

- 1 pacchetti
- 2 Preparazione dati
- 3 Model MTT

MTT assay

RRF

2026-06-05

1 pacchetti

```
library(tidyverse)
```

```
## Warning: il pacchetto 'tidyverse' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'ggplot2' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tibble' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tidyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'readr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'purrr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'dplyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'stringr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'forcats' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'lubridate' è stato creato con R versione 4.5.1
```

```
## — Attaching core tidyverse packages ————— tidyverse
2.0.0 —
## ✓ dplyr      1.1.4      ✓ readr      2.1.5
## ✓ forcats   1.0.0      ✓ stringr    1.5.1
## ✓ ggplot2    3.5.2      ✓ tibble     3.3.0
## ✓ lubridate 1.9.4      ✓ tidyr      1.3.1
## ✓ purrr      1.0.4
## — Conflicts ————— tidyverse_confli
cts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force al
l conflicts to become errors
```

```
library(ggplot2)
library(lme4)
```

```
## Warning: il pacchetto 'lme4' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: Matrix
##
## Caricamento pacchetto: 'Matrix'
##
## I seguenti oggetti sono mascherati da 'package:tidyr':
##
##      expand, pack, unpack
```

```
library(lmerTest)
```

```
## Warning: il pacchetto 'lmerTest' è stato creato con R versione 4.5.1
```

```
##
## Caricamento pacchetto: 'lmerTest'
##
## Il seguente oggetto è mascherato da 'package:lme4':
##
##      lmer
##
## Il seguente oggetto è mascherato da 'package:stats':
##
##      step
```

```
library(sjPlot)
```

```
## Warning: il pacchetto 'sjPlot' è stato creato con R versione 4.5.1
```

```
library(car)
```

```
## Warning: il pacchetto 'car' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: carData
```

```
## Warning: il pacchetto 'carData' è stato creato con R versione 4.5.1
```

```
##  
## Caricamento pacchetto: 'car'  
##  
## Il seguente oggetto è mascherato da 'package:dplyr':  
##  
##      recode  
##  
## Il seguente oggetto è mascherato da 'package:purrr':  
##  
##      some
```

2 Preparazione dati

```
db <- read.csv2("C:/Users/utente2/Desktop/Report PNRR/dataset MTT csv.csv")  
  
db$group <- factor(db$GROUP, levels=c('CTRL','AD'), ordered=FALSE)  
  
db$time2 <- factor(db$TIME, exclude = c("", NA))
```

3 Model MTT

```
Model_1 <- lmer(log(MTT) ~ time2 + group + group*time2 + (1 | ID),  
               data = db)  
summary(Model_1)
```

```

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(MTT) ~ time2 + group + group * time2 + (1 | ID)
## Data: db
##
## REML criterion at convergence: 69.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.59019 -0.27100 -0.06778  0.29345  1.50770
##
## Random effects:
## Groups Name Variance Std.Dev.
## ID      (Intercept) 0.5647  0.7515
## Residual              0.8260  0.9088
## Number of obs: 30, groups: ID, 6
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   -0.85490    0.68085 12.05156  -1.256  0.23305
## time21         -0.16375    0.74206 16.00000  -0.221  0.82814
## time22         -0.29572    0.74206 16.00000  -0.399  0.69552
## time23        -1.62651    0.74206 16.00000  -2.192  0.04352 *
## time24        -2.30935    0.74206 16.00000  -3.112  0.00671 **
## groupAD         0.16400    0.96286 12.05156   0.170  0.86758
## time21:groupAD  0.04855    1.04943 16.00000   0.046  0.96367
## time22:groupAD  0.14425    1.04943 16.00000   0.137  0.89239
## time23:groupAD -0.23925    1.04943 16.00000  -0.228  0.82255
## time24:groupAD  0.02925    1.04943 16.00000   0.028  0.97811
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) time21 time22 time23 time24 gropAD t21:AD t22:AD t23:
AD
## time21         -0.545
## time22         -0.545  0.500
## time23         -0.545  0.500  0.500
## time24         -0.545  0.500  0.500  0.500
## groupAD        -0.707  0.385  0.385  0.385  0.385
## tim21:grpAD     0.385 -0.707 -0.354 -0.354 -0.354 -0.545
## tim22:grpAD     0.385 -0.354 -0.707 -0.354 -0.354 -0.545  0.500
## tim23:grpAD     0.385 -0.354 -0.354 -0.707 -0.354 -0.545  0.500  0.500
## tim24:grpAD     0.385 -0.354 -0.354 -0.354 -0.707 -0.545  0.500  0.500  0.5
00

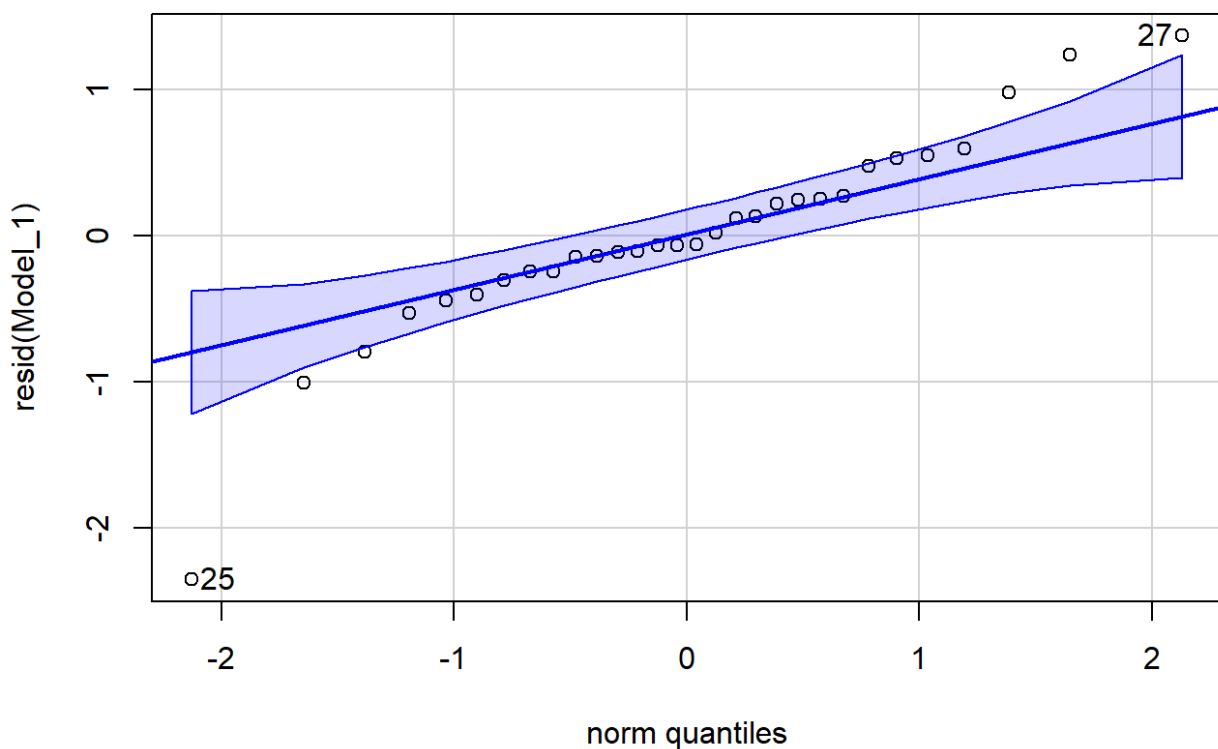
```



```
confint(Model_1, level = 0.95, method = "Wald")
```

```
##              2.5 %    97.5 %  
## .sig01         NA         NA  
## .sigma         NA         NA  
## (Intercept)  -2.189337  0.4795370  
## time21       -1.618157  1.2906484  
## time22       -1.750126  1.1586795  
## time23       -3.080914 -0.1721087  
## time24       -3.763750 -0.8549450  
## groupAD      -1.723182  2.0511757  
## time21:groupAD -2.008281  2.1053902  
## time22:groupAD -1.912588  2.2010837  
## time23:groupAD -2.296082  1.8175892  
## time24:groupAD -2.027582  2.0860893
```

```
qqPlot(resid(Model_1))
```



