LDHA/B EM wt LMM Analysis

Erik

2/2022

```
#Sample: Mouse Optic Nerve
#Immunogold (LDHA or LDHB) stained TEM images
#Animal line: LDHAB_fl, LDHAB KO under CNP-Cre promotor
#mut = LDHA and LDHB KO
#ctr = wt littermates
#Gold = Gold of LDHA and LDHB, respectively
```

```
## Loading required package: tidyverse

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Loading required package: readxl
##
## Loading required package: ggpubr
##
## Loading required package: lme4
##
## Loading required package: Matrix
##
##
## Attaching package: 'Matrix'
##
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
##
## Loading required package: here
##
## here() starts at /Users/erik/Documents/R-Projects/LDHAB-Oligodendrocytes
##
## Loading required package: emmeans
```

```
##
## Loading required package: performance
##
## Loading required package: interactions
##
## Loading required package: lmerTest
##
##
## Attaching package: ’lmerTest’
##
##
## The following object is masked from ’package:lme4’:
##
##     lmer
##
##
## The following object is masked from ’package:stats’:
##
##     step
##
##
## Loading required package: DHARMa
##
## This is DHARMa 0.4.6. For overview type ’?DHARMa’. For recent changes, type news(package = ’DHARMa’)
##
## Loading required package: reshape2
##
##
## Attaching package: ’reshape2’
##
##
## The following object is masked from ’package:tidyr’:
##
##     smiths
##
##
## Loading required package: simr
##
##
## Attaching package: ’simr’
##
##
## The following object is masked from ’package:lme4’:
##
##     getData
##
##
## The following object is masked from ’package:stringr’:
##
##     fixed
```

```
##  [1] "tidyverse : 2.0.0"    "readxl : 1.4.2"       "ggpubr : 0.6.0"
##  [4] "lme4 : 1.1.33"        "here : 1.0.1"         "emmeans : 1.8.7"
##  [7] "performance : 0.10.4" "interactions : 1.1.5" "lmerTest : 3.1.3"
## [10] "DHARMa : 0.4.6"       "reshape2 : 1.4.4"     "simr : 1.0.7"
```

```
#Imports excel file
file = here::here("LDHAB_EM_ON_Quantification.xlsx")

ldha <- read_excel(path = file, sheet = "OL - LDHA") %>%
  group_by(Animal, Nr) %>%
  summarise(
    Image = first(Image),
    Genotype = first(Genotype),
    OligoCellBodyArea = sum(OligoCellBodyArea),
    OligoNucleusArea = sum(OligoNucleusArea),
    OligoArea = OligoCellBodyArea-OligoNucleusArea,
    OligoGold = sum(OligoGold)
  ) %>%
  full_join(
    read_excel(path = file, sheet = "Axons - LDHA")) %>%
  mutate(AxonDiameter = sqrt(AxonArea/pi)*2,
         FiberDiameter = sqrt(FiberArea/pi)*2,
         gRatio = AxonDiameter/FiberDiameter,
         gRatio = AxonDiameter/FiberDiameter,
         MyelinArea = FiberArea-AxonArea
         )
```

```
## ‘summarise()‘ has grouped output by ’Animal’. You can override using the
## ‘.groups‘ argument.
## Joining with ‘by = join_by(Animal, Nr, Image, Genotype)‘
```

```
ldha
```

```
## # A tibble: 1,086 x 21
## # Groups:   Animal [8]
##    Animal    Nr Image      Genotype OligoCellBodyArea OligoNucleusArea OligoArea
##    <chr>  <dbl> <chr>      <chr>                <dbl>            <dbl>     <dbl>
##  1 0888       1 36_0888_2~ mut                   20.6             15.0       5.59
##  2 0888       2 36_0888_4~ mut                   15.1              8.61      6.44
##  3 0888       3 36_0888_7~ mut                   14.3              8.87      5.44
##  4 0888       4 36_0888_9~ mut                   39.1             30.5       8.66
##  5 0888       5 36_0888_1~ mut                   31.9             16.0      15.9
##  6 0891       1 36_0891_1~ ctr                   31.5             26.3       5.22
##  7 0891       2 36_0891_3~ ctr                   38.1             17.1      21
##  8 0891       3 36_0891_5~ ctr                   51.2             31.2      20.0
##  9 0891       4 36_0891_1~ ctr                   29.0             24.6       4.38
## 10 0891       5 36_0891_1~ ctr                   36.5             24.6      11.9
## # i 1,076 more rows
## # i 14 more variables: OligoGold <dbl>, Scale <chr>, AxonArea <dbl>,
## #   AxonGold <dbl>, FiberArea <dbl>, MyelinGold <dbl>, AstrocyteGold <dbl>,
## #   AstrocyteArea <dbl>, GapJunctionGold <dbl>, GapJunctionLength <dbl>,
## #   AxonDiameter <dbl>, FiberDiameter <dbl>, gRatio <dbl>, MyelinArea <dbl>
```

```
ldha_ctr <- ldha %>% subset(subset = Genotype == "ctr") %>%
  dplyr::select(Animal, Image, AxonArea, AxonGold, MyelinArea, MyelinGold, AstrocyteArea, AstrocyteGold
  group_by(Image) %>%
```

```r
  summarize(Animal = first(Animal),
             across(matches("Gold|Area"),
                  list(mean = ~ mean(.,na.rm = TRUE)),
                  .names = "{.fn}_{.col}"),
            .groups = "drop") %>%
   rename_with(~ sub("mean_", "", .x), .cols = starts_with("mean_")) %>%
   pivot_longer(
    cols = matches("(Area|Gold)"),
    names_to = c("Type", ".value"),
    names_pattern = "(.*)(Area|Gold)"
  ) %>%
  add_column(StainingPerArea = .$Gold/.$Area) %>%
  na.omit() %>%
  rename(Staining = Gold)
#write.csv(ldha_ctr, file = here("ldha_ctr.csv"), row.names = FALSE)
data <- ldha_ctr

# Fit a Linear Mixer Model
model <- lmer(StainingPerArea ~ Type + (1|Animal) + (1|Animal:Type), data = data)

# Results Cell Type
summary(model) # Check the model summary
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: StainingPerArea ~ Type + (1 | Animal) + (1 | Animal:Type)
##    Data: data
##
## REML criterion at convergence: 717.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9011 -0.4911 -0.0692  0.3470  3.4856
##
## Random effects:
##  Groups      Name        Variance Std.Dev.
##  Animal:Type (Intercept) 1.6992   1.3035
##  Animal      (Intercept) 0.9542   0.9768
##  Residual                3.3075   1.8187
## Number of obs: 172, groups:  Animal:Type, 16; Animal, 4
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   8.1981     0.8560  9.1119   9.578 4.68e-06 ***
## TypeAxon     -1.4121     0.9912  8.2049  -1.425 0.191167
## TypeMyelin   -5.8094     0.9912  8.2049  -5.861 0.000343 ***
## TypeOligo    -6.4662     1.0413  9.9895  -6.210 0.000101 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) TypAxn TypMyl
## TypeAxon   -0.582
## TypeMyelin -0.582  0.503
## TypeOligo  -0.554  0.479  0.479
```

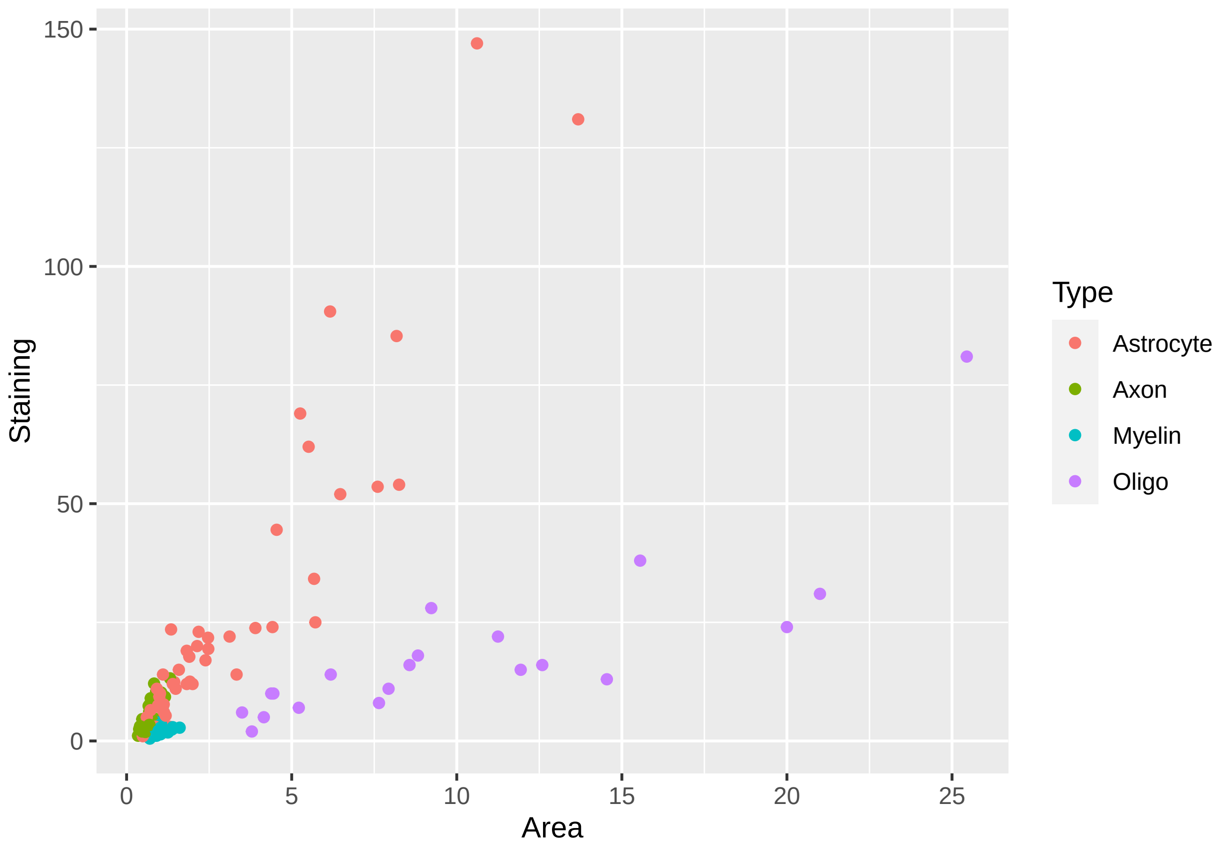
```
anova(model) # Compute ANOVA
```

```
## Type III Analysis of Variance Table with Satterthwaite’s method
##      Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
## Type 195.86  65.286     3 8.9349  19.738 0.0002785 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
emm <- emmeans(model, ~ Type) # Compute the estimated marginal means
pairs(emm) # Perform pairwise comparisons
```

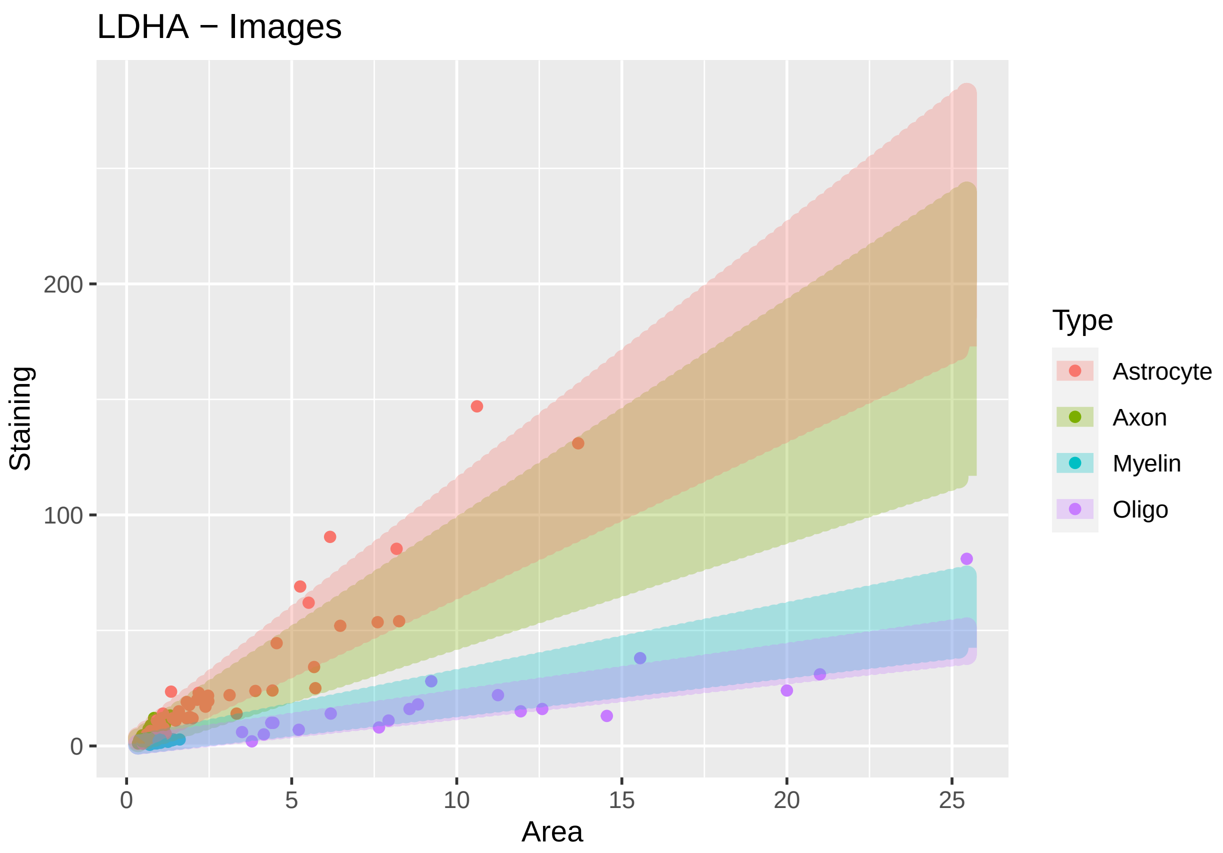
```
##  contrast           estimate    SE   df t.ratio p.value
##  Astrocyte - Axon      1.412 0.991  8.3   1.425  0.5185
##  Astrocyte - Myelin    5.809 0.991  8.3   5.861  0.0015
##  Astrocyte - Oligo     6.466 1.041 10.1   6.210  0.0005
##  Axon - Myelin         4.397 0.988  8.2   4.449  0.0087
##  Axon - Oligo          5.054 1.039 10.0   4.866  0.0031
##  Myelin - Oligo        0.657 1.039 10.0   0.632  0.9192
##
## Degrees-of-freedom method: kenward-roger
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
ggplot(data, aes(x = Area, y = Staining, color = Type)) +
  geom_point() +
  labs(x = "Area", y = "Staining", color = "Type")
```

