1, plot = F)
residuals(simulationOutput, quantileFunction = qnorm, outlierValues = c(-7,7))
```

```
##  [1]  1.04504970  0.99445788 -0.19167090 -0.63106198 -0.32656093  0.03008408
##  [7] -0.25334710  0.23269275 -0.23269275  1.55477359 -0.59476585 -0.03008408
## [13]  0.54755135  1.28155157 -1.58926756  0.42340472 -1.49085336 -0.23269275
## [19] -0.20189348 -0.59476585 -2.25712924  0.74544955  1.30468539  1.06251930
## [25]  0.01002668  1.91103565 -0.47891373  7.00000000  0.32656093 -0.73227620
## [31] -0.82741832 -0.15096922 -0.30548079 -1.13589622 -0.91536509 -1.08031934
## [37]  0.73227620  0.50152740 -0.44544251  0.51293041 -1.46105627
```

```
plot(simulationOutput)
```