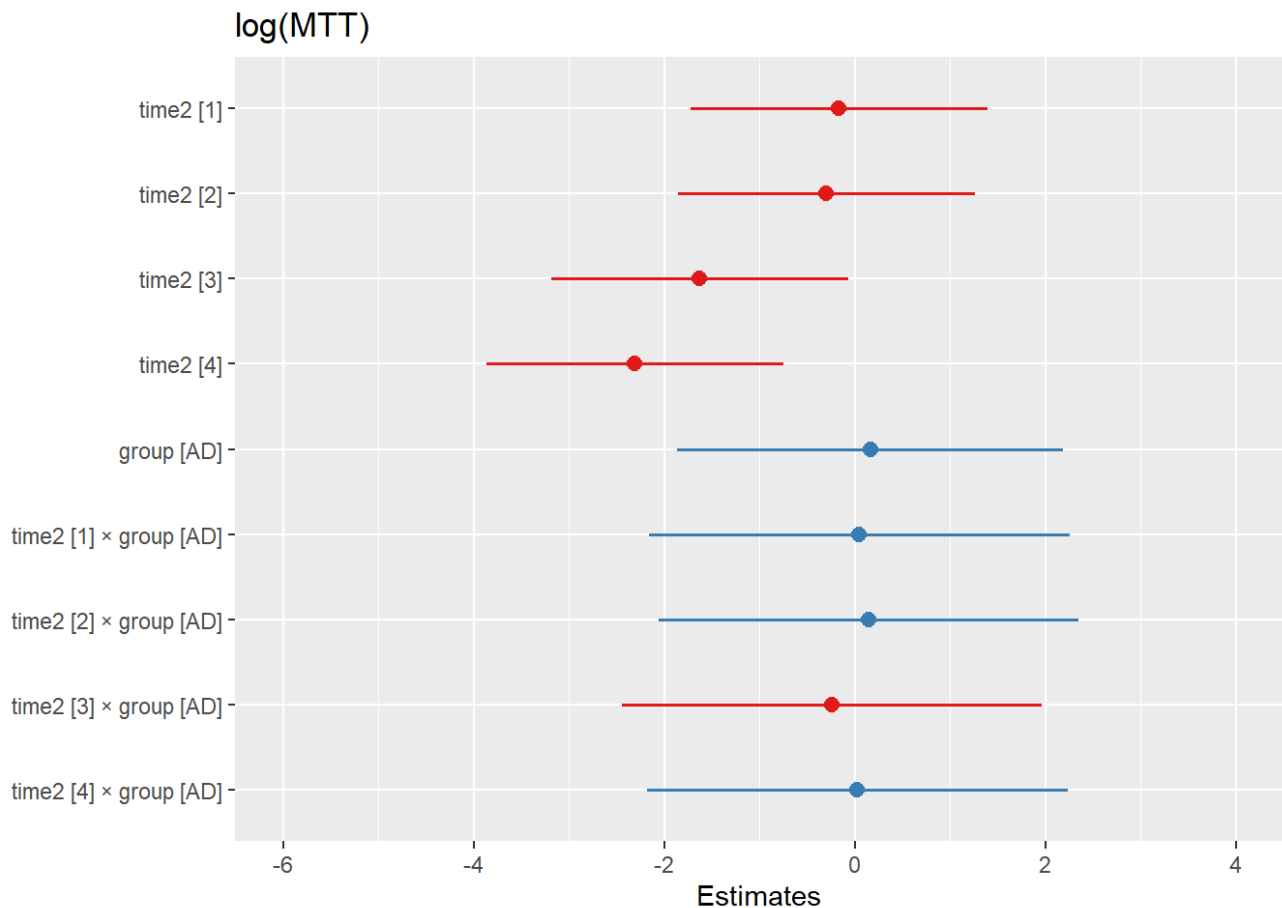
```
## [1] 25 27
```

```
shapiro.test(resid(Model_1))
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: resid(Model_1)  
## W = 0.91275, p-value = 0.01746
```

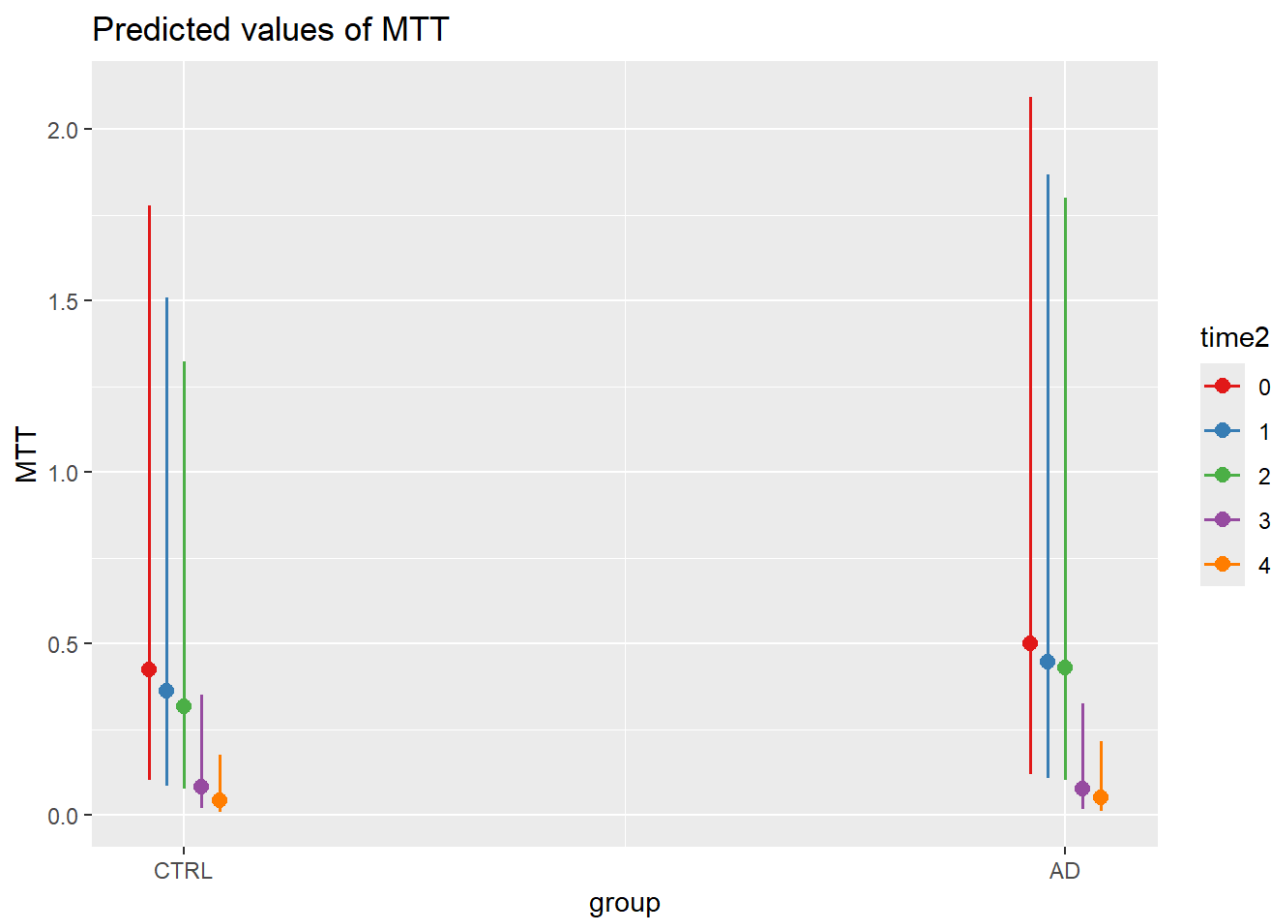
3.1 Visualizzazione del modello

```
plot_model(Model_1)
```