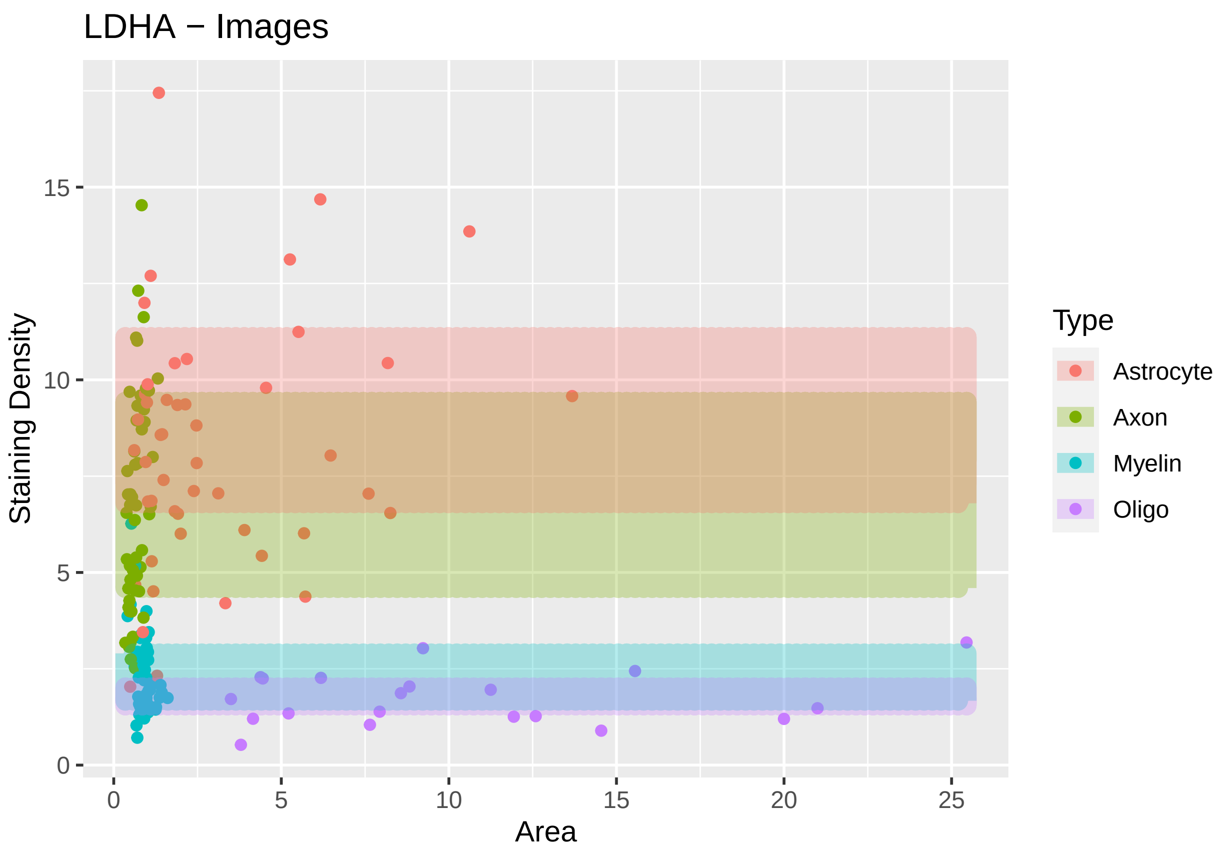


```
# Create a new dataset for predictions
prediction <- expand.grid(Area = seq(min(data$Area), max(data$Area), length.out = 100),
                          Type = unique(data$Type),
                          Animal = unique(data$Animal))

# Add predictions from the models
prediction$Staining <- predict(model, newdata = prediction) * prediction$Area
prediction$StainingDensity <- predict(model, newdata = prediction)


# Plot the data and the fitted models
ggplot(data, aes(x = Area, y = Staining, color = Type)) +
  geom_point() +
  geom_line(data = prediction, aes(x = Area, y = Staining, color = Type), linetype = "solid", size = 3.
  labs(x = "Area", y = "Staining", color = "Type", title = "LDHA - Images")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

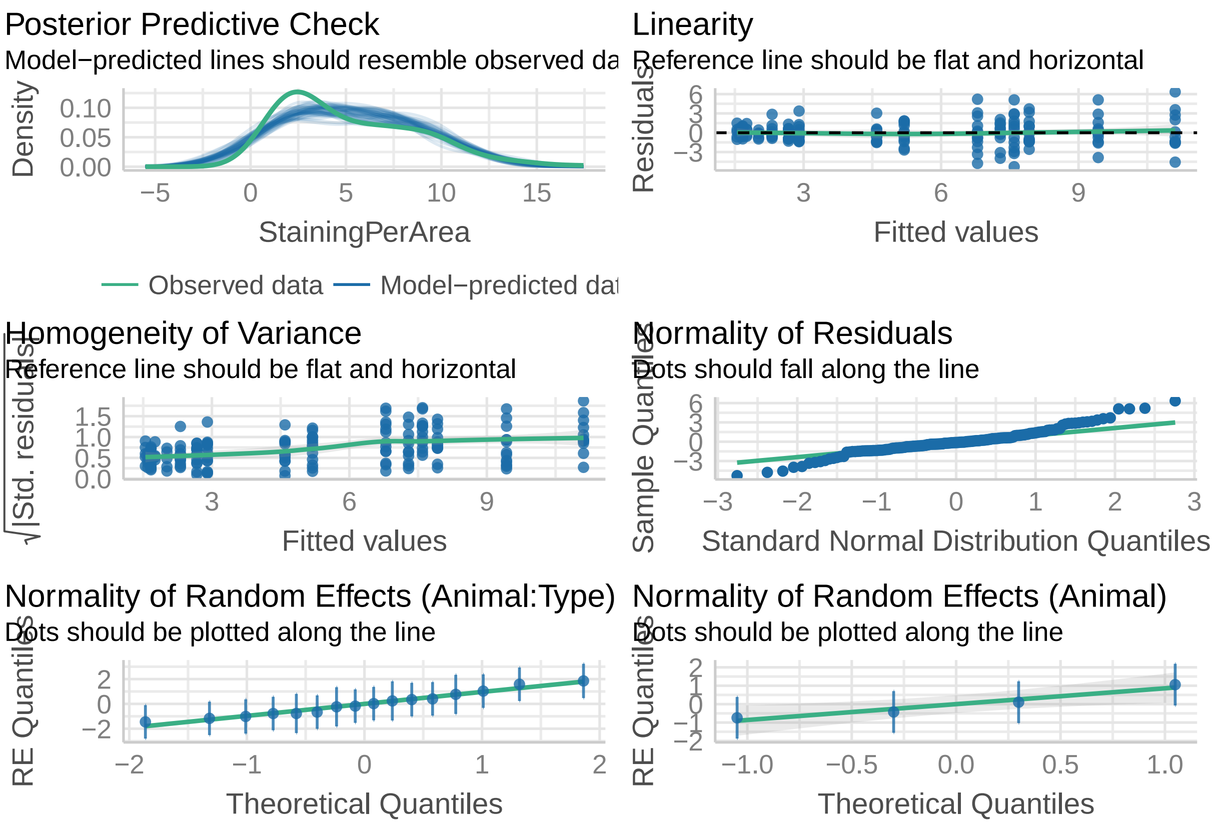


```
ggplot(data, aes(x = Area, y = StainingPerArea, color = Type)) +
  geom_point() +
  geom_line(data = prediction, aes(x = Area, y = StainingDensity, color = Type), linetype = "solid", si
  labs(x = "Area", y = "Staining Density", color = "Type", title = "LDHA - Images")
```

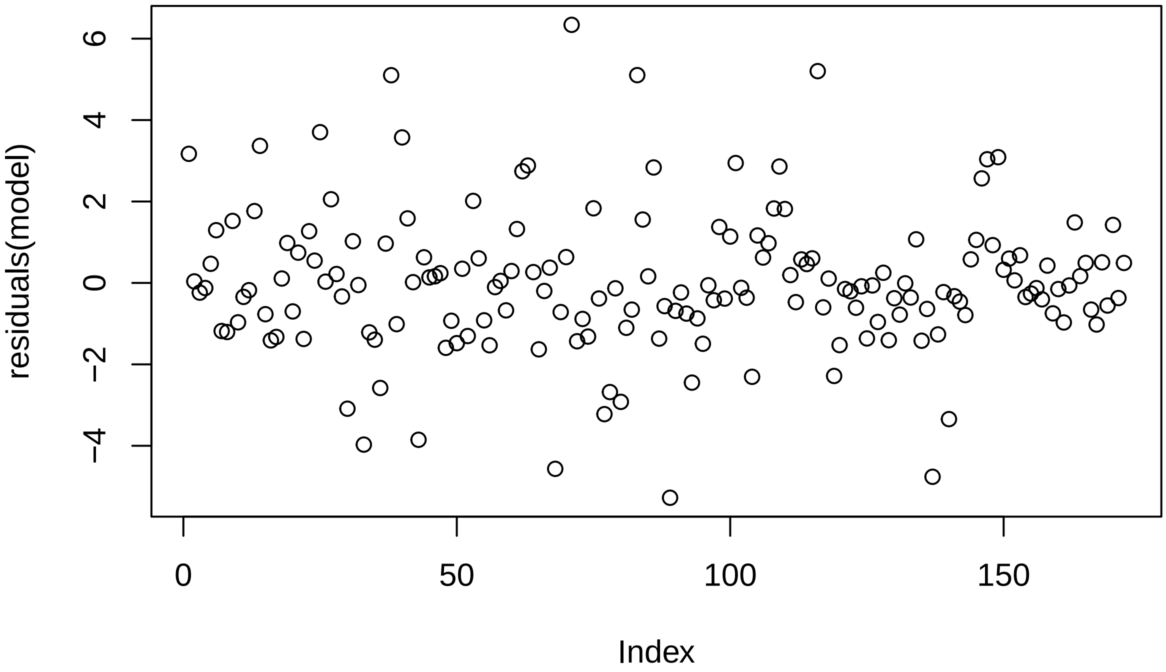


```
# Visual check of model assumptions
check_model(model)
```

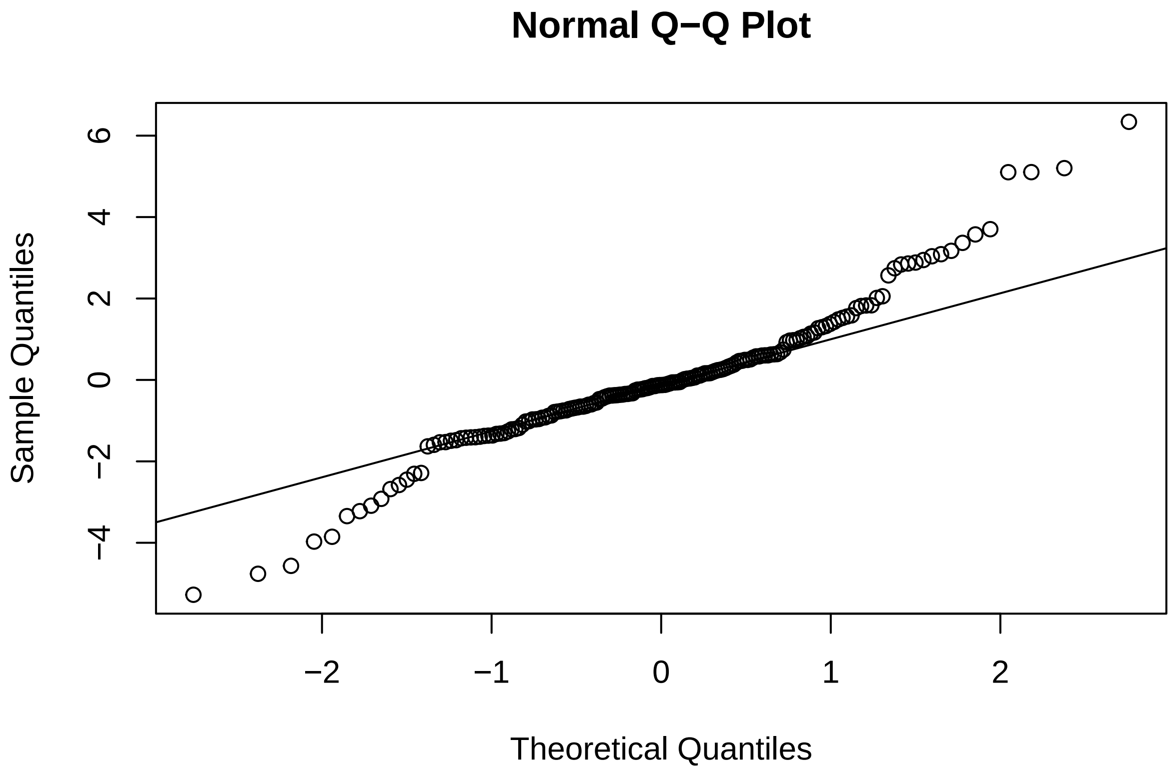
```
## Not enough model terms in the conditional part of the model to check for
##   multicollinearity.
```



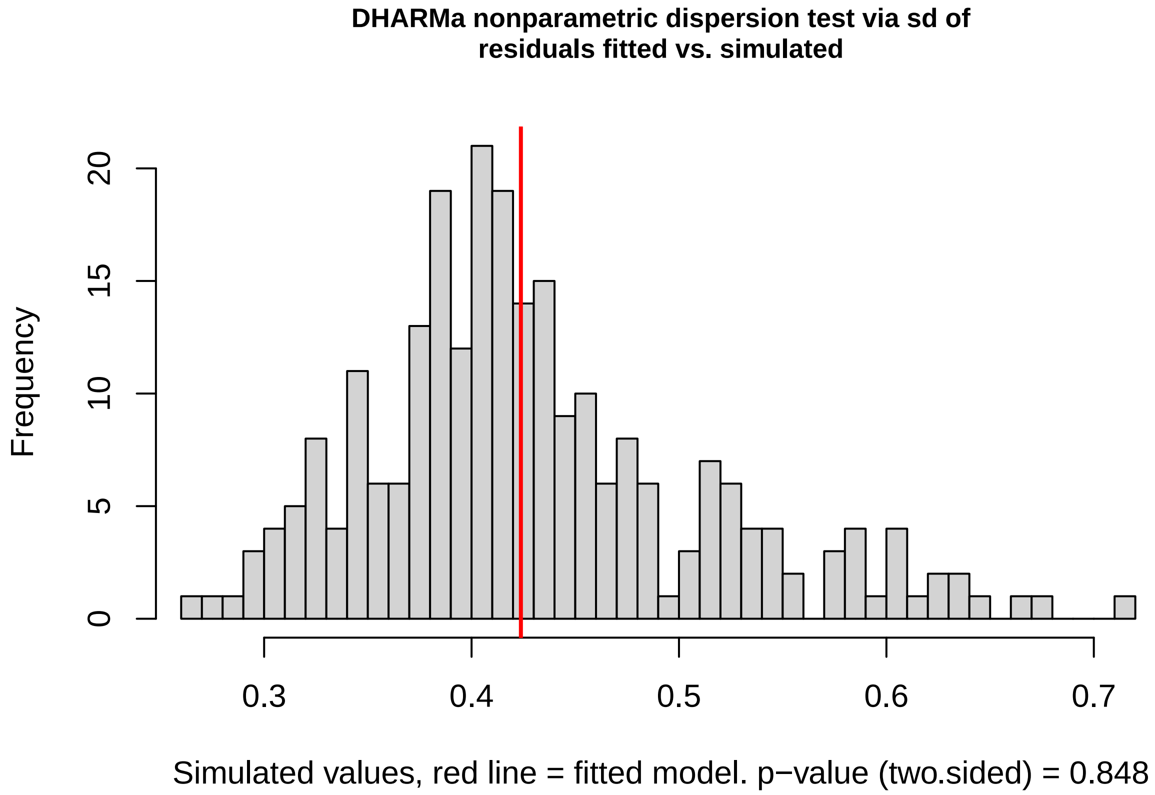
```
# Check residuals
plot(residuals(model))
```



```
qqnorm(resid(model))
qqline(resid(model))
```

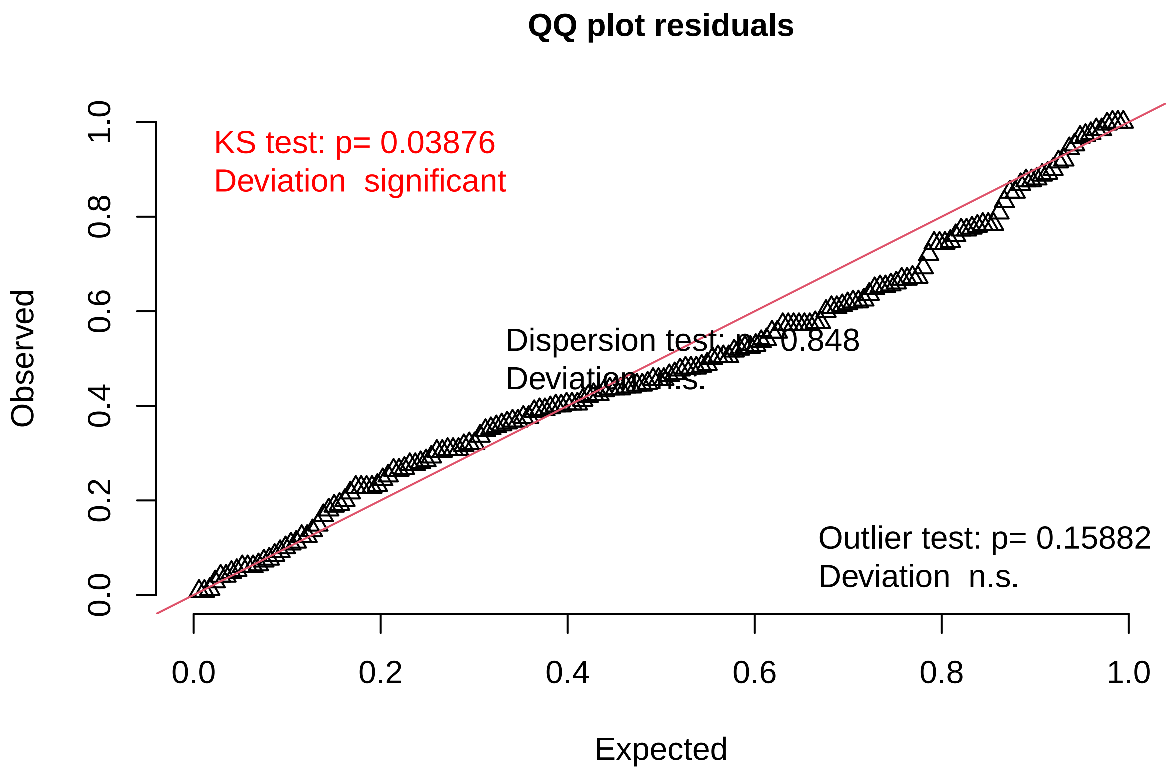


```
# Create simulated residuals
simulationOutput <- simulateResiduals(fittedModel = model)