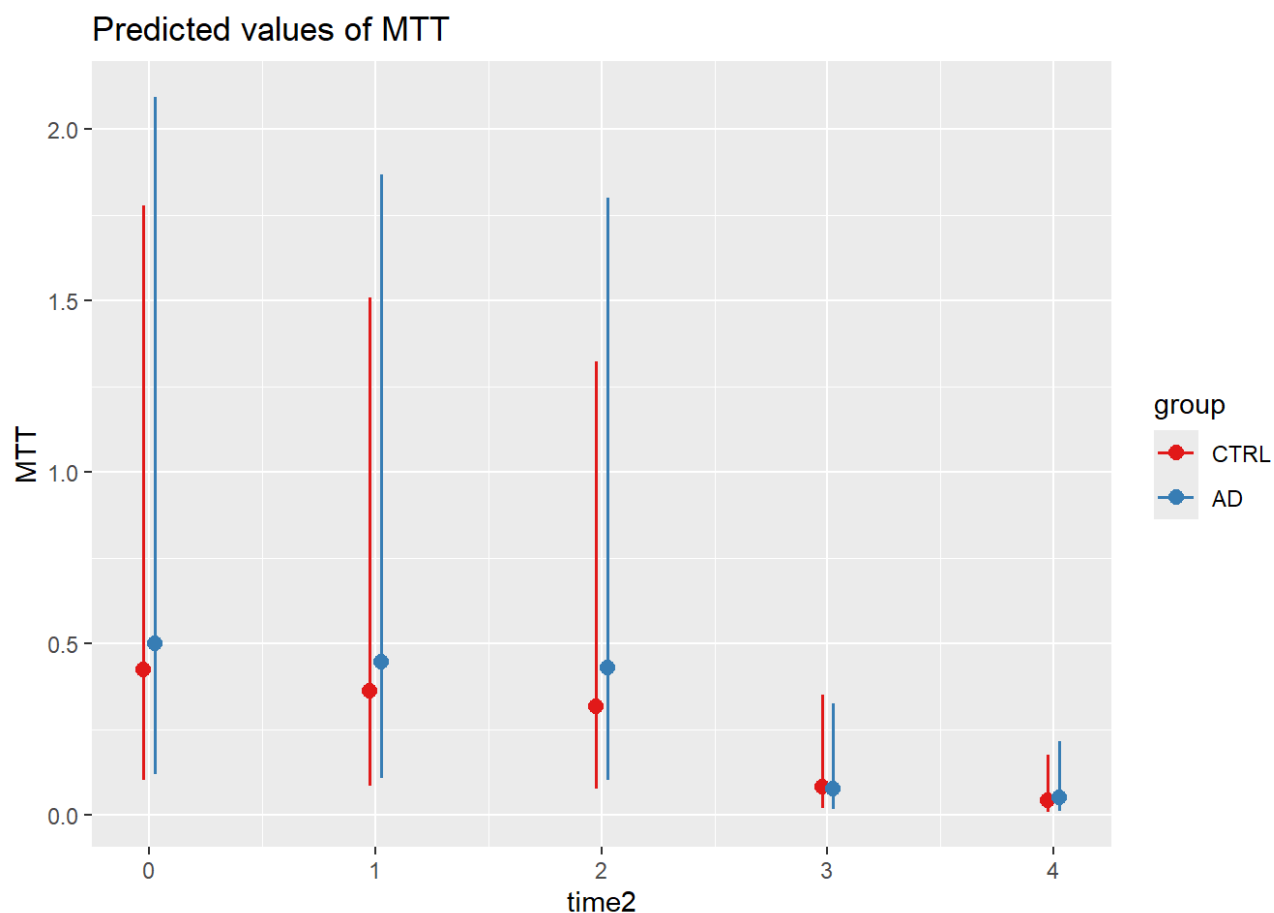
```
plot_model(Model_1,type="pred",  
            terms=c("group", "time2"),  
            pred.type="fe")
```

```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```



```
plot_model(Model_1,type="pred",  
            terms=c("time2", "group"),  
            pred.type="fe")
```

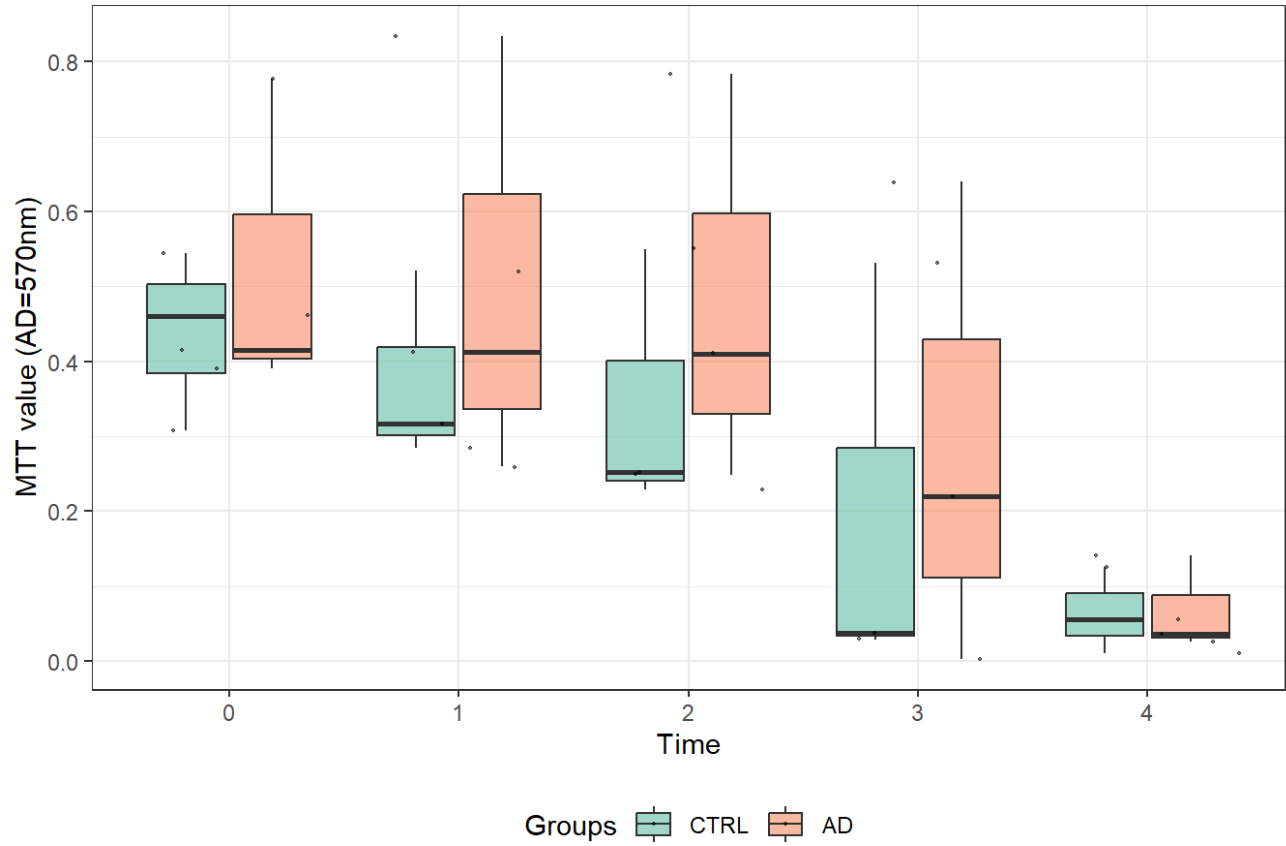
```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```



Personalizzazione

```
db %>%
  drop_na(time2, group) %>%
  ggplot(aes(y = MTT, x = time2, fill = group)) +
  geom_boxplot(alpha = 0.6, varwidth = TRUE, na.rm = TRUE) +
  geom_jitter(color = "black", alpha = 0.5, size = 0.3, na.rm = TRUE) +
  scale_fill_brewer(palette = "Set2") +
  scale_color_brewer(palette = "Set2") +
  theme_bw(base_family = "sans") +
  theme(legend.position = "bottom") +
  ggtitle("Cell viability (MTT assay)") +
  xlab("Time") +
  ylab("MTT value (AD=570nm)") +
  labs(color = "GROUP", fill = "Groups")
```

Cell viability (MTT assay)



- 1 pacchetti
- 2 Preparazione dati
- 3 Model Mitotracker

Mitotracker

RRF

2026-06-05

1 pacchetti

```
library(tidyverse)
```

```
## Warning: il pacchetto 'tidyverse' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'ggplot2' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tibble' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tidyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'readr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'purrr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'dplyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'stringr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'forcats' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'lubridate' è stato creato con R versione 4.5.1
```



```
## — Attaching core tidyverse packages — tidyverse
2.0.0 —
## ✓ dplyr      1.1.4      ✓ readr      2.1.5
## ✓ forcats   1.0.0      ✓ stringr    1.5.1
## ✓ ggplot2    3.5.2      ✓ tibble     3.3.0
## ✓ lubridate  1.9.4      ✓ tidyr      1.3.1
## ✓ purrr      1.0.4
## — Conflicts — tidyverse_confli
cts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force al
l conflicts to become errors
```

```
library(ggplot2)
library(lme4)
```

```
## Warning: il pacchetto 'lme4' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: Matrix
##
## Caricamento pacchetto: 'Matrix'
##
## I seguenti oggetti sono mascherati da 'package:tidyr':
##
##     expand, pack, unpack
```

```
library(lmerTest)
```

```
## Warning: il pacchetto 'lmerTest' è stato creato con R versione 4.5.1
```

```
##
## Caricamento pacchetto: 'lmerTest'
##
## Il seguente oggetto è mascherato da 'package:lme4':
##
##     lmer
##
## Il seguente oggetto è mascherato da 'package:stats':
##
##     step
```

```
library(sjPlot)
```

```
## Warning: il pacchetto 'sjPlot' è stato creato con R versione 4.5.1
```

```
library(car)
```

```
## Warning: il pacchetto 'car' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: carData
```

```
## Warning: il pacchetto 'carData' è stato creato con R versione 4.5.1
```

```
##  
## Caricamento pacchetto: 'car'  
##  
## Il seguente oggetto è mascherato da 'package:dplyr':  
##  
##      recode  
##  
## Il seguente oggetto è mascherato da 'package:purrr':  
##  
##      some
```