# Check for dispersion
testDispersion(simulationOutput)
```



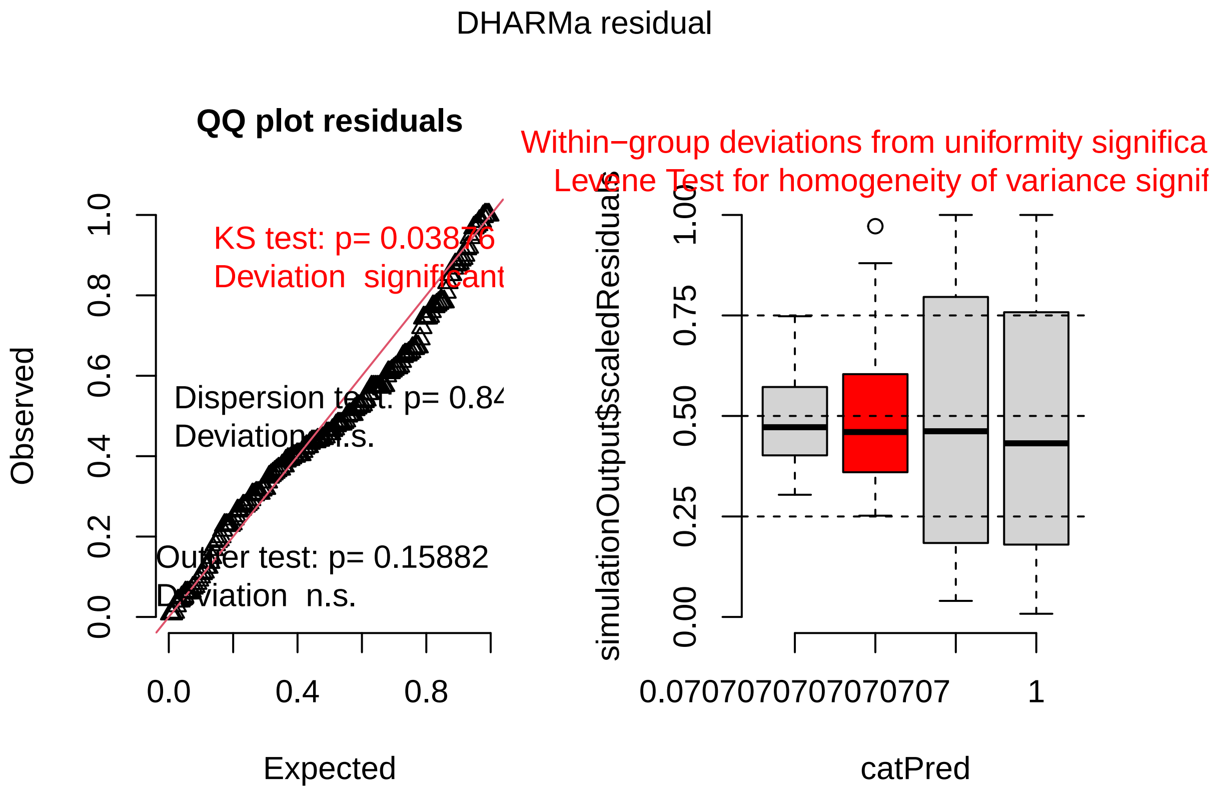
```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 0.98515, p-value = 0.848
## alternative hypothesis: two.sided
```

```
# Check for uniformity of residuals
testUniformity(simulationOutput)
```



```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  simulationOutput$scaledResiduals
## D = 0.10707, p-value = 0.03876
## alternative hypothesis: two-sided
```

```
# Create diagnostic plots
plot(simulationOutput)
```



```r
# Averages over each image
# Note: Because there is only one oligo quantified per image, no StDev needs to be
#       taken for oligodendrocytes at this level.
ldhaImages <- ldha %>%
  dplyr::select(-OligoCellBodyArea, -OligoNucleusArea, -matches("Gap")) %>%
  mutate(AxonGoldDensity = AxonGold/AxonArea,
         MyelinGoldDensity = MyelinGold/MyelinArea,
         AstrocyteGoldDensity = AstrocyteGold/AstrocyteArea,
         OligoGoldDensity = OligoGold/OligoArea,
         OligoGold = OligoGold) %>%
  group_by(Image) %>%
  summarize(Animal = first(Animal),
            Image = first(Image),
            Genotype = first(Genotype),
            across(matches("Gold|Area"),
                  list(mean = ~ mean(.,na.rm = TRUE),
                       StDev = ~ sd(., na.rm = TRUE)),
                  .names = "{.fn}_{.col}"),
            .groups = "drop") %>%
  arrange(Genotype)
ldhaImages
```

```
## # A tibble: 147 x 29
##     Image      Animal Genotype mean_OligoArea StDev_OligoArea mean_OligoGold
##     <chr>      <chr>  <chr>             <dbl>           <dbl>          <dbl>
##   1 36-0891-1  0891   ctr                 NaN              NA            NaN
##  2 36-0891-10 0891   ctr                 NaN              NA             NaN
##  3 36-0891-11 0891   ctr                 NaN              NA             NaN
##  4 36-0891-12 0891   ctr                 NaN              NA             NaN
##  5 36-0891-13 0891   ctr                 NaN              NA             NaN
##  6 36-0891-2  0891   ctr                 NaN              NA             NaN
##  7 36-0891-3  0891   ctr                 NaN              NA             NaN
##  8 36-0891-4  0891   ctr                 NaN              NA             NaN
##  9 36-0891-5  0891   ctr                 NaN              NA             NaN
## 10 36-0891-6  0891   ctr                 NaN              NA             NaN
## # i 137 more rows
## # i 23 more variables: StDev_OligoGold <dbl>, mean_AxonArea <dbl>,
## #   StDev_AxonArea <dbl>, mean_AxonGold <dbl>, StDev_AxonGold <dbl>,
## #   mean_FiberArea <dbl>, StDev_FiberArea <dbl>, mean_MyelinGold <dbl>,
## #   StDev_MyelinGold <dbl>, mean_AstrocyteGold <dbl>,
## #   StDev_AstrocyteGold <dbl>, mean_AstrocyteArea <dbl>,
## #   StDev_AstrocyteArea <dbl>, mean_MyelinArea <dbl>, ...
```

```
# Averages over each animal
# SEM for error propagation
# StDev for Oligodendrocytes (because it is the first averaging)
ldhaAnimals <- ldhaImages %>%
  group_by(Animal) %>%
  summarize(Animal = first(Animal),
            ImageNr = n(),
            Genotype = first(Genotype),
            across(matches("^mean"),
              list(mean = ~ mean(., na.rm = TRUE),
                   StDev = ~ sd(., na.rm = TRUE),
                   SEM = ~ sd(., na.rm = TRUE) / sqrt(sum(!is.na(.)))
                   ),
             .names = "{.fn}_{.col}"
            ),
            .groups = "drop")  %>%
  rename_with(~ sub("mean_mean_", "mean_", .x), .cols = starts_with("mean_mean_")) %>%
  rename_with(~ sub("StDev_mean_", "StDev_", .x), .cols = starts_with("StDev_mean_")) %>%
  rename_with(~ sub("SEM_mean_", "SEM_", .x), .cols = starts_with("SEM_mean_")) %>%
  arrange(Genotype)
ldhaAnimals
```

```
## # A tibble: 8 x 42
##   Animal ImageNr Genotype mean_OligoArea StDev_OligoArea SEM_OligoArea
##   <chr>    <int> <chr>             <dbl>           <dbl>         <dbl>
## 1 0891        18 ctr               12.5             7.87         3.52
## 2 0893        18 ctr               11.1             3.18         1.42
## 3 0895        18 ctr                7.98            4.56         2.04
## 4 0896        18 ctr                9.63            8.97         4.01
## 5 0888        21 mut                8.40            4.38         1.96
## 6 0892        19 mut                7.76            4.08         1.67
## 7 0894        18 mut                6.76            3.01         1.35
## 8 0899        17 mut                5.45            1.90         0.851
## # i 36 more variables: mean_OligoGold <dbl>, StDev_OligoGold <dbl>,
## #   SEM_OligoGold <dbl>, mean_AxonArea <dbl>, StDev_AxonArea <dbl>,
## #   SEM_AxonArea <dbl>, mean_AxonGold <dbl>, StDev_AxonGold <dbl>,
## #   SEM_AxonGold <dbl>, mean_FiberArea <dbl>, StDev_FiberArea <dbl>,
## #   SEM_FiberArea <dbl>, mean_MyelinGold <dbl>, StDev_MyelinGold <dbl>,
## #   SEM_MyelinGold <dbl>, mean_AstrocyteGold <dbl>, StDev_AstrocyteGold <dbl>,
## #   SEM_AstrocyteGold <dbl>, mean_AstrocyteArea <dbl>, ...
```

```
# Averages over each genotype
ldhaGenotype <- ldhaAnimals %>%
  group_by(Genotype) %>%
  summarize(AnimalNr = n(),
            ImageNr = sum(ImageNr),
            across(matches("^mean"),
              list(mean = ~ mean(., na.rm = TRUE),
                   StDev = ~ sd(., na.rm = TRUE),
                   SEM = ~ sd(., na.rm = TRUE) / sqrt(sum(!is.na(.)))
                   ),
              .names = "{.fn}_{.col}"
            ),
            .groups = "drop") %>%
  rename_with(~ sub("mean_mean_", "mean_", .x), .cols = starts_with("mean_mean_")) %>%
  rename_with(~ sub("StDev_mean_", "StDev_", .x), .cols = starts_with("StDev_mean_")) %>%
  rename_with(~ sub("SEM_mean_", "SEM_", .x), .cols = starts_with("SEM_mean_"))

ldhaGenotype
```

```
## # A tibble: 2 x 42
##   Genotype AnimalNr ImageNr mean_OligoArea StDev_OligoArea SEM_OligoArea
##   <chr>       <int>   <int>          <dbl>           <dbl>         <dbl>
## 1 ctr             4      72          10.3             1.95         0.974
## 2 mut             4      75           7.09            1.29         0.643
## # i 36 more variables: mean_OligoGold <dbl>, StDev_OligoGold <dbl>,
## #   SEM_OligoGold <dbl>, mean_AxonArea <dbl>, StDev_AxonArea <dbl>,
## #   SEM_AxonArea <dbl>, mean_AxonGold <dbl>, StDev_AxonGold <dbl>,
## #   SEM_AxonGold <dbl>, mean_FiberArea <dbl>, StDev_FiberArea <dbl>,
## #   SEM_FiberArea <dbl>, mean_MyelinGold <dbl>, StDev_MyelinGold <dbl>,
## #   SEM_MyelinGold <dbl>, mean_AstrocyteGold <dbl>, StDev_AstrocyteGold <dbl>,
## #   SEM_AstrocyteGold <dbl>, mean_AstrocyteArea <dbl>, ...
```

```
df_gRatio <- ldha %>%
  dplyr::select(Animal, Image, Genotype, AxonDiameter, FiberDiameter, gRatio) %>%
  filter(!is.na(FiberDiameter)) %>% # Exclude all rows that don't contain gRatio measurements
  filter(!grepl("Neg", Image)) %>% # Excludes negative controls
  mutate(gRatio = AxonDiameter/FiberDiameter) %>%
  mutate(Genotype = as.factor(Genotype),
         Animal = as.factor(Animal))
df_gRatio
```

```
## # A tibble: 1,030 x 6
## # Groups:   Animal [8]
##    Animal Image     Genotype AxonDiameter FiberDiameter gRatio
##    <fct>  <chr>     <fct>           <dbl>         <dbl>  <dbl>
##  1 0888   36-0888-1 mut             1.42          2.35   0.603
##  2 0888   36-0888-1 mut             1.06          1.66   0.635
##  3 0888   36-0888-1 mut             0.980         1.69   0.581
##  4 0888   36-0888-1 mut             1.34          1.84   0.730
##  5 0888   36-0888-1 mut             0.768         1.11   0.689
##  6 0888   36-0888-1 mut             0.451         0.789  0.571
##  7 0888   36-0888-1 mut             0.931         1.58   0.589
##  8 0888   36-0888-1 mut             0.749         1.08   0.692
##  9 0888   36-0888-1 mut             0.579         1.05   0.552
## 10 0888   36-0888-1 mut             0.627         1.05   0.596
## # i 1,020 more rows
```

```
# Fit a Linear Mixer Model
model <- lmer(gRatio ~ AxonDiameter * Genotype + (1 | Animal), data = df_gRatio)

# Predict the fiber diameter
df_gRatio$EstimatedGRatio <- predict(model, newdata = df_gRatio)

# Compute the estimated g-Ratio
df_gRatio$EstimatedFiberDiameter <- df_gRatio$AxonDiameter  / df_gRatio$EstimatedGRatio

# Results
summary(model) # Check the model summary
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite’s method [
## lmerModLmerTest]
## Formula: gRatio ~ AxonDiameter * Genotype + (1 | Animal)
##    Data: df_gRatio
##
## REML criterion at convergence: -2585.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.4113 -0.6431  0.0656  0.6896  2.9205
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Animal   (Intercept) 0.001402 0.03744
##  Residual             0.004519 0.06722
## Number of obs: 1030, groups:  Animal, 8
##
## Fixed effects:
##                            Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)               5.795e-01  2.058e-02  8.318e+00  28.160 1.51e-09 ***
## AxonDiameter              5.583e-02  9.638e-03  1.022e+03   5.793 9.22e-09 ***
## Genotypectr               4.852e-02  2.907e-02  8.282e+00   1.669   0.1324
## AxonDiameter:Genotypectr -2.829e-02  1.329e-02  1.022e+03  -2.129   0.0335 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Correlation of Fixed Effects:
##             (Intr) AxnDmt Gntypc
## AxonDiametr -0.389
## Genotypectr -0.708  0.276
## AxnDmtr:Gnt  0.282 -0.725 -0.387
```

```
anova(model) # Compute ANOVA
```

```
## Type III Analysis of Variance Table with Satterthwaite’s method
##                          Sum Sq  Mean Sq NumDF   DenDF F value    Pr(>F)    
## AxonDiameter          0.177915 0.177915     1 1022.41 39.3701 5.169e-10 ***
## Genotype              0.012586 0.012586     1    8.28  2.7851   0.13241    
## AxonDiameter:Genotype 0.020488 0.020488     1 1022.41  4.5338   0.03347 *  
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
# Compute the estimated marginal means for the interaction
emm_int <- emmeans(model, ~ Genotype:AxonDiameter)
summary(emm_int)
```

```
##  Genotype AxonDiameter emmean    SE   df lower.CL upper.CL
##  mut             0.849  0.627 0.019 6.00    0.581    0.673
##  ctr             0.849  0.651 0.019 5.99    0.605    0.698
## 
## Degrees-of-freedom method: kenward-roger 
## Confidence level used: 0.95
```

```
pairs(emm_int) # Perform pairwise comparisons
```

```
##  contrast                                                              estimate
##  mut AxonDiameter0.848593945928665 - ctr AxonDiameter0.848593945928665  -0.0245
##      SE df t.ratio p.value
##  0.0268  6  -0.914  0.3958
## 
## Degrees-of-freedom method: kenward-roger
```

```
# Test the fit
AIC(model)
```

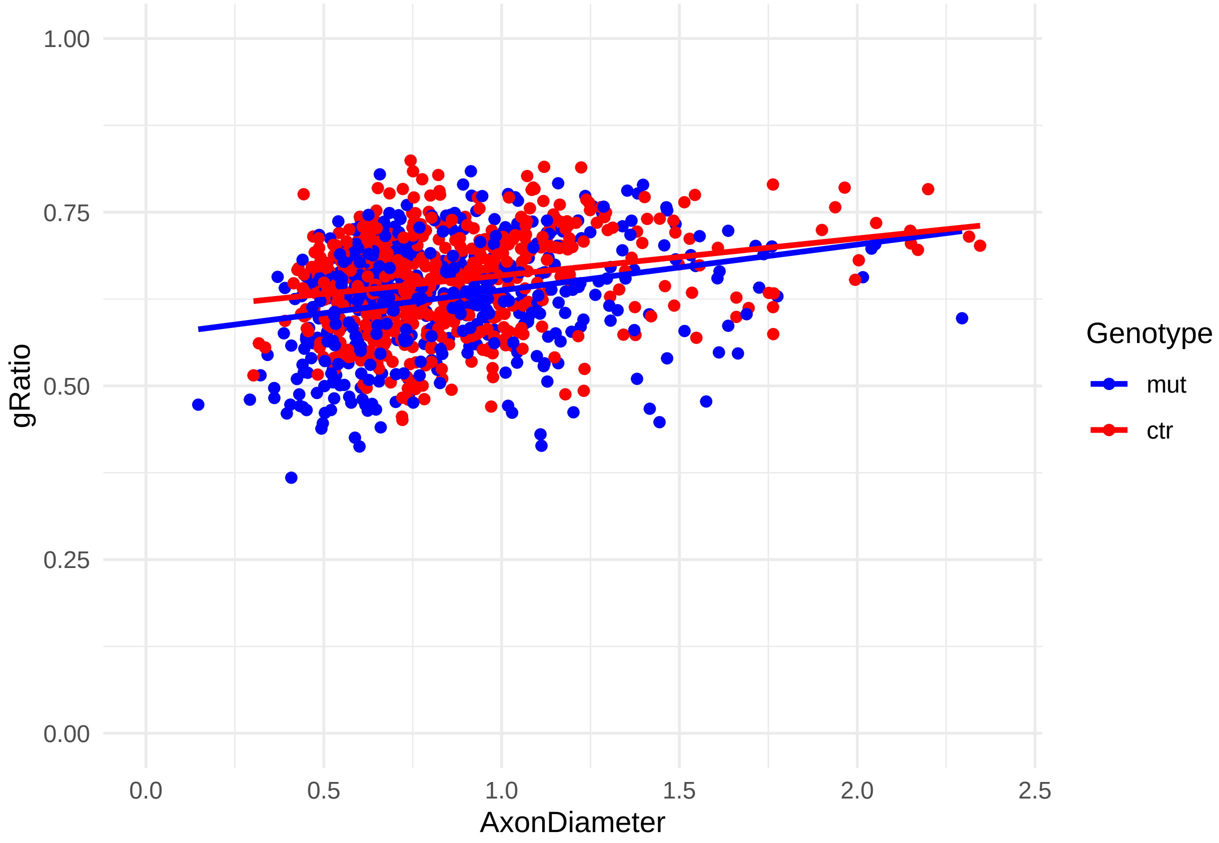
```
## [1] -2573.551
```

```
best_model <- lmerTest::step(model, direction="both")
summary(best_model)
```

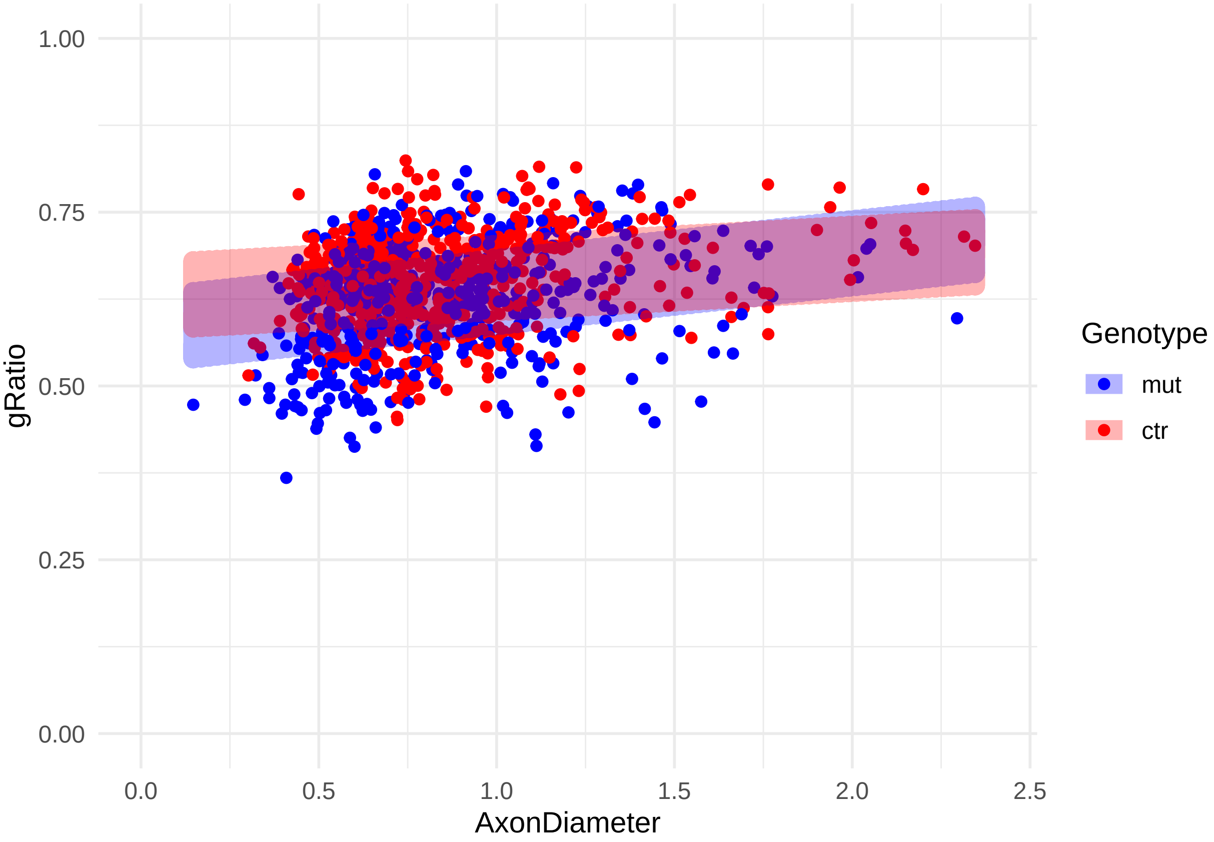
```
##        Length Class Mode
## random 7      anova list
## fixed  7      anova list
```

```
data <- df_gRatio
ggplot(data, aes(x = AxonDiameter, y = gRatio, color = Genotype)) +
  geom_point() +
  geom_smooth(method = "lm", aes(group = Genotype), se = FALSE) +
  labs(x = "AxonDiameter", y = "gRatio", color = "Genotype") +
  scale_color_manual(values = c("ctr" = "red", "mut" = "blue")) +
  xlim(0, 2.4) +
  ylim(0, 1) +
  theme_minimal()
```

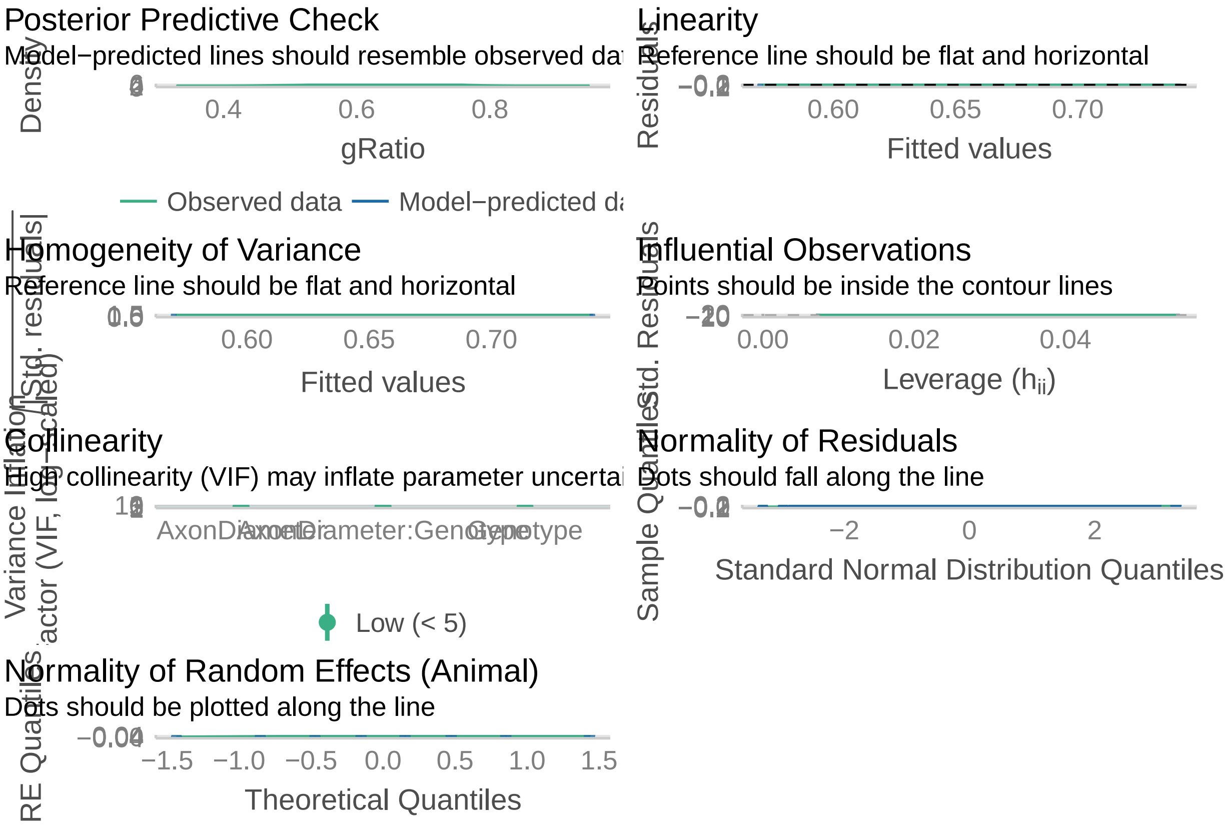
```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



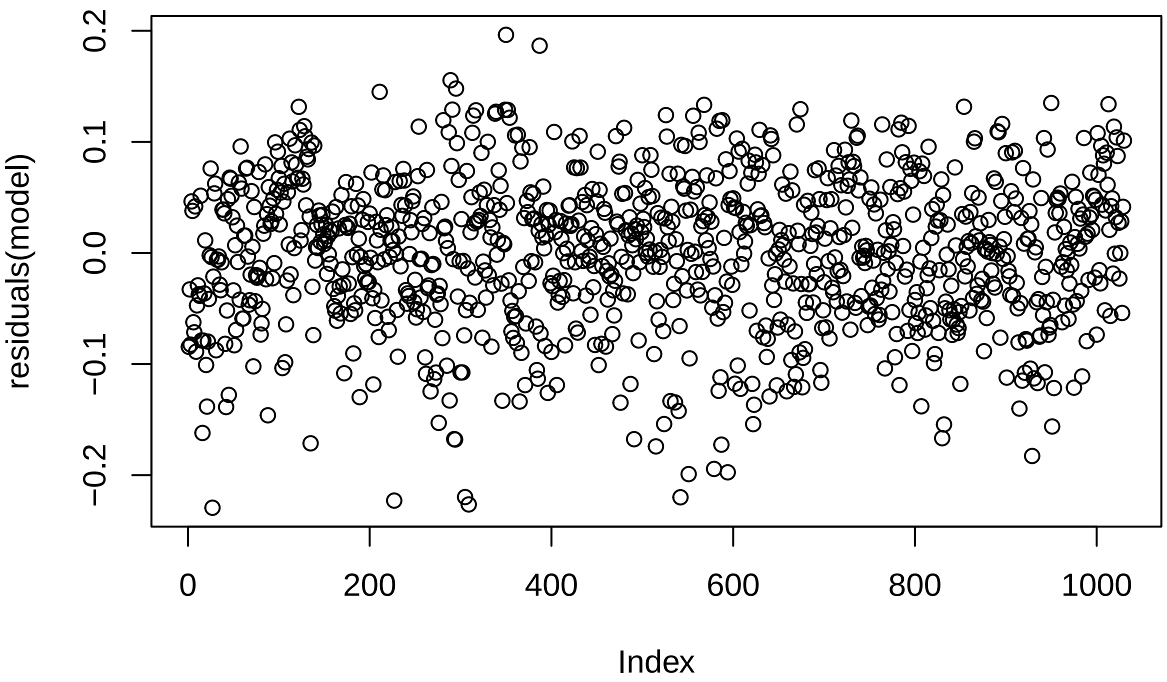
```
# Create a new dataset for predictions
prediction <- expand.grid(AxonDiameter = seq(min(data$AxonDiameter), max(data$AxonDiameter), length.out
                          Animal = unique(data$Animal),
                          Genotype = unique(data$Genotype))

# Add predictions from the models
prediction$Pred_Model <- predict(model, newdata = prediction)
prediction$gRatio <- prediction$Pred_Model

# Plot the data and the fitted models
ggplot(data, aes(x = AxonDiameter, y = gRatio, color = Genotype)) +
  geom_point() +
  geom_line(data = prediction, aes(y = gRatio), linetype = "solid", size = 3.5, alpha = 0.3) +
  labs(x = "AxonDiameter", y = "gRatio", color = "Genotype") +
  xlim(0, 2.4) +
  ylim(0, 1) +
  scale_color_manual(values = c("ctr" = "red", "mut" = "blue")) +
  theme_minimal()
```



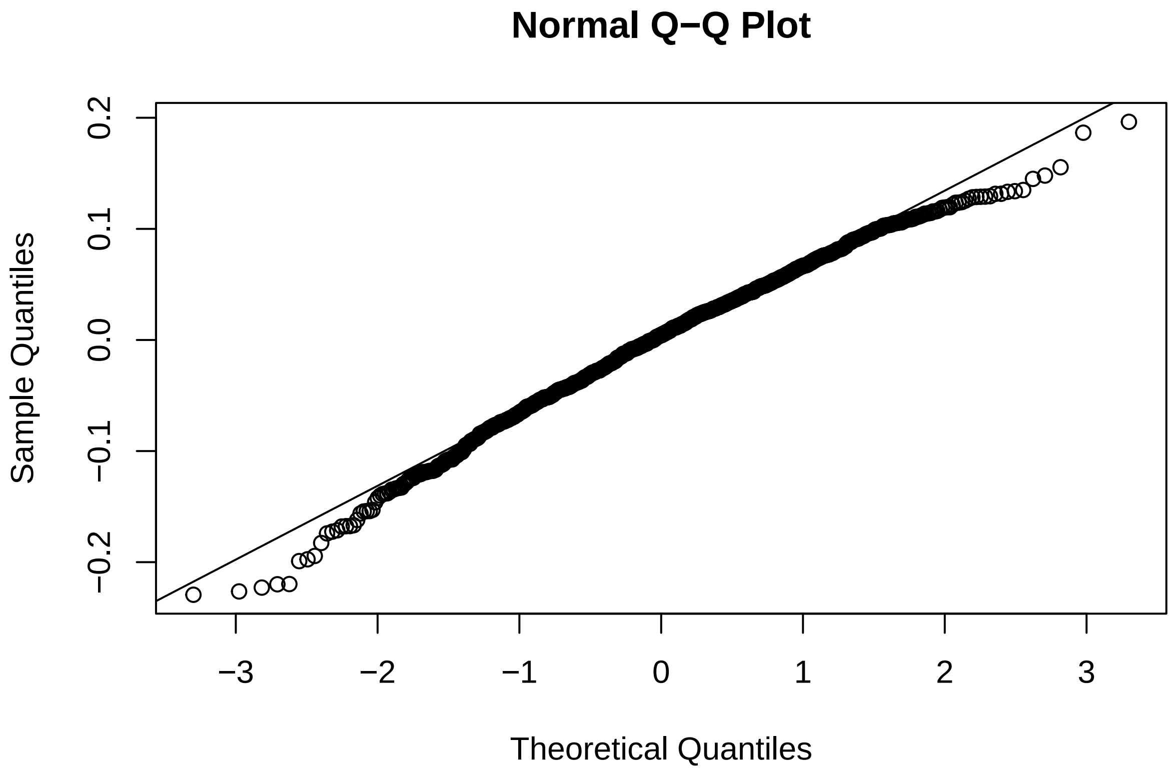
```
# Visual check of model assumptions
check_model(model)
```



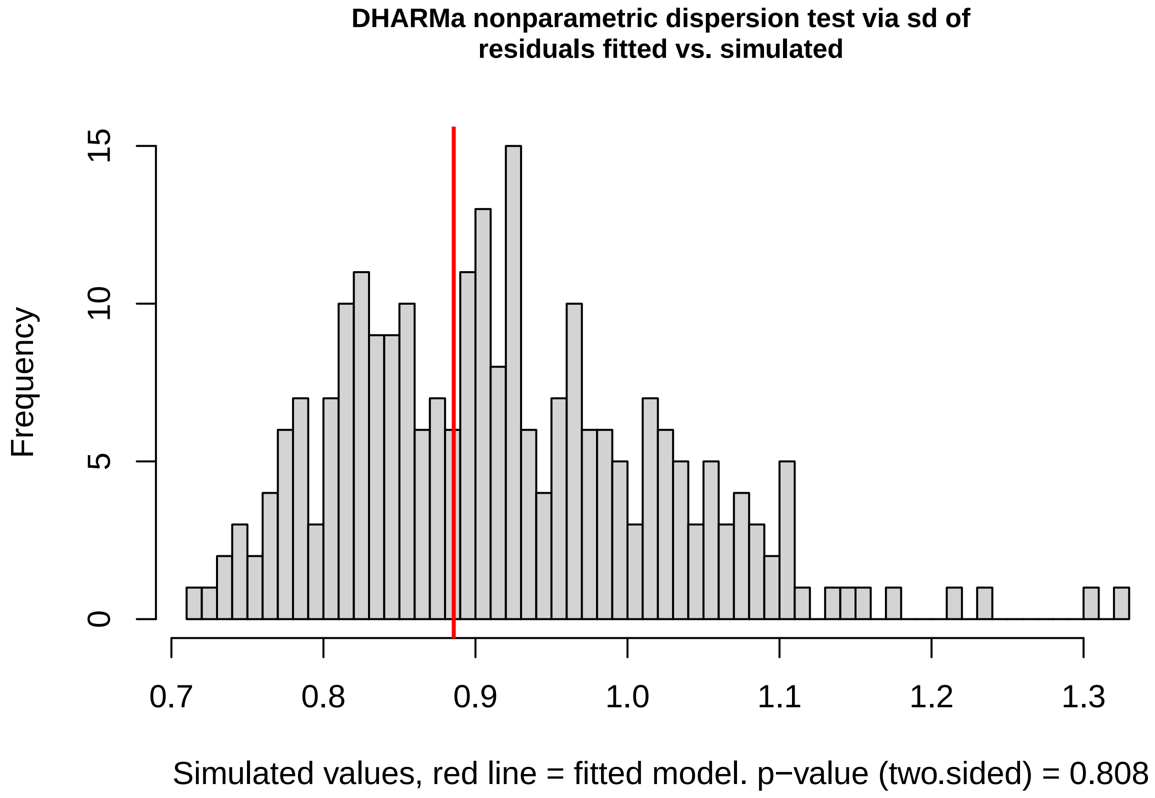
```
# Check residuals
plot(residuals(model))
```



```
qqnorm(resid(model))
qqline(resid(model))
```