2 Preparazione dati

```
db <- read.csv2("C:/Users/utente2/Desktop/Report PNRR/mito.csv")  
  
db$group <- factor(db$group, levels=c('CTRL','AD'), ordered=FALSE)  
  
db$time2 <- factor(db$time, exclude = c("", NA))
```

3 Model Mitotracker

```
Model_1 <- lmer(log(mitotr) ~ time2 + group + group*time2 + (1 | id),  
               data = db)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(Model_1)
```

```

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(mitotr) ~ time2 + group + group * time2 + (1 | id)
## Data: db
##
## REML criterion at convergence: 37.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.23998 -0.40684 -0.06368  0.55566  1.26294
##
## Random effects:
## Groups Name Variance Std.Dev.
## id      (Intercept) 0.0000  0.0000
## Residual          0.2167  0.4655
## Number of obs: 30, groups: id, 6
##
## Fixed effects:
##              Estimate Std. Error    df t value Pr(>|t|)
## (Intercept)   9.802341   0.268768 20.000000   36.471 < 2e-16 ***
## time21        0.156691   0.380095 20.000000    0.412  0.68455
## time22        0.285572   0.380095 20.000000    0.751  0.46121
## time23       -0.260558   0.380095 20.000000   -0.686  0.50089
## time24       -2.152700   0.380095 20.000000  -5.664 1.53e-05 ***
## groupAD       -0.599798   0.380095 20.000000   -1.578  0.13025
## time21:groupAD -0.017817   0.537536 20.000000   -0.033  0.97389
## time22:groupAD  0.475283   0.537536 20.000000    0.884  0.38710
## time23:groupAD -0.007025   0.537536 20.000000   -0.013  0.98970
## time24:groupAD  1.631154   0.537536 20.000000    3.035  0.00655 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) time21 time22 time23 time24 gropAD t21:AD t22:AD t23:
AD
## time21      -0.707
## time22      -0.707  0.500
## time23      -0.707  0.500  0.500
## time24      -0.707  0.500  0.500  0.500
## groupAD     -0.707  0.500  0.500  0.500  0.500
## tim21:grpAD  0.500 -0.707 -0.354 -0.354 -0.354 -0.707
## tim22:grpAD  0.500 -0.354 -0.707 -0.354 -0.354 -0.707  0.500
## tim23:grpAD  0.500 -0.354 -0.354 -0.707 -0.354 -0.707  0.500  0.500
## tim24:grpAD  0.500 -0.354 -0.354 -0.354 -0.707 -0.707  0.500  0.500  0.5
00

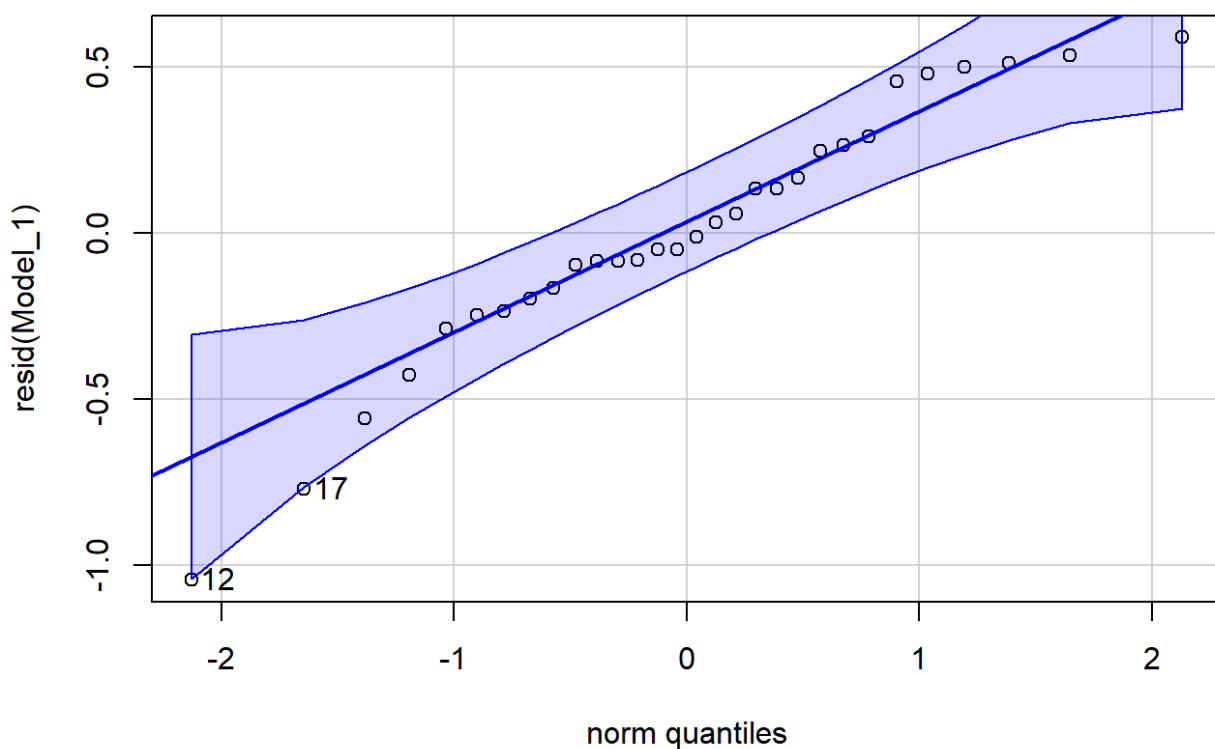
```

```
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
confint(Model_1, level = 0.95, method = "Wald")
```

```
##           2.5 %    97.5 %
## .sig01      NA      NA
## .sigma      NA      NA
## (Intercept)  9.2755656 10.3291165
## time21      -0.5882825  0.9016635
## time22      -0.4594008  1.0305452
## time23      -1.0055310  0.4844150
## time24      -2.8976727 -1.4077267
## groupAD      -1.3447709  0.1451751
## time21:groupAD -1.0713683  1.0357335
## time22:groupAD -0.5782683  1.5288335
## time23:groupAD -1.0605762  1.0465256
## time24:groupAD  0.5776028  2.6847046
```

```
qqPlot(resid(Model_1))
```