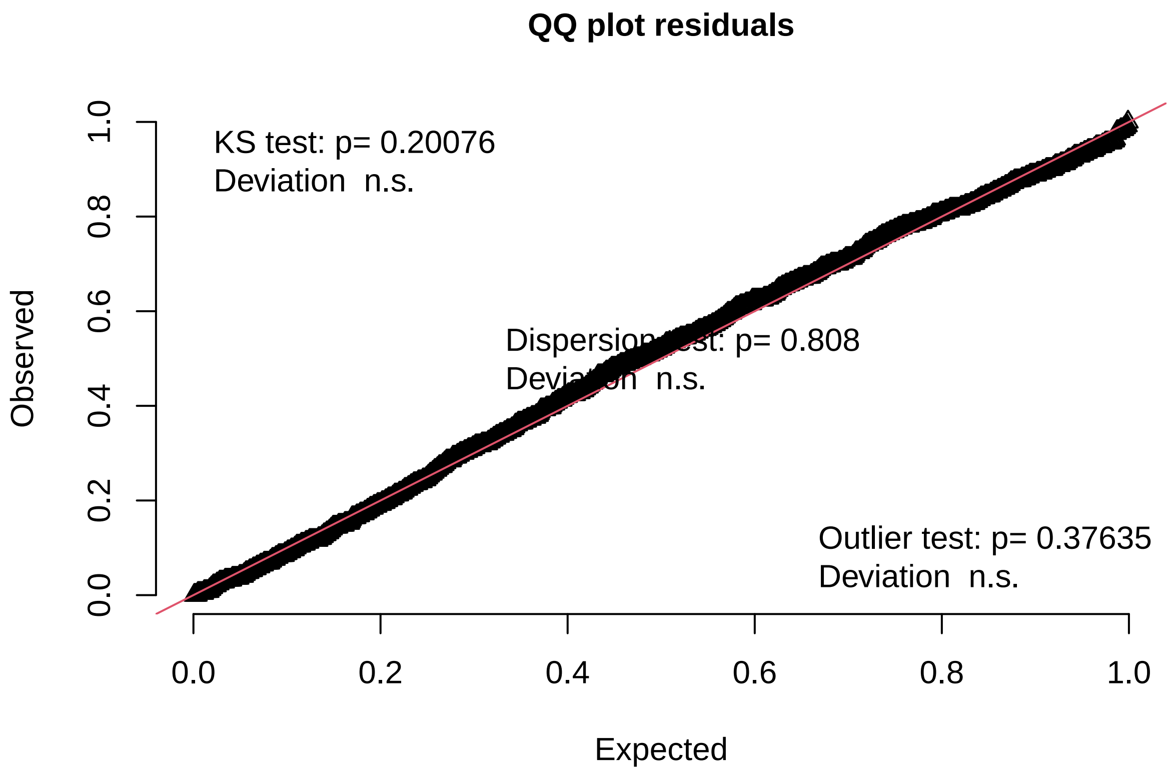


```
# Create simulated residuals
simulationOutput <- simulateResiduals(fittedModel = model)

# Check for dispersion
testDispersion(simulationOutput)
```



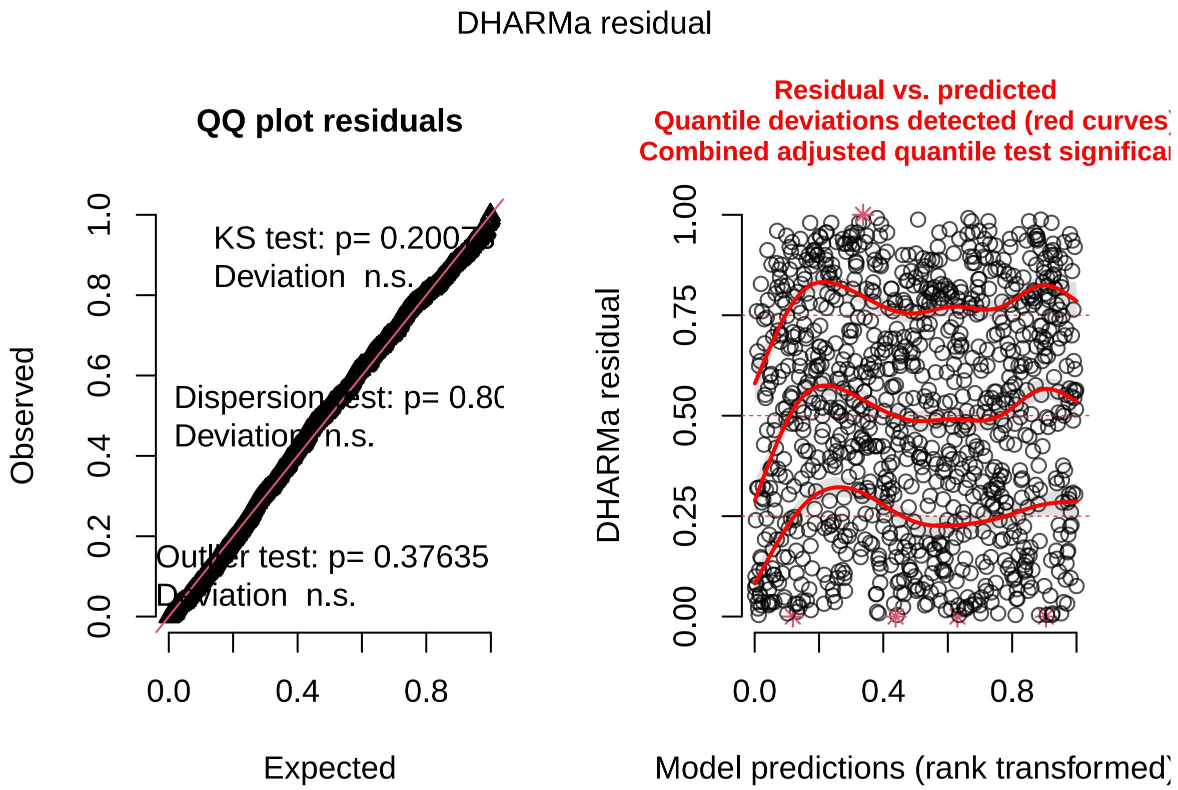
```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 0.96358, p-value = 0.808
## alternative hypothesis: two.sided
```

```
# Check for uniformity of residuals
testUniformity(simulationOutput)
```



```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  simulationOutput$scaledResiduals
## D = 0.033398, p-value = 0.2008
## alternative hypothesis: two-sided
```

```
# Create diagnostic plots
plot(simulationOutput)
```



```r
#Imports excel file
rnaFile = here::here("LDHAB_RNAscope_Quantification.xlsx")

ldha_Rna <- read_excel(path = rnaFile, sheet = "LDHA") %>%
  mutate(CellID = row_number()) %>%
  pivot_longer(
    cols = -CellID, # Exclude CellID from the reshaping
    names_to = c("Genotype", "Animal"), # New column names
    names_pattern = "(wt|KO) (.*)" # Regular expression to match and separate genotype and animal numbe
  ) %>%
  rename(RnaDots = value)
ldha_Rna$Genotype[which(ldha_Rna$Genotype == "KO")] <- "mut"


ldha_Rna <- ldha_Rna %>%
  mutate(Animal = paste(Genotype, Animal, sep = " - "))

# Convert Genotype to factor
ldha_Rna$Genotype <- as.factor(ldha_Rna$Genotype)

# Negative Binomial GLMM
model <- glmer.nb(RnaDots ~ Genotype + (1|Animal), data = ldha_Rna)

# Results Cell Type
summary(model) # Check the model summary
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: Negative Binomial(15.8185)  ( log )
## Formula: RnaDots ~ Genotype + (1 | Animal)
##    Data: ldha_Rna
##
##      AIC      BIC   logLik deviance df.resid
##  22334.7  22361.2 -11163.3  22326.7     5518
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.7077 -0.6486 -0.0872  0.5307  6.0643
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  Animal (Intercept) 0.004419 0.06647
## Number of obs: 5522, groups:  Animal, 6
##
## Fixed effects:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.17682    0.04064  28.956   <2e-16 ***
## Genotypewt  -0.05823    0.05693  -1.023    0.306
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Correlation of Fixed Effects:
##            (Intr)
## Genotypewt -0.714
```

```
anova(model) # Compute ANOVA
```

```
## Analysis of Variance Table
##          npar Sum Sq Mean Sq F value
## Genotype    1 1.0466  1.0466  1.0466
```

```
# Obtain the EMMs
emm <- emmeans(model, ~ Genotype)
pairs(emm) # Perform pairwise comparisons
```

```
##  contrast estimate     SE  df z.ratio p.value
##  mut - wt   0.0582 0.0569 Inf   1.023  0.3064
##
## Results are given on the log (not the response) scale.
```

```
# Transform the estimates
emm_exp <- regrid(emm, transform = "response")

# Print the results
summary(emm_exp)
```

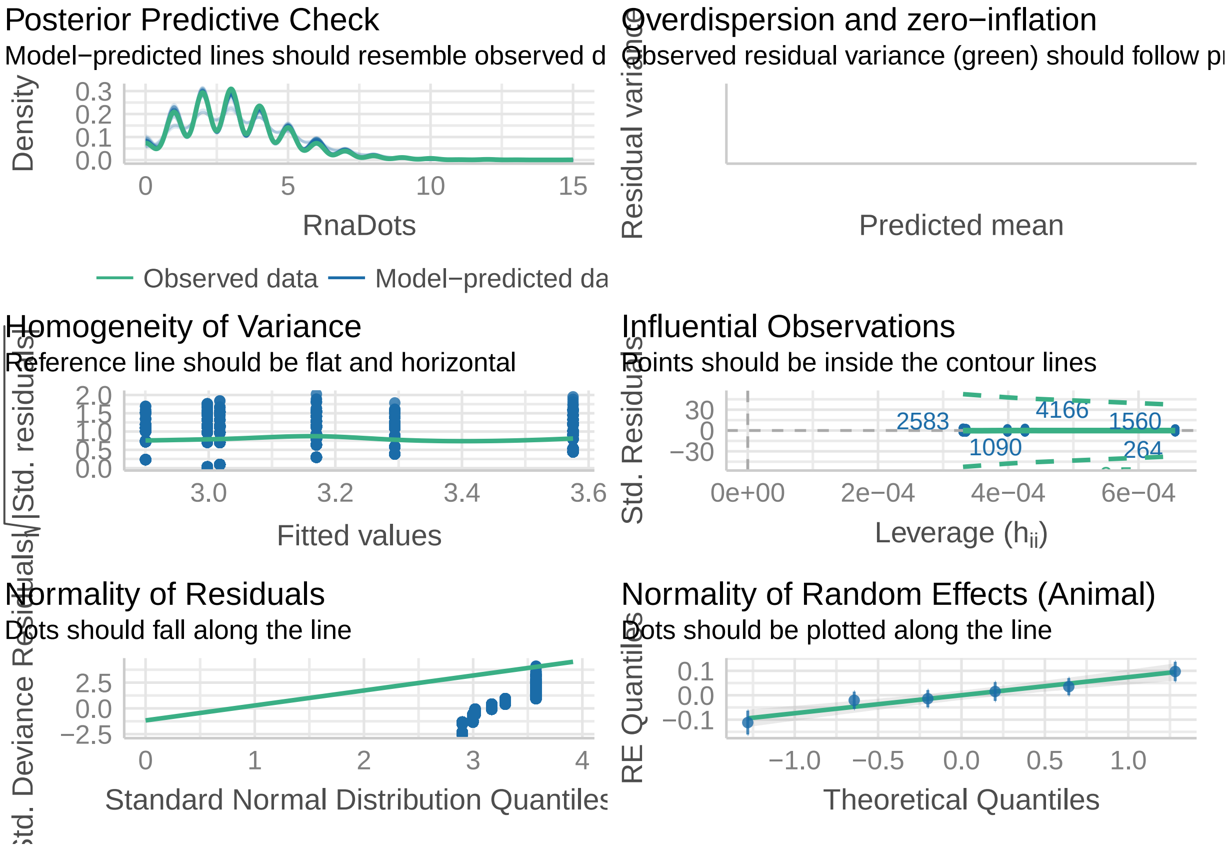
```
##  Genotype response    SE  df asymp.LCL asymp.UCL
##  mut          3.24 0.132 Inf      2.99       3.5
##  wt           3.06 0.122 Inf      2.82      3.3
##
## Confidence level used: 0.95
```

```
# Calculate contrasts
contrasts <- contrast(emm_exp, method = "pairwise")

# Print the results
summary(contrasts)
```

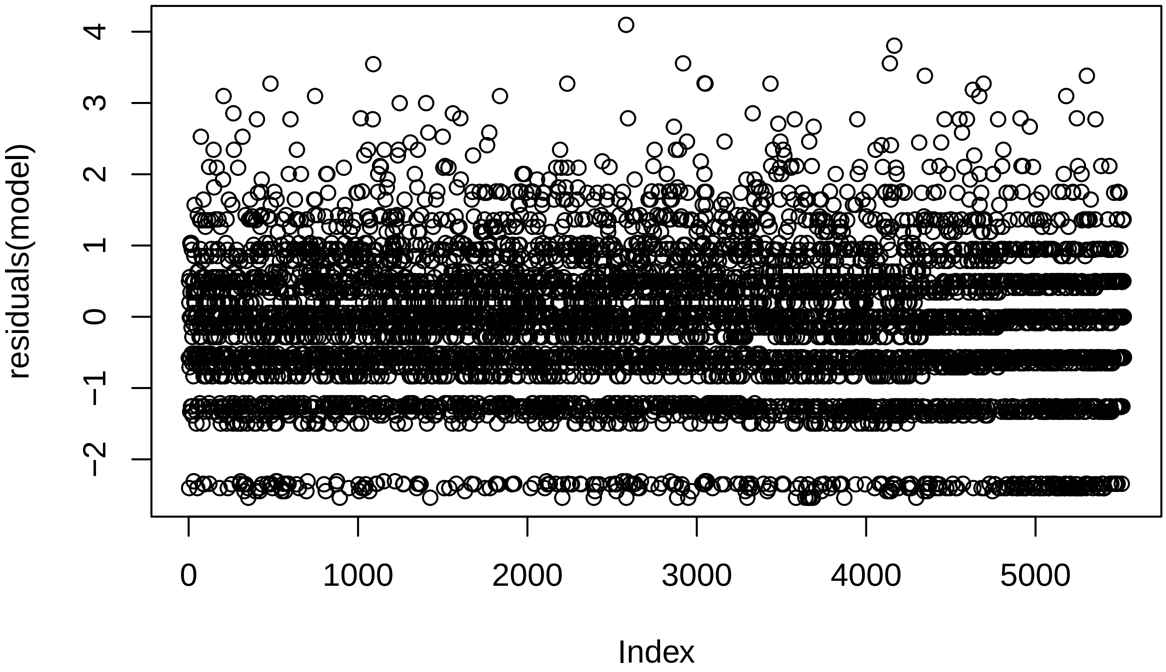
```
##  contrast estimate   SE  df z.ratio p.value
##  mut - wt    0.183 0.18 Inf   1.021  0.3070
```

```
# Visual check of model assumptions
check_model(model)
```

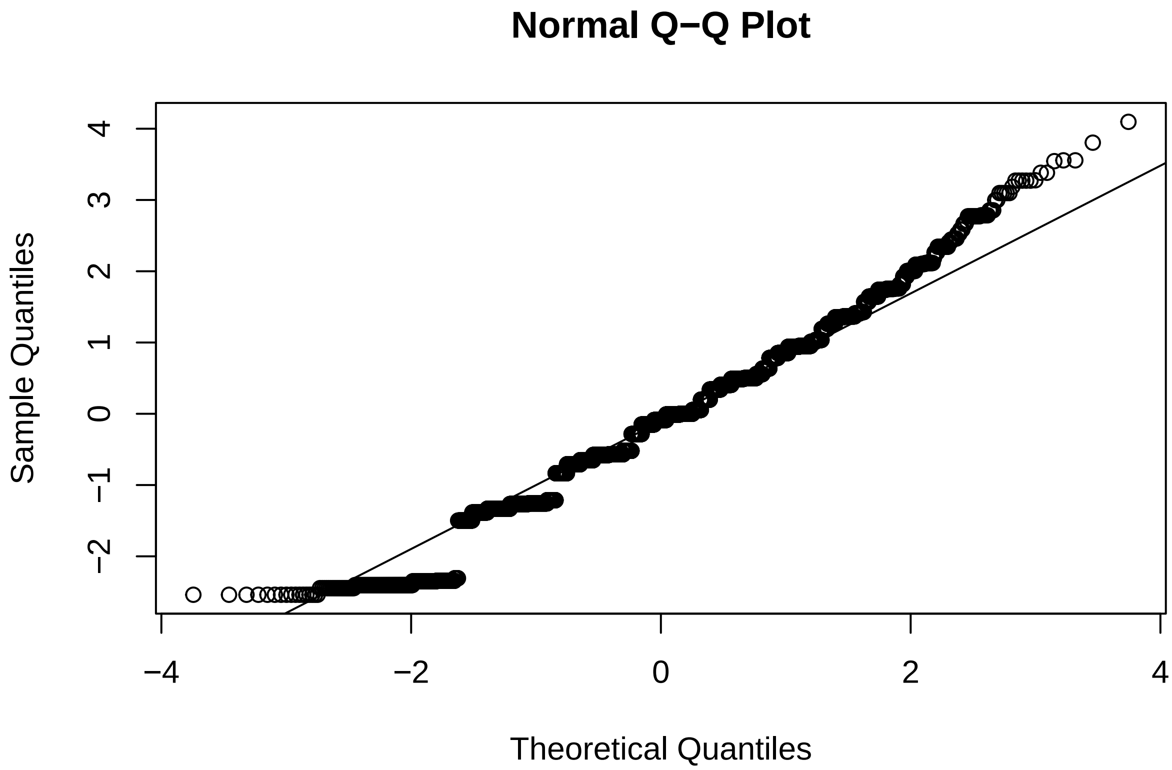
```
## Not enough model terms in the conditional part of the model to check for
##   multicollinearity.
```



```
# Check residuals
plot(residuals(model))
```

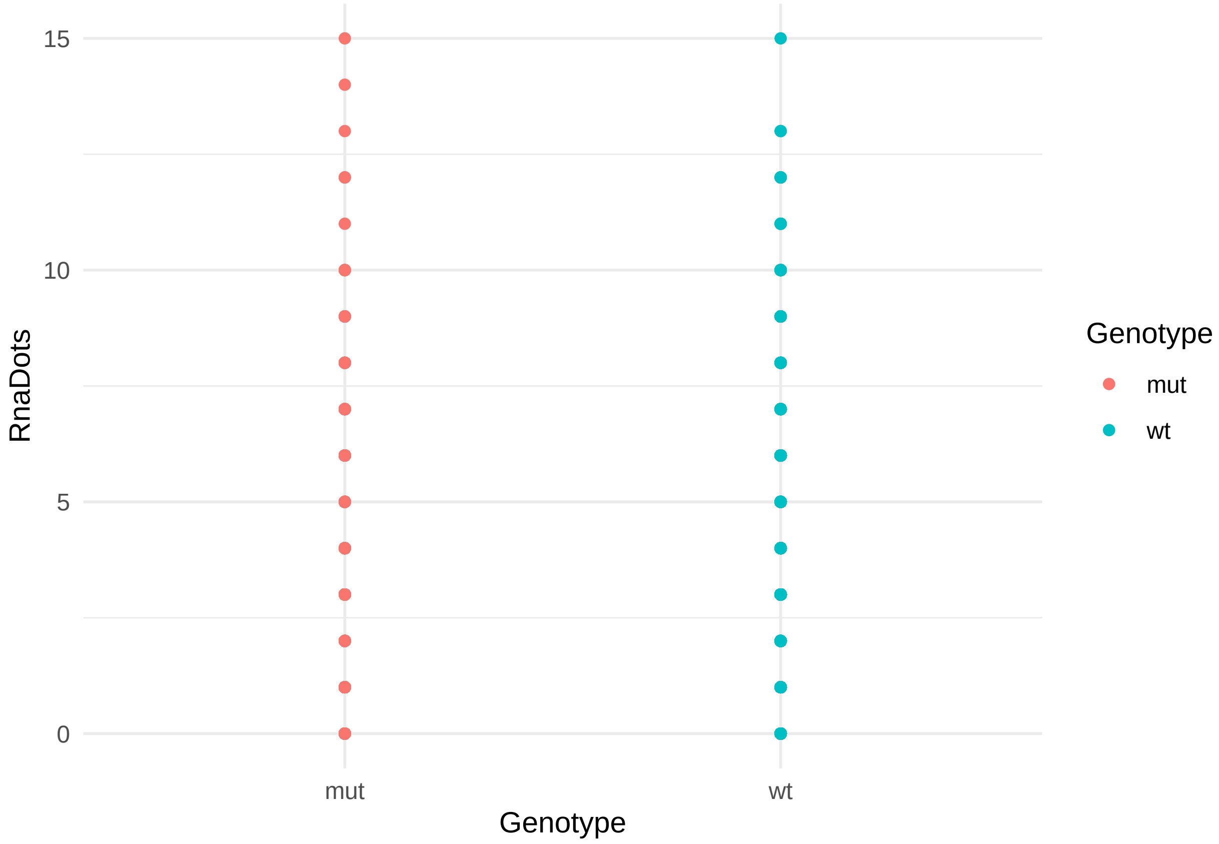


```
qqnorm(resid(model))
qqline(resid(model))
```



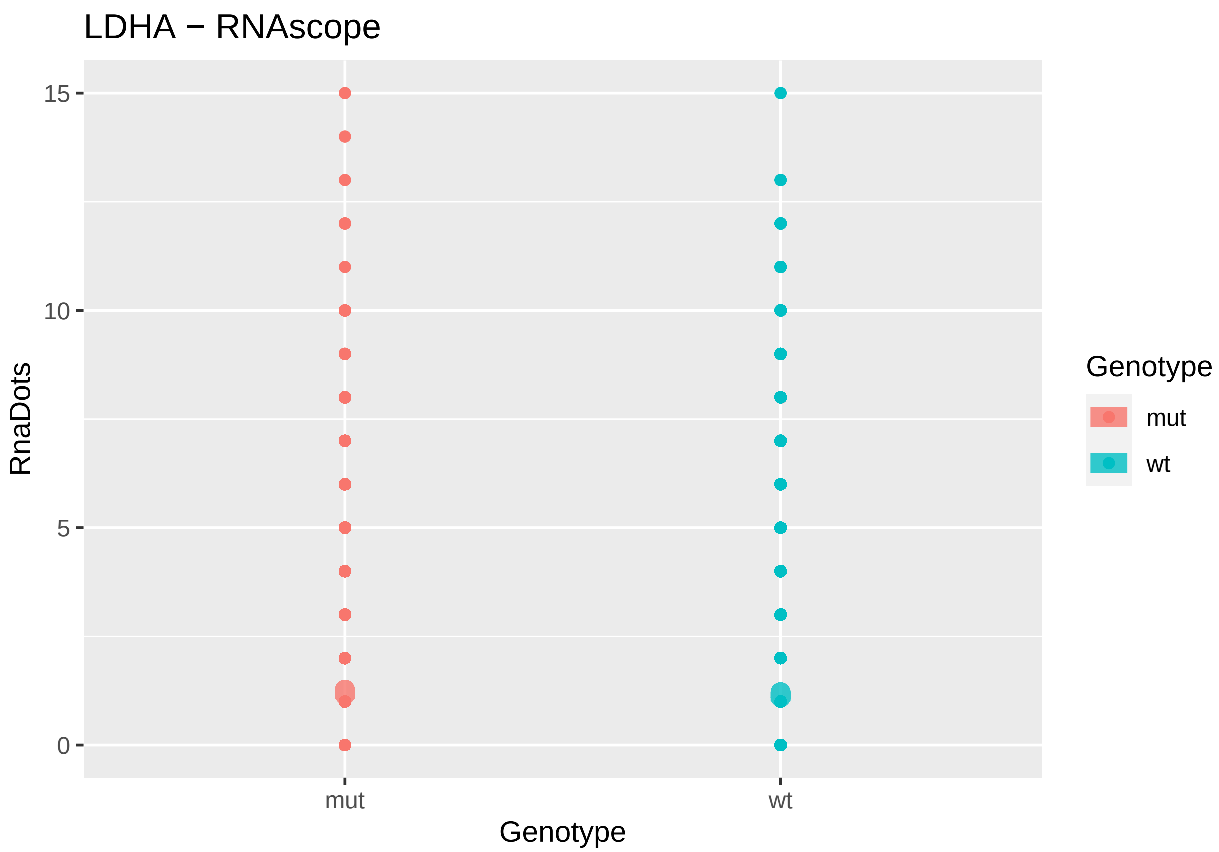
```
data <- ldha_Rna
ggplot(data, aes(x = Genotype, y = RnaDots, color = Genotype)) +
  geom_point() +
  labs(x = "Genotype", y = "RnaDots", color = "Genotype") +
  theme_minimal()
```

```
## Warning: Removed 1228 rows containing missing values (‘geom_point()‘).
```



```r
# Create a new dataset for predictions
prediction <- expand.grid(Genotype = unique(data$Genotype),
                          Animal = unique(data$Animal))

# Add predictions from the models
prediction$Pred_Model <- predict(model, newdata = prediction)

# Plot the data and the fitted models
ggplot(data, aes(x = Genotype, y = RnaDots, color = Genotype)) +
  geom_point() +
  geom_line(data = prediction, aes(x = Genotype, y = Pred_Model, color = Genotype), linetype = "solid",
  labs(x = "Genotype", y = "RnaDots", color = "Genotype", title = "LDHA - RNAscope")
```

```
## Warning: Removed 1228 rows containing missing values (‘geom_point()‘).
```



```
ldhb_Rna <- read_excel(path = rnaFile, sheet = "LDHB") %>%
  mutate(CellID = row_number()) %>%
  pivot_longer(
    cols = -CellID, # Exclude CellID from the reshaping
    names_to = c("Genotype", "Animal"), # New column names
    names_pattern = "(wt|KO) (.*)" # Regular expression to match and separate genotype and animal numbe
  ) %>%
  rename(RnaDots = value)
ldhb_Rna$Genotype[which(ldhb_Rna$Genotype == "KO")] <- "mut"


ldhb_Rna <- ldhb_Rna %>%
  mutate(Animal = paste(Genotype, Animal, sep = " - "))

# Convert Genotype to factor
ldhb_Rna$Genotype <- as.factor(ldhb_Rna$Genotype) %>%
  relevel(ref = "wt")

# Negative Binomial GLMM
model <- glmer.nb(RnaDots ~ Genotype + (1|Animal), data = ldhb_Rna)

# Results Cell Type
summary(model) # Check the model summary
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: Negative Binomial(7.9582)  ( log )
## Formula: RnaDots ~ Genotype + (1 | Animal)
##    Data: ldhb_Rna
##
##      AIC      BIC   logLik deviance df.resid
##  24386.1  24412.6 -12189.1  24378.1     5518
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.7374 -0.6801 -0.1744  0.5614  6.2018
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  Animal (Intercept) 0.02378  0.1542
## Number of obs: 5522, groups:  Animal, 6
##
## Fixed effects:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.34241    0.08966  14.973   <2e-16 ***
## Genotypemut -0.03672    0.12708  -0.289    0.773
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## Genotypemut -0.705
```

```
anova(model) # Compute ANOVA
```

```
## Analysis of Variance Table
##          npar   Sum Sq  Mean Sq F value
## Genotype    1 0.083463 0.083463  0.0835
```

```
# Obtain the EMMs
emm <- emmeans(model, ~ Genotype)
pairs(emm) # Perform pairwise comparisons
```

```
##  contrast estimate    SE  df z.ratio p.value
##  wt - mut   0.0367 0.127 Inf   0.289  0.7726
##
## Results are given on the log (not the response) scale.
```

```
# Transform the estimates
emm_exp <- regrid(emm, transform = "response")

# Print the results
summary(emm_exp)
```

```
##  Genotype response    SE  df asymp.LCL asymp.UCL
##  wt           3.83 0.343 Inf      3.16      4.50
##  mut          3.69 0.332 Inf      3.04      4.34
```

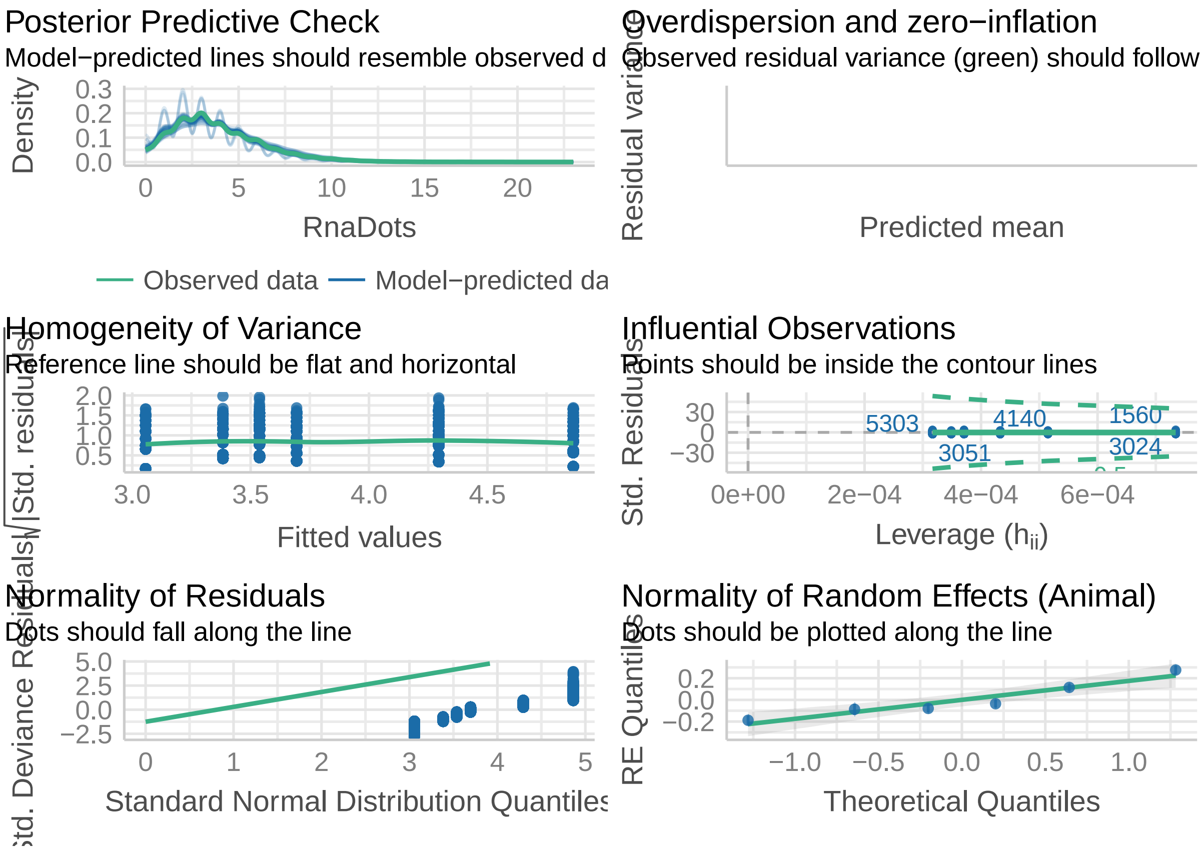
```
##
## Confidence level used: 0.95
```

```
# Calculate contrasts
contrasts <- contrast(emm_exp, method = "pairwise")

# Print the results
summary(contrasts)
```

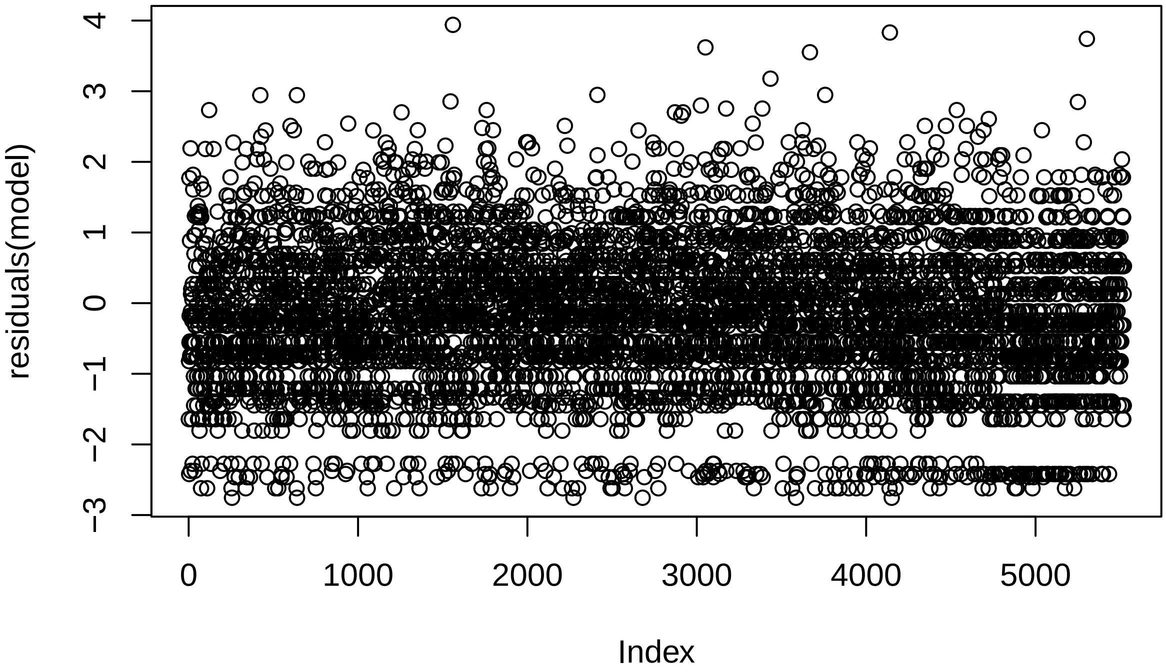
```
##  contrast estimate    SE  df z.ratio p.value
##  wt - mut    0.138 0.478 Inf   0.289  0.7726
```

```
# Visual check of model assumptions
check_model(model)
```