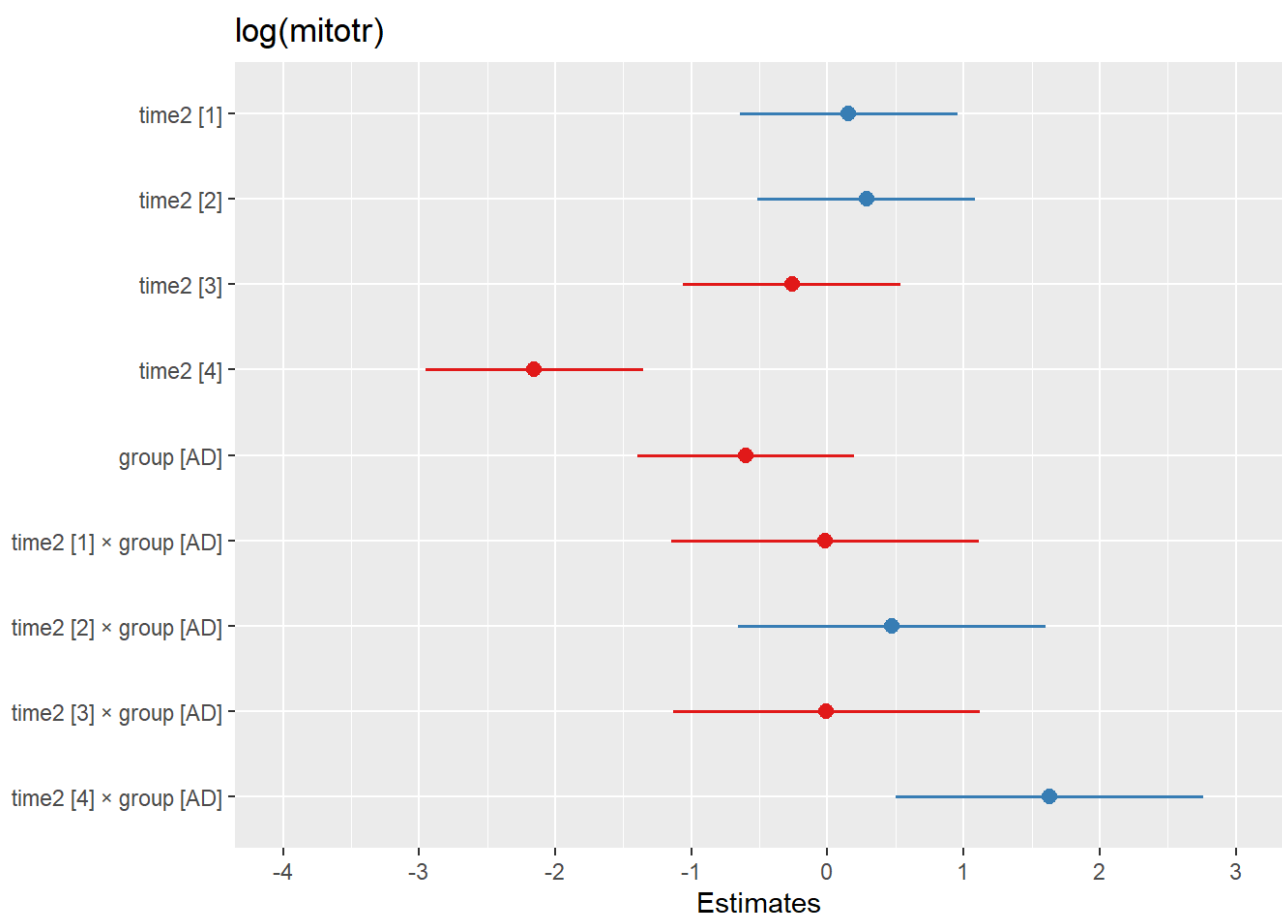
```
## [1] 12 17
```

```
shapiro.test(resid(Model_1))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  resid(Model_1)  
## W = 0.95244, p-value = 0.1964
```

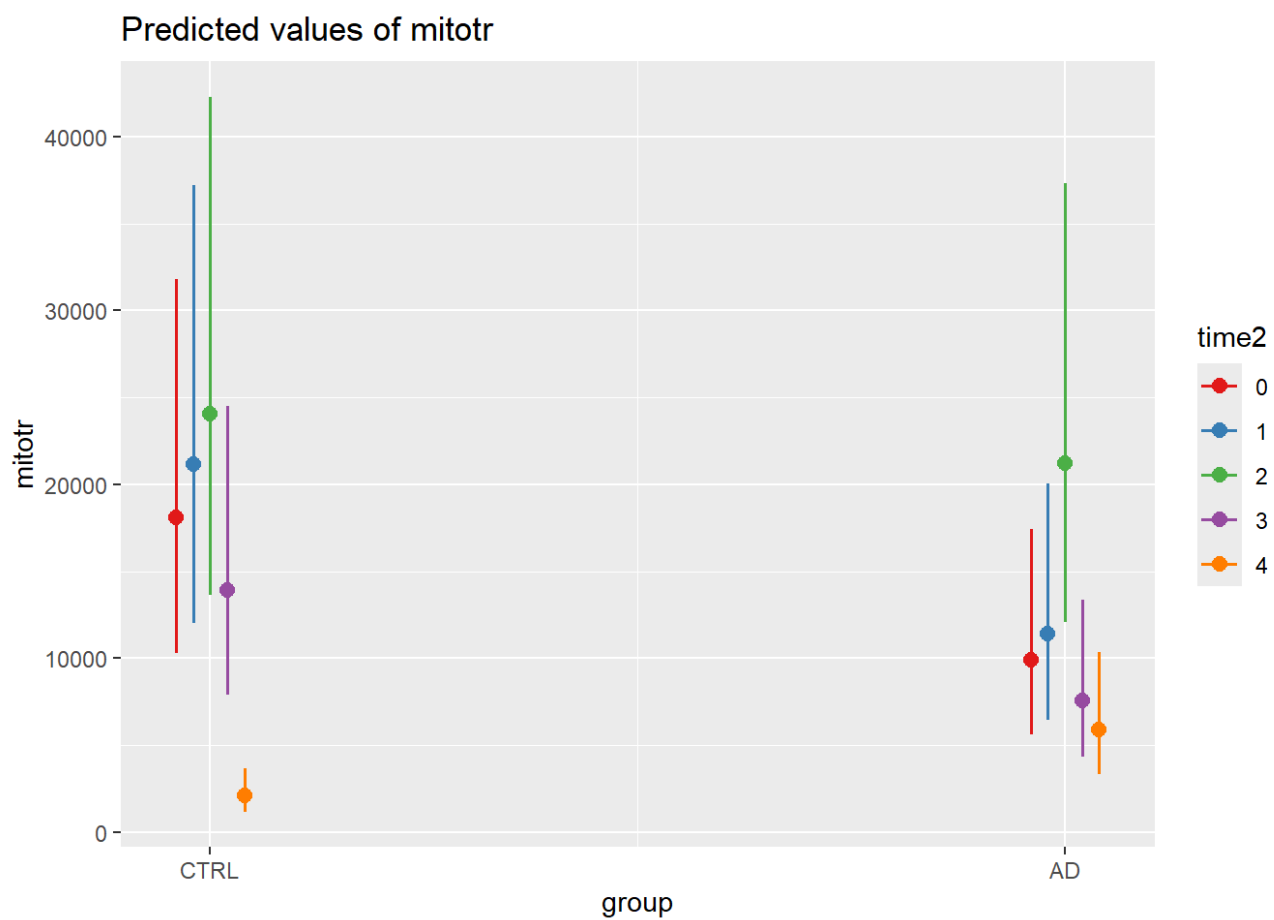
3.1 Visualizzazione del modello

```
plot_model(Model_1)
```



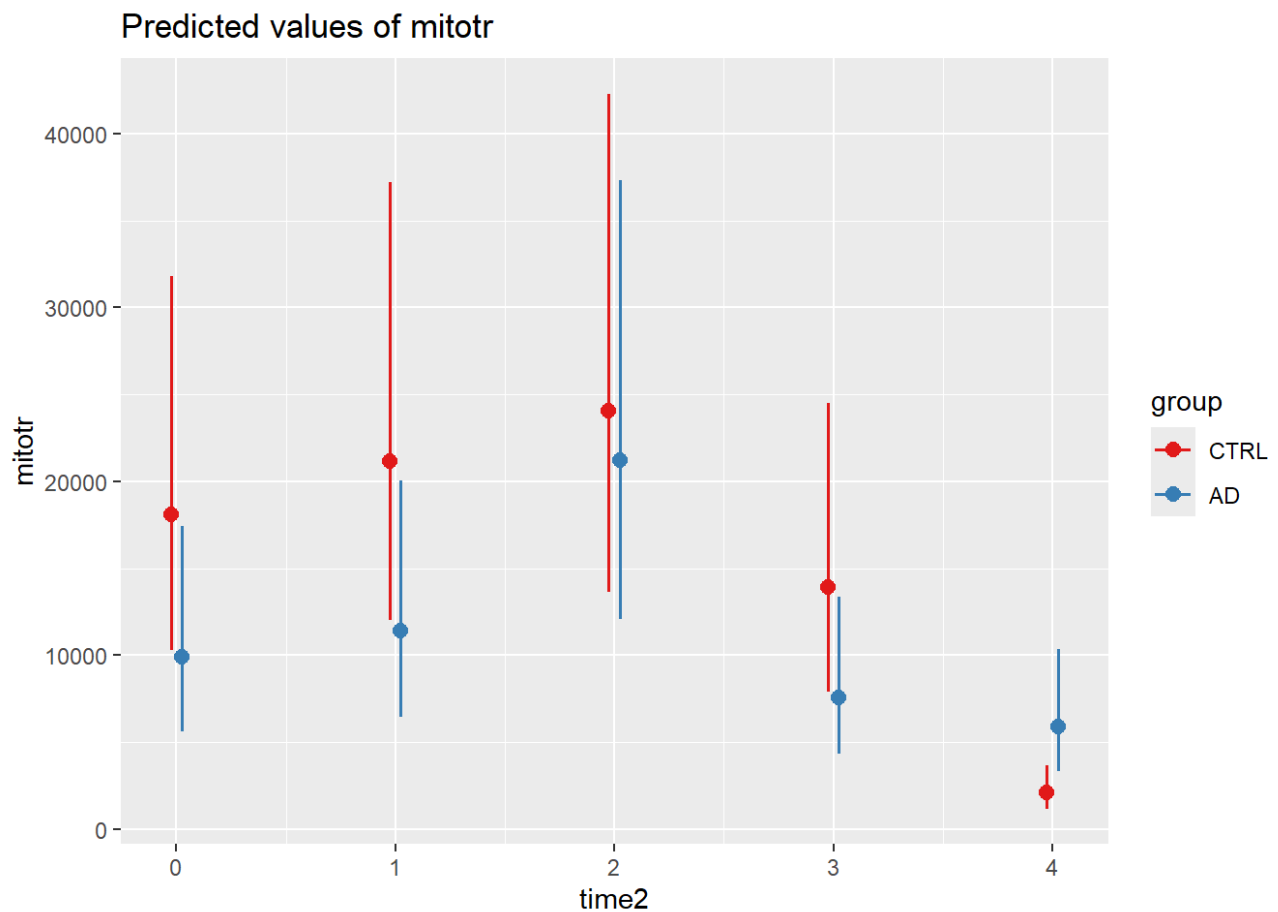
```
plot_model(Model_1, type="pred",  
            terms=c("group", "time2"),  
            pred.type="fe")
```

```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```