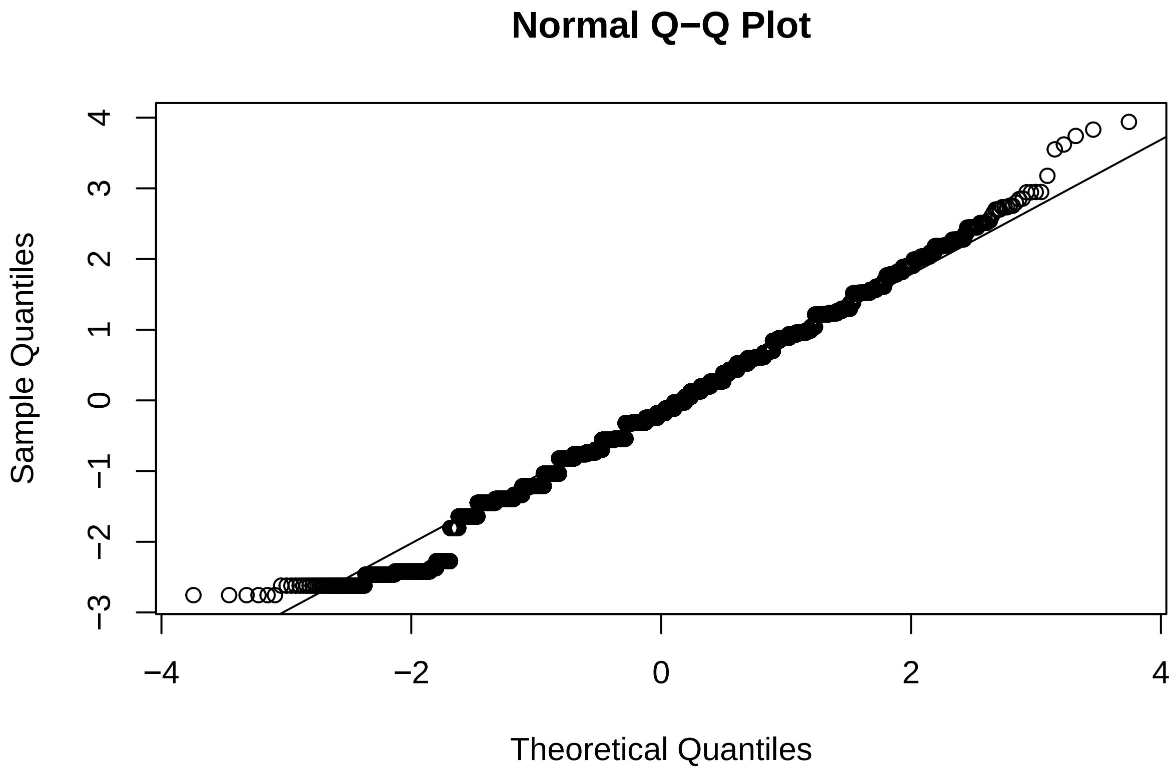
```
## Not enough model terms in the conditional part of the model to check for
##   multicollinearity.
```



```
# Check residuals
plot(residuals(model))
```

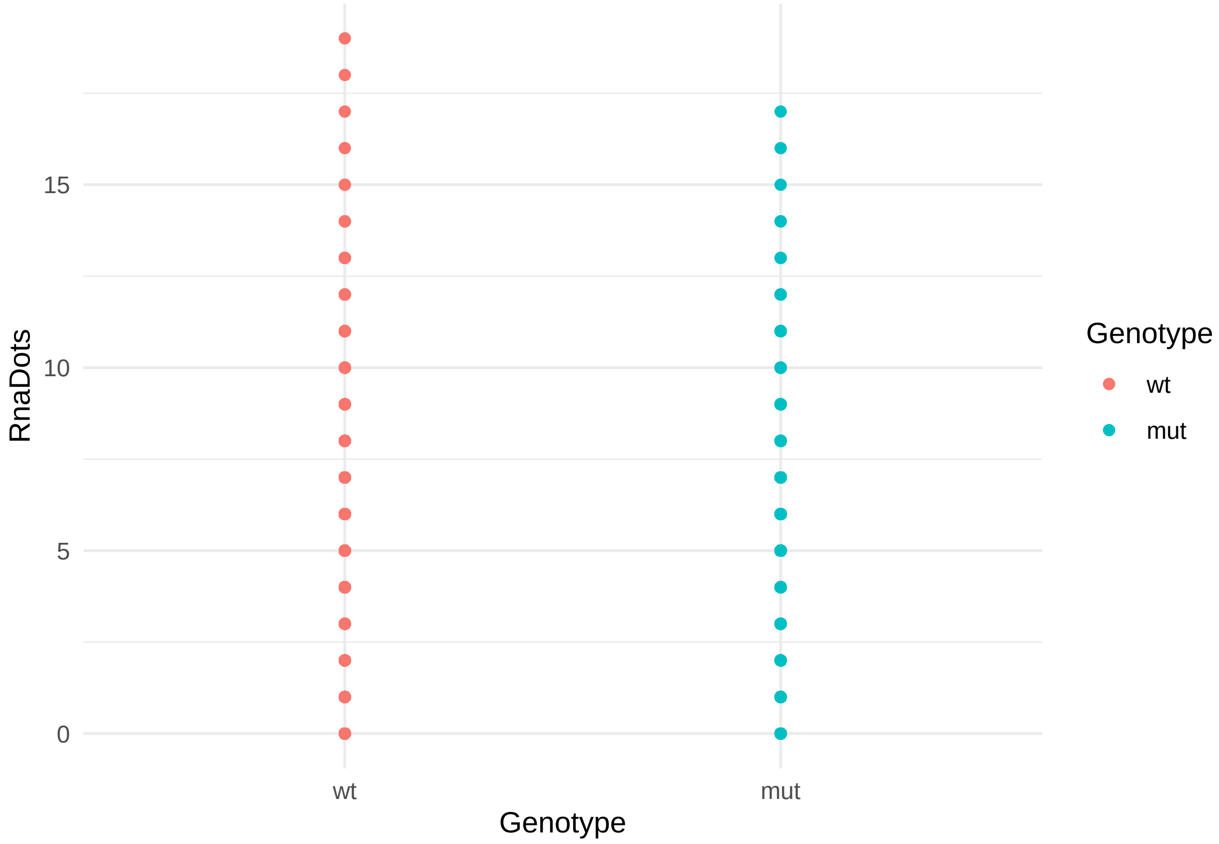


```
qqnorm(resid(model))
qqline(resid(model))
```



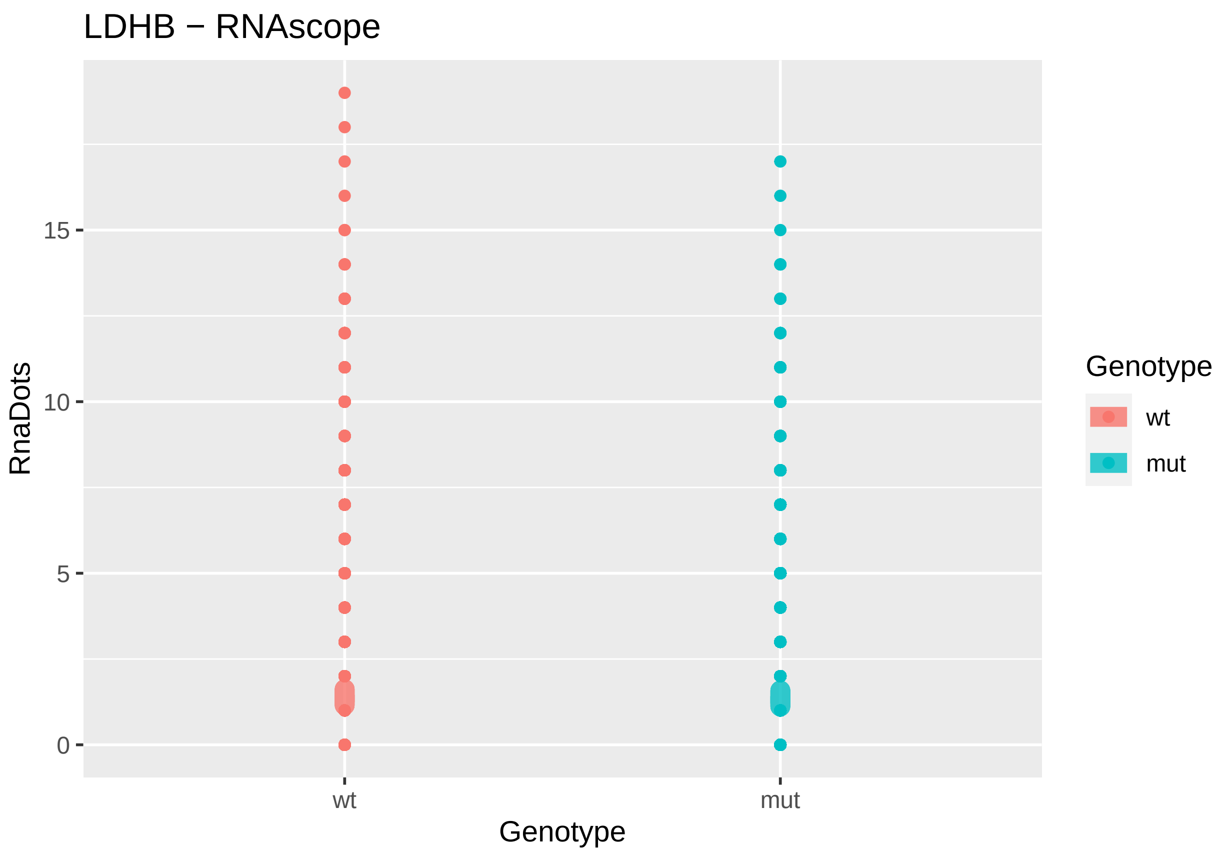
```
data <- ldhb_Rna
ggplot(data, aes(x = Genotype, y = RnaDots, color = Genotype)) +
  geom_point() +
  labs(x = "Genotype", y = "RnaDots", color = "Genotype") +
  theme_minimal()
```

```
## Warning: Removed 1228 rows containing missing values (‘geom_point()‘).
```

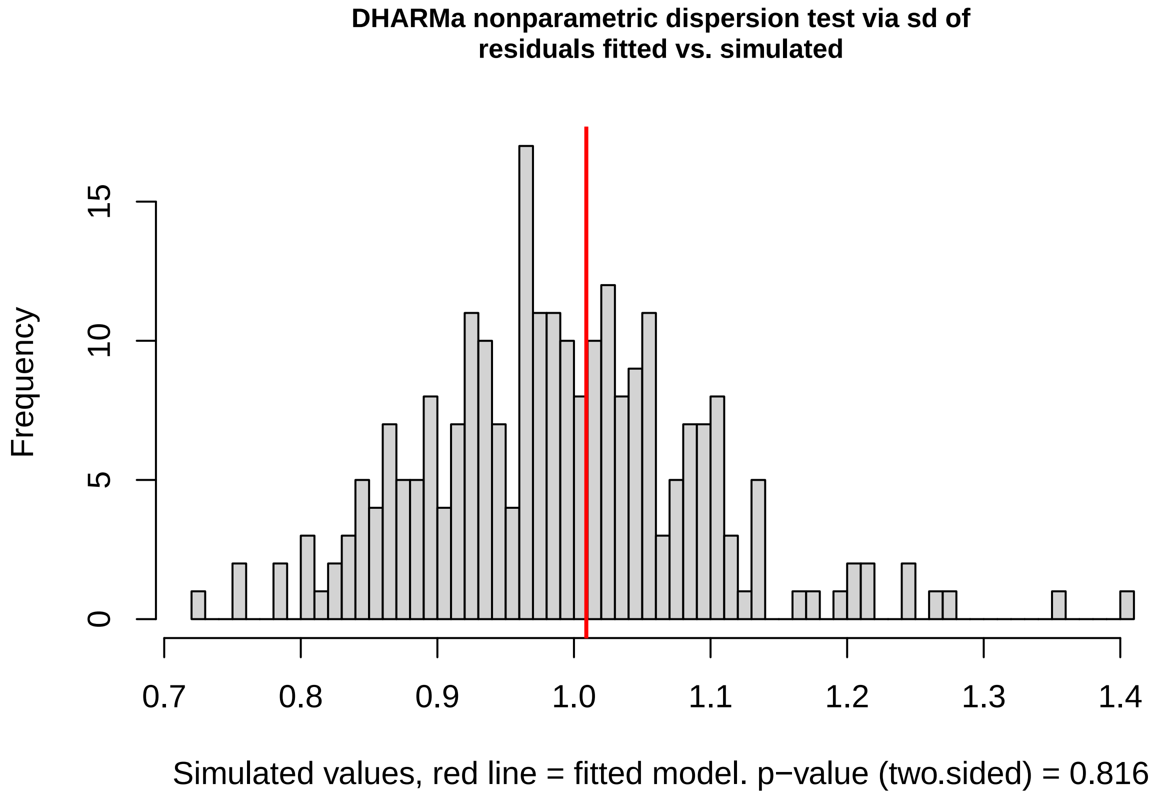


```
# Create a new dataset for predictions
prediction <- expand.grid(Genotype = unique(data$Genotype),
                          Animal = unique(data$Animal))

# Add predictions from the models
prediction$Pred_Model <- predict(model, newdata = prediction)

# Plot the data and the fitted models
ggplot(data, aes(x = Genotype, y = RnaDots, color = Genotype)) +
  geom_point() +
  geom_line(data = prediction, aes(x = Genotype, y = Pred_Model, color = Genotype), linetype = "solid",
  labs(x = "Genotype", y = "RnaDots", color = "Genotype", title = "LDHB - RNAscope")
```

```
## Warning: Removed 1228 rows containing missing values (‘geom_point()‘).
```

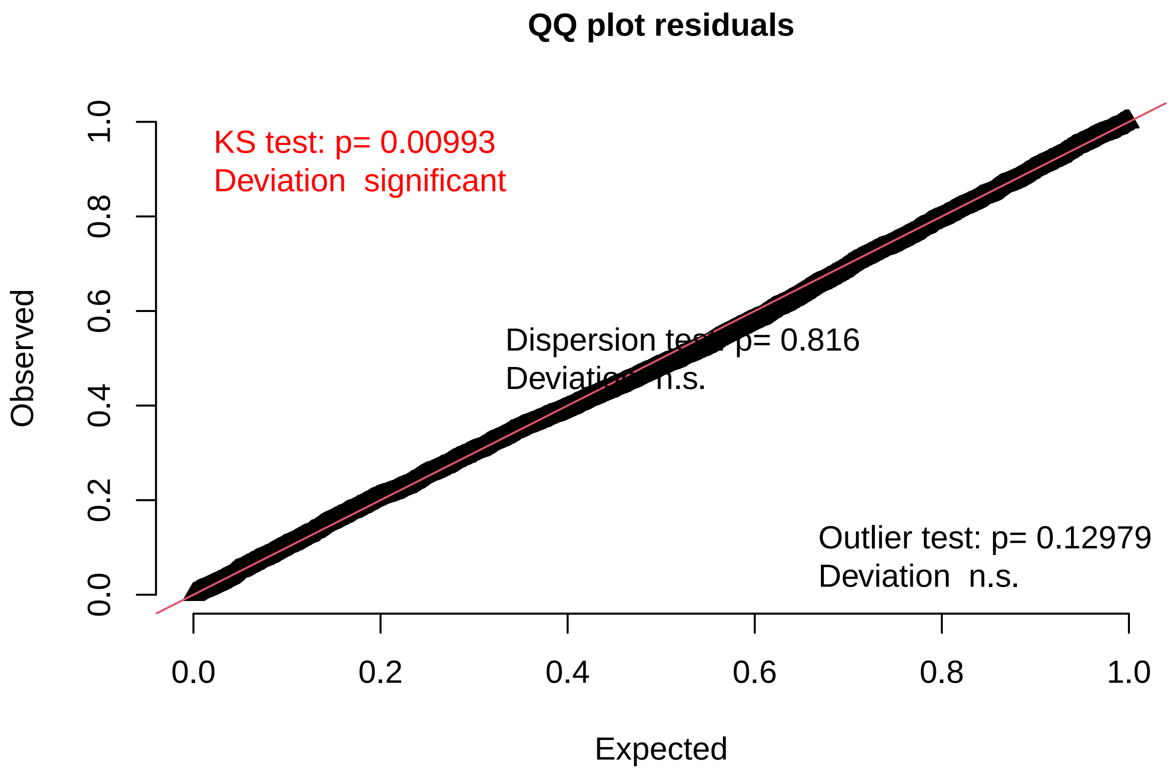


```
# Create simulated residuals
simulationOutput <- simulateResiduals(fittedModel = model)

# Check for dispersion
testDispersion(simulationOutput)
```



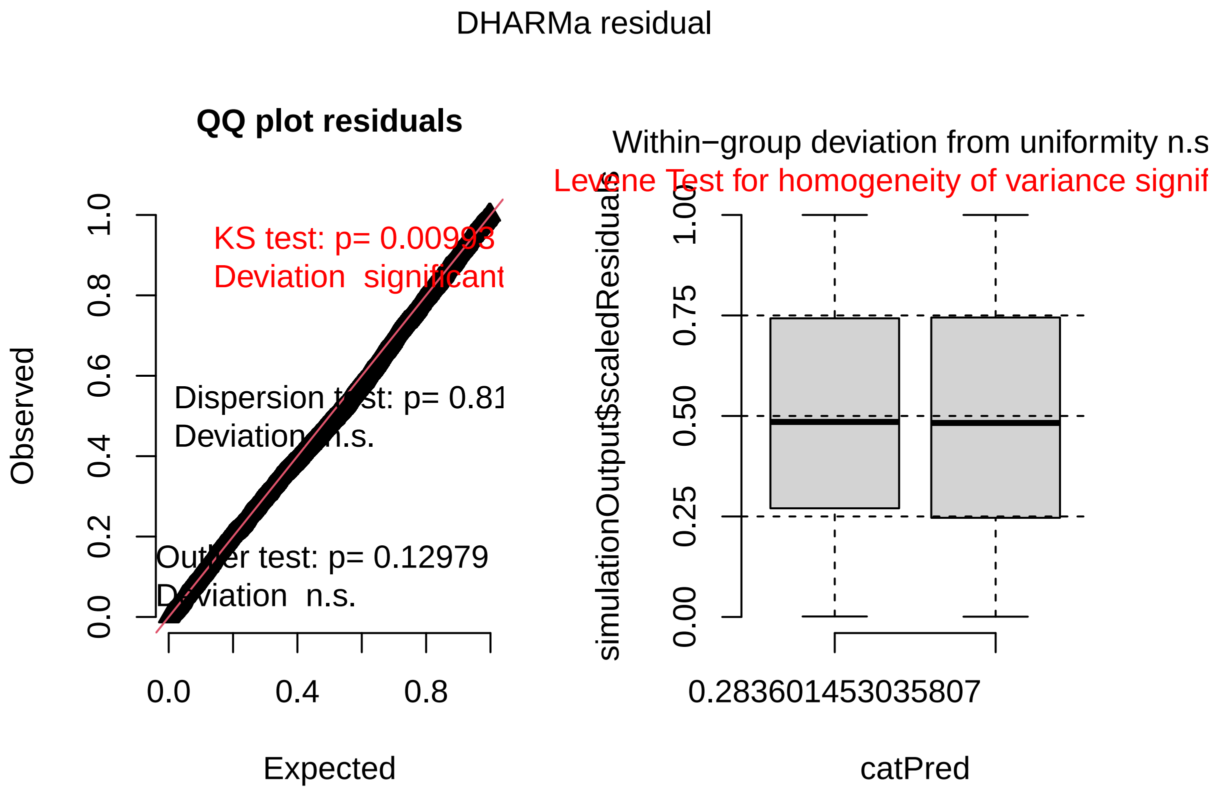
```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 1.0191, p-value = 0.816
## alternative hypothesis: two.sided
```

```
# Check for uniformity of residuals
testUniformity(simulationOutput)
```



```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  simulationOutput$scaledResiduals
## D = 0.021918, p-value = 0.009929
## alternative hypothesis: two-sided
```

```
# Create diagnostic plots
plot(simulationOutput)
```



```
#Imports excel file
LDHAB_ON_CA2_file <- "LDHAB_EM_ON_CAII_Quantification.xlsx"
ldha_OL_CA2 <- read_excel(path = here::here(LDHAB_ON_CA2_file), sheet = "LDHA") %>%
  mutate(Staining = "LDHA")

#ldh_OL_CA2 <- dplyr::bind_rows(ldha_OL_CA2,ldhb_OL_CA2)

model <- lmer(goldDotsPerArea ~ Genotype + (1|Mouse) + (1|Mouse:Genotype), data = ldha_OL_CA2)
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: goldDotsPerArea ~ Genotype + (1 | Mouse) + (1 | Mouse:Genotype)
##    Data: ldha_OL_CA2
##
## REML criterion at convergence: 67.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9909 -0.4807  0.1087  0.5286  2.5549
##
## Random effects:
##  Groups         Name        Variance Std.Dev.
##  Mouse          (Intercept) 0.01867  0.1366
##  Mouse:Genotype (Intercept) 0.02137  0.1462
##  Residual                   0.33664  0.5802
## Number of obs: 36, groups:  Mouse, 8; Mouse:Genotype, 8
##
## Fixed effects:
##              Estimate Std. Error     df t value Pr(>|t|)    
## (Intercept)   1.1647     0.1667 5.2524   6.987 0.000755 ***
## Genotypemut   0.1140     0.2409 5.5974   0.473 0.653948    
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Correlation of Fixed Effects:
##             (Intr)
## Genotypemut -0.692
```

```
anova(model)
```

```
## Type III Analysis of Variance Table with Satterthwaite’s method
##            Sum Sq  Mean Sq NumDF  DenDF F value Pr(>F)
## Genotype 0.075384 0.075384     1 5.5974  0.2239 0.6539
```

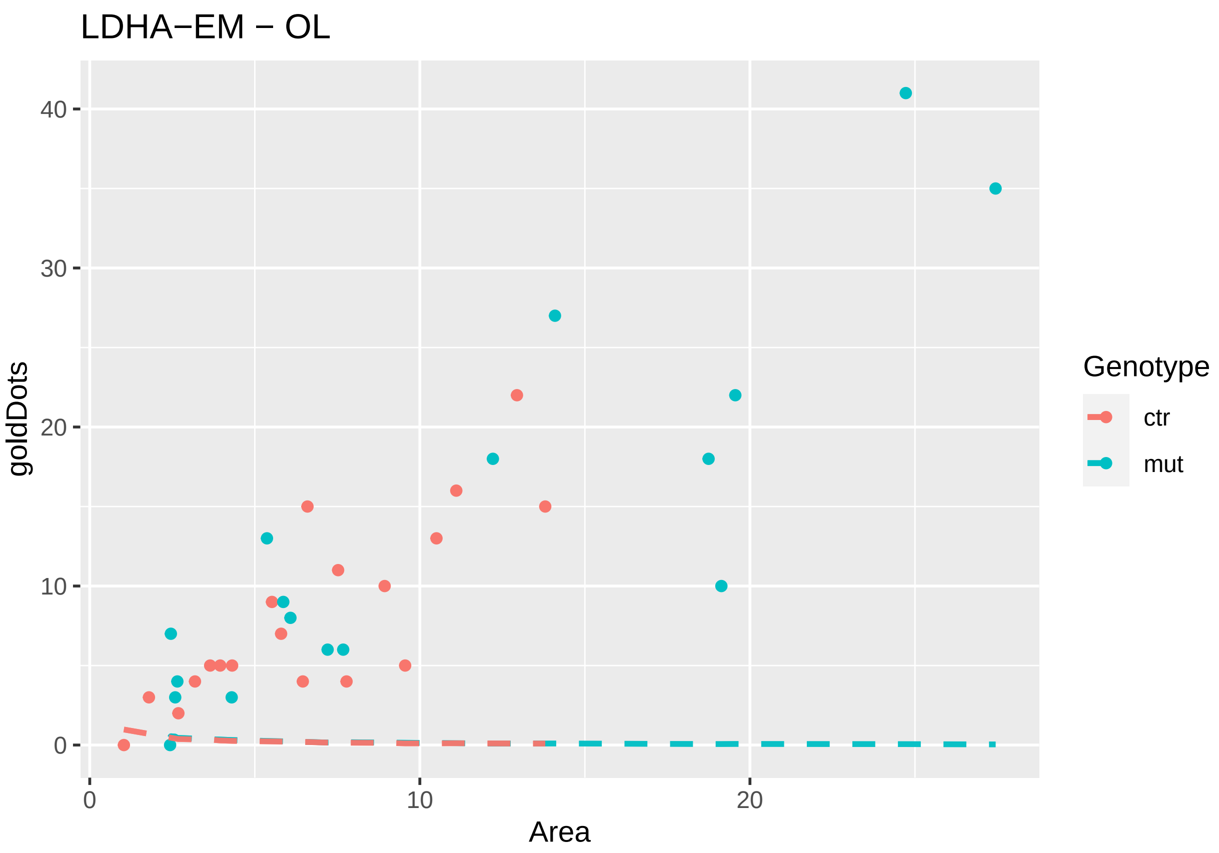
```
# Pairwise compaison
emm <- emmeans(model, ~ Genotype)
pairs(emm)
```

```
##  contrast  estimate    SE   df t.ratio p.value
##  ctr - mut   -0.114 0.242 5.85  -0.470  0.6552
##
## Degrees-of-freedom method: kenward-roger
```

```
data <- ldha_OL_CA2
# Create a new dataset for predictions
prediction <- data %>%
  dplyr::select(Genotype, Mouse, Area) %>%
  distinct()

# Add predictions from the models
prediction$Pred_Model <- predict(model, newdata = prediction) /prediction$Area

lowerLimit_mut <- prediction %>%
  dplyr::filter(Genotype == "mut") %>%
  dplyr::group_by(Area) %>%
  dplyr::filter(Pred_Model == min(Pred_Model))
upperLimit_mut <- prediction %>%
  dplyr::filter(Genotype == "mut") %>%
  dplyr::group_by(Area) %>%
  dplyr::filter(Pred_Model == max(Pred_Model))
lowerLimit_ctr <- prediction %>%
  dplyr::filter(Genotype != "mut") %>%
  dplyr::group_by(Area) %>%
  dplyr::filter(Pred_Model == min(Pred_Model))
upperLimit_ctr <- prediction %>%
  dplyr::filter(Genotype != "mut") %>%
  dplyr::group_by(Area) %>%
  dplyr::filter(Pred_Model == max(Pred_Model))

prediction$Pred_Model[seq(from = 1, to = length(prediction$Pred_Model), by = 2)]
```

```
##          1          3          5          7          9         11         13
## 0.25862646 0.05615674 0.09850340 0.10562320 0.97604419 0.19418655 0.14030380
##         15         17         19         21         23         25         27
## 0.19003550 0.06294636 0.11351896 0.09739720 0.09134825 0.35614714 0.10809188
##         29         31         33         35
## 0.55622940 0.23309835 0.16029896 0.47457450
```

```
# Plot the data and the fitted models
ggplot(data, aes(x = Area, y = goldDots, color = Genotype)) +
  geom_point() +
  geom_line(data = lowerLimit_mut, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed"
  geom_line(data = upperLimit_mut, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed"
  geom_line(data = upperLimit_ctr, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed"
  geom_line(data = lowerLimit_ctr, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed"
  labs(x = "Area", y = "goldDots", color = "Genotype", title = "LDHA-EM - OL")
```



```
#Imports excel file
ldhb_OL_CA2 <- read_excel(path = here::here(LDHAB_ON_CA2_file), sheet = "LDHB") %>%
  mutate(goldDotsPerArea = goldDots / Area) %>%
  na.omit
```

```
dependentVar <- "goldDotsPerArea"
fixedEffect <- "Genotype"
groupingVars <- (c("Mouse", "Genotype"))
analysisName <- "LDHB_OLs+CA2"
formula <- goldDotsPerArea ~ Genotype + (1|Mouse) + (1|Mouse:Genotype)
Interaction <- NULL

model <- lmer(goldDotsPerArea ~ Genotype + (1|Mouse) + (1|Mouse:Genotype), data = ldhb_OL_CA2)
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: goldDotsPerArea ~ Genotype + (1 | Mouse) + (1 | Mouse:Genotype)
##    Data: ldhb_OL_CA2
## 
## REML criterion at convergence: 45.2
## 
## Scaled residuals: 
##      Min       1Q   Median       3Q      Max 
## -1.27929 -0.66144 -0.00569  0.65069  2.31729 
## 
## Random effects:
##  Groups         Name        Variance Std.Dev.
##  Mouse          (Intercept) 0.00460  0.06783 
##  Mouse:Genotype (Intercept) 0.05205  0.22814 
##  Residual                   0.17883  0.42288 
## Number of obs: 33, groups:  Mouse, 8; Mouse:Genotype, 8
## 
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)  
## (Intercept)  0.47059    0.15790 6.04026   2.980   0.0244 *
## Genotypemut  0.07138    0.22466 6.19787   0.318   0.7611  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Correlation of Fixed Effects:
##             (Intr)
## Genotypemut -0.703
```

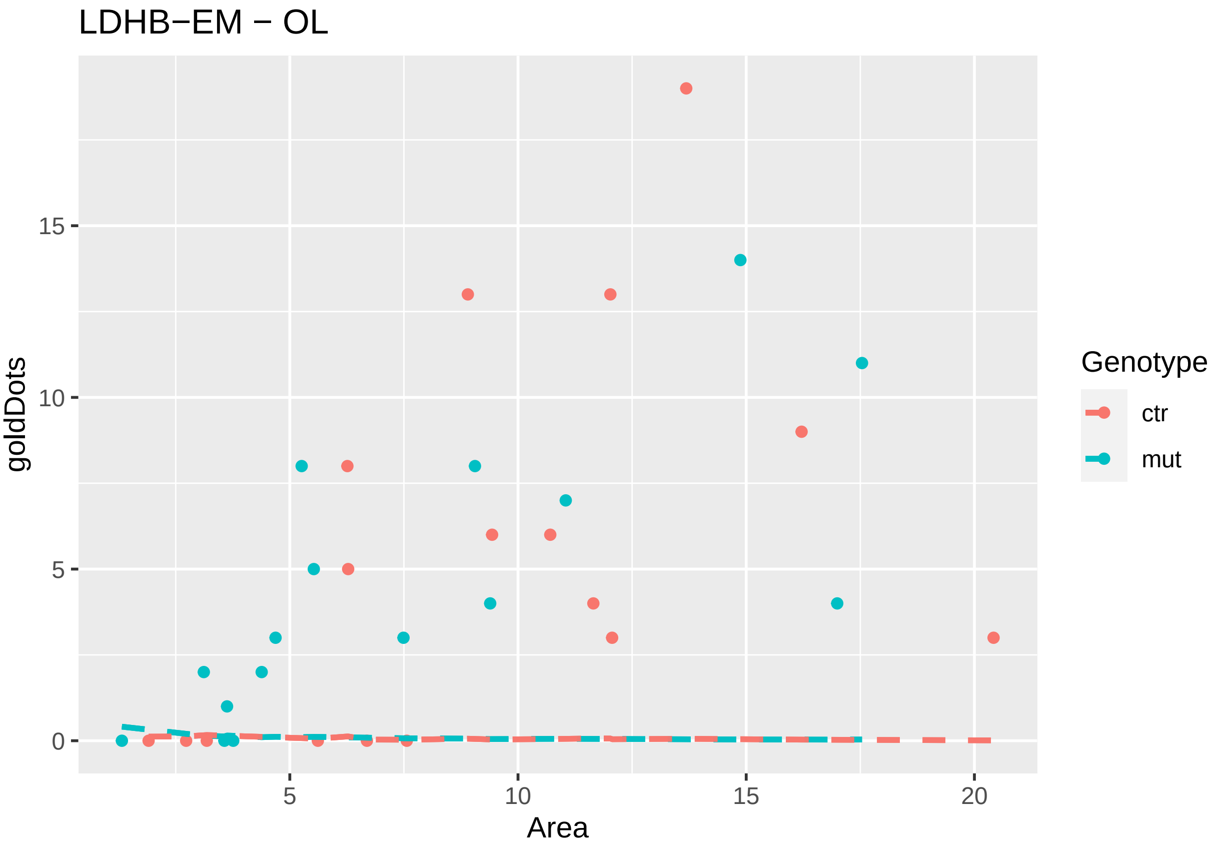
```
anova(model)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##            Sum Sq  Mean Sq NumDF  DenDF F value Pr(>F)
## Genotype 0.018051 0.018051     1 6.1979  0.1009 0.7611
```

```
emm <- emmeans(model, ~ Genotype)
pairs(emm)
```

```
##  contrast  estimate    SE   df t.ratio p.value
##  ctr - mut  -0.0714 0.225 5.92  -0.317  0.7623
## 
## Degrees-of-freedom method: kenward-roger
```

```r
data <- ldhb_OL_CA2
# Create a new dataset for predictions
prediction <- data %>%
  dplyr::select(Genotype, Mouse, Area) %>%
  distinct()

# Add predictions from the models
prediction$Pred_Model <- predict(model, newdata = prediction) /prediction$Area

lowerLimit_mut <- prediction %>%
  dplyr::filter(Genotype == "mut") %>%
  dplyr::group_by(Area) %>%
  dplyr::filter(Pred_Model == min(Pred_Model))
upperLimit_mut <- prediction %>%
  dplyr::filter(Genotype == "mut") %>%
  dplyr::group_by(Area) %>%
  dplyr::filter(Pred_Model == max(Pred_Model))
lowerLimit_ctr <- prediction %>%
  dplyr::filter(Genotype != "mut") %>%
  dplyr::group_by(Area) %>%
  dplyr::filter(Pred_Model == min(Pred_Model))
upperLimit_ctr <- prediction %>%
  dplyr::filter(Genotype != "mut") %>%
  dplyr::group_by(Area) %>%
  dplyr::filter(Pred_Model == max(Pred_Model))

# Plot the data and the fitted models
ggplot(data, aes(x = Area, y = goldDots, color = Genotype)) +
  geom_point() +
  geom_line(data = prediction, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed", si
  geom_line(data = lowerLimit_mut, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed"
  geom_line(data = upperLimit_mut, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed"
  geom_line(data = upperLimit_ctr, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed"
  geom_line(data = lowerLimit_ctr, aes(x = Area, y = Pred_Model, color = Genotype), linetype = "dashed"
  labs(x = "Area", y = "goldDots", color = "Genotype", title = "LDHB-EM - OL")
```


LDHB−EM − OL
goldDots
0
5
10
15
Area
5
10
15
20
Genotype
ctr
mut