```
plot_model(Model_1,type="pred",  
            terms=c("time2", "group"),  
            pred.type="fe")
```

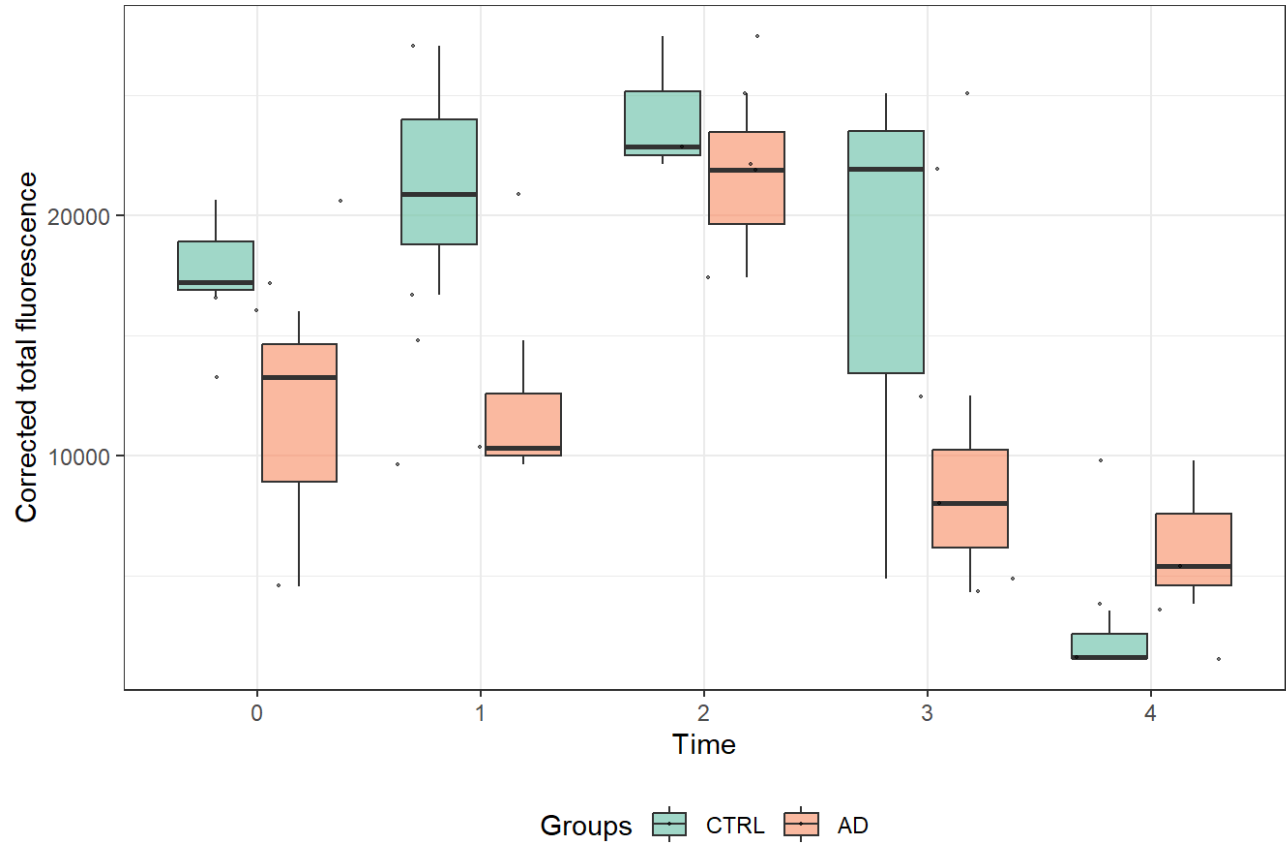
```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```



Personalizzazione

```
db %>%
  drop_na(time2, group) %>%
  ggplot(aes(y = mitotr, x = time2, fill = group)) +
  geom_boxplot(alpha = 0.6, varwidth = TRUE, na.rm = TRUE) +
  geom_jitter(color = "black", alpha = 0.5, size = 0.3, na.rm = TRUE) +
  scale_fill_brewer(palette = "Set2") +
  scale_color_brewer(palette = "Set2") +
  theme_bw(base_family = "sans") +
  theme(legend.position = "bottom") +
  ggtitle("Mitotracker Red CMXRos") +
  xlab("Time") +
  ylab("Corrected total fluorescence") +
  labs(color = "GROUP", fill = "Groups")
```

Mitotracker Red CMXRos



- 1 pacchetti
- 2 Preparazione dati
- 3 Model Oxidative damage

DNA oxidative damage 8-oxo-dG

RRF

2026-06-05

1 pacchetti

```
library(tidyverse)
```

```
## Warning: il pacchetto 'tidyverse' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'ggplot2' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tibble' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tidyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'readr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'purrr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'dplyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'stringr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'forcats' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'lubridate' è stato creato con R versione 4.5.1
```

```
## — Attaching core tidyverse packages ————— tidyverse
2.0.0 —
## ✓ dplyr      1.1.4      ✓ readr      2.1.5
## ✓ forcats   1.0.0      ✓ stringr    1.5.1
## ✓ ggplot2    3.5.2      ✓ tibble     3.3.0
## ✓ lubridate 1.9.4      ✓ tidyr      1.3.1
## ✓ purrr      1.0.4
## — Conflicts ————— tidyverse_confli
cts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force al
l conflicts to become errors
```

```
library(ggplot2)
library(lme4)
```

```
## Warning: il pacchetto 'lme4' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: Matrix
##
## Caricamento pacchetto: 'Matrix'
##
## I seguenti oggetti sono mascherati da 'package:tidyr':
##
##      expand, pack, unpack
```

```
library(lmerTest)
```

```
## Warning: il pacchetto 'lmerTest' è stato creato con R versione 4.5.1
```

```
##
## Caricamento pacchetto: 'lmerTest'
##
## Il seguente oggetto è mascherato da 'package:lme4':
##
##      lmer
##
## Il seguente oggetto è mascherato da 'package:stats':
##
##      step
```

```
library(sjPlot)
```

```
## Warning: il pacchetto 'sjPlot' è stato creato con R versione 4.5.1
```

```
library(car)
```

```
## Warning: il pacchetto 'car' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: carData
```

```
## Warning: il pacchetto 'carData' è stato creato con R versione 4.5.1
```

```
##  
## Caricamento pacchetto: 'car'  
##  
## Il seguente oggetto è mascherato da 'package:dplyr':  
##  
##      recode  
##  
## Il seguente oggetto è mascherato da 'package:purrr':  
##  
##      some
```

2 Preparazione dati

```
db <- read.csv2("C:/Users/utente2/Desktop/Report PNRR/oxodg.csv")  
  
db$group <- factor(db$group, levels=c('CTRL','AD'), ordered=FALSE)  
  
db$time2 <- factor(db$time, exclude = c("", NA))
```

3 Model Oxidative damage

```
Model_1 <- lmer(log(oxo) ~ time2 + group + group*time2 + (1 | id),  
               data = db)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(Model_1)
```

```

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(oxo) ~ time2 + group + group * time2 + (1 | id)
## Data: db
##
## REML criterion at convergence: -11.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.94865 -0.44577 -0.06216  0.52854  1.90944
##
## Random effects:
## Groups Name Variance Std.Dev.
## id      (Intercept) 0.0000  0.0000
## Residual          0.0187  0.1368
## Number of obs: 30, groups: id, 6
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   8.2982860  0.0789615 20.0000000 105.093 < 2e-16 ***
## time21        0.1726752  0.1116684 20.0000000  1.546  0.1377
## time22       -0.0760513  0.1116684 20.0000000 -0.681  0.5037
## time23        0.5466442  0.1116684 20.0000000  4.895 8.75e-05 ***
## time24        0.1239388  0.1116684 20.0000000  1.110  0.2802
## groupAD       0.6104898  0.1116684 20.0000000  5.467 2.37e-05 ***
## time21:groupAD -0.1864947  0.1579230 20.0000000 -1.181  0.2515
## time22:groupAD  0.3705043  0.1579230 20.0000000  2.346  0.0294 *
## time23:groupAD -0.0001672  0.1579230 20.0000000 -0.001  0.9992
## time24:groupAD  0.1699600  0.1579230 20.0000000  1.076  0.2946
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) time21 time22 time23 time24 gropAD t21:AD t22:AD t23:
AD
## time21      -0.707
## time22      -0.707  0.500
## time23      -0.707  0.500  0.500
## time24      -0.707  0.500  0.500  0.500
## groupAD     -0.707  0.500  0.500  0.500  0.500
## tim21:grpAD  0.500 -0.707 -0.354 -0.354 -0.354 -0.707
## tim22:grpAD  0.500 -0.354 -0.707 -0.354 -0.354 -0.707  0.500
## tim23:grpAD  0.500 -0.354 -0.354 -0.707 -0.354 -0.707  0.500  0.500
## tim24:grpAD  0.500 -0.354 -0.354 -0.354 -0.707 -0.707  0.500  0.500  0.5
00

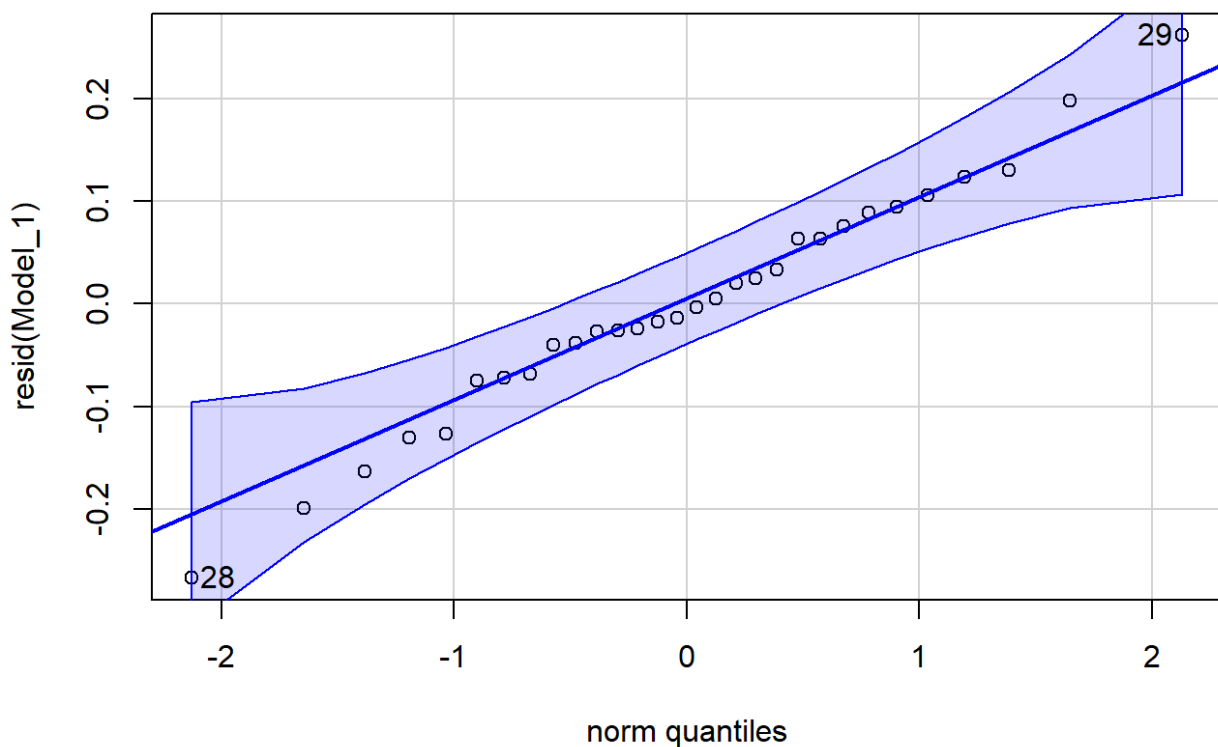
```

```
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
confint(Model_1, level = 0.95, method = "Wald")
```

```
##              2.5 %    97.5 %
## .sig01         NA      NA
## .sigma         NA      NA
## (Intercept)    8.14352429 8.4530477
## time21        -0.04619091 0.3915412
## time22        -0.29491741 0.1428147
## time23         0.32777818 0.7655103
## time24        -0.09492732 0.3428048
## groupAD        0.39162370 0.8293558
## time21:groupAD -0.49601811 0.1230286
## time22:groupAD  0.06098093 0.6800276
## time23:groupAD -0.30969061 0.3093561
## time24:groupAD -0.13956335 0.4794834
```

```
qqPlot(resid(Model_1))
```



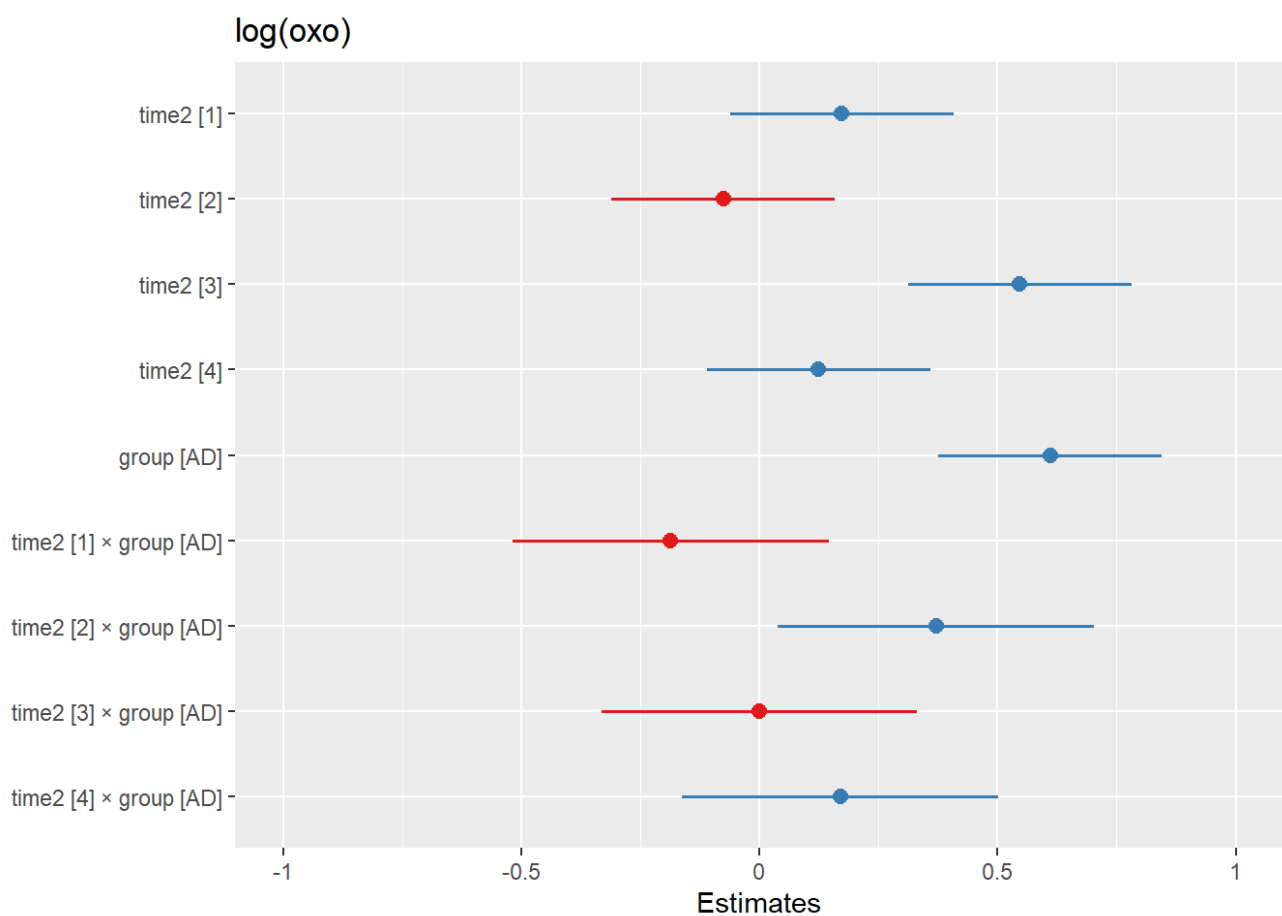
```
## [1] 28 29
```

```
shapiro.test(resid(Model_1))
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  resid(Model_1)  
## W = 0.98881, p-value = 0.9837
```

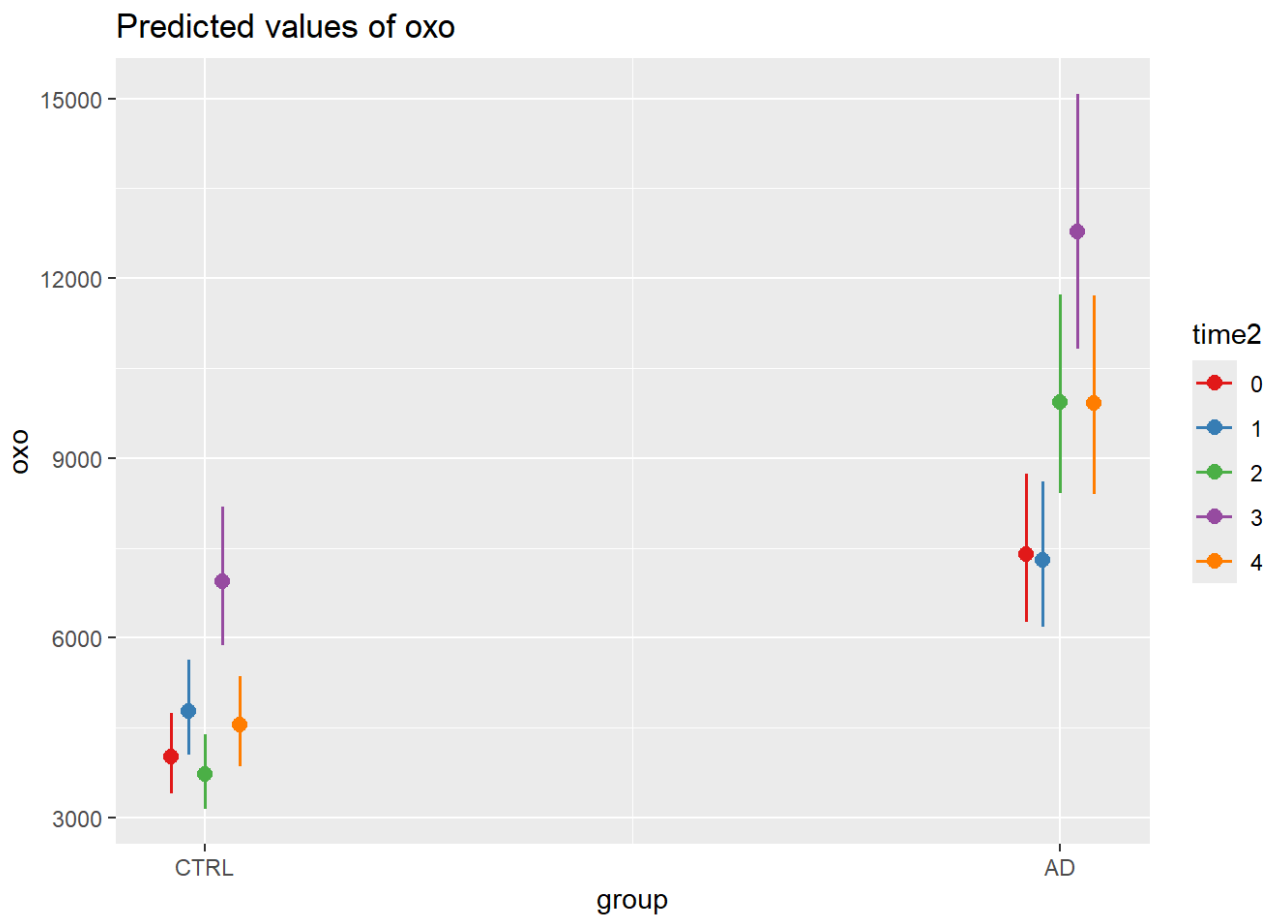
3.1 Visualizzazione del modello

```
plot_model(Model_1)
```



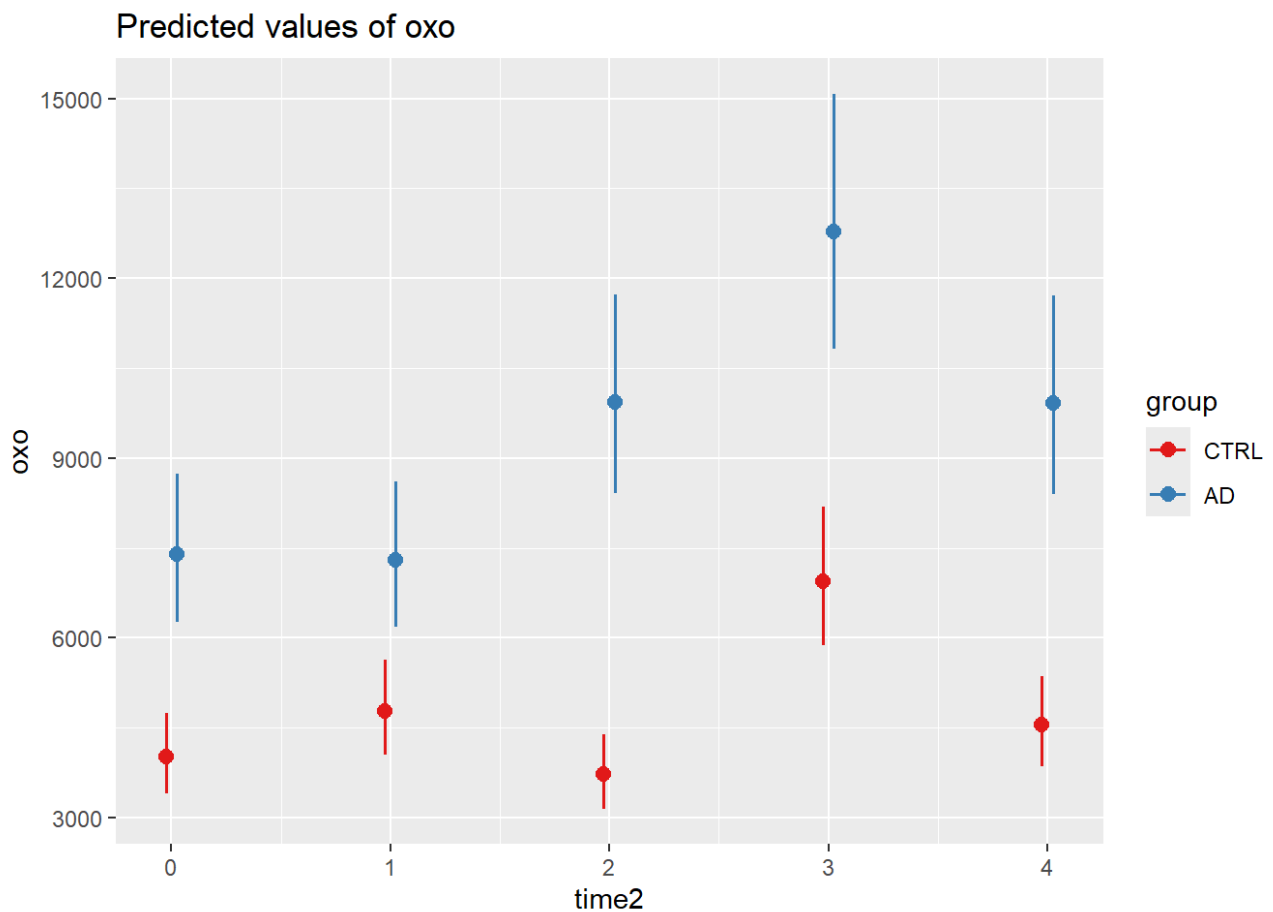
```
plot_model(Model_1,type="pred",  
           terms=c("group", "time2"),  
           pred.type="fe")
```

```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```



```
plot_model(Model_1,type="pred",
            terms=c("time2", "group"),
            pred.type="fe")
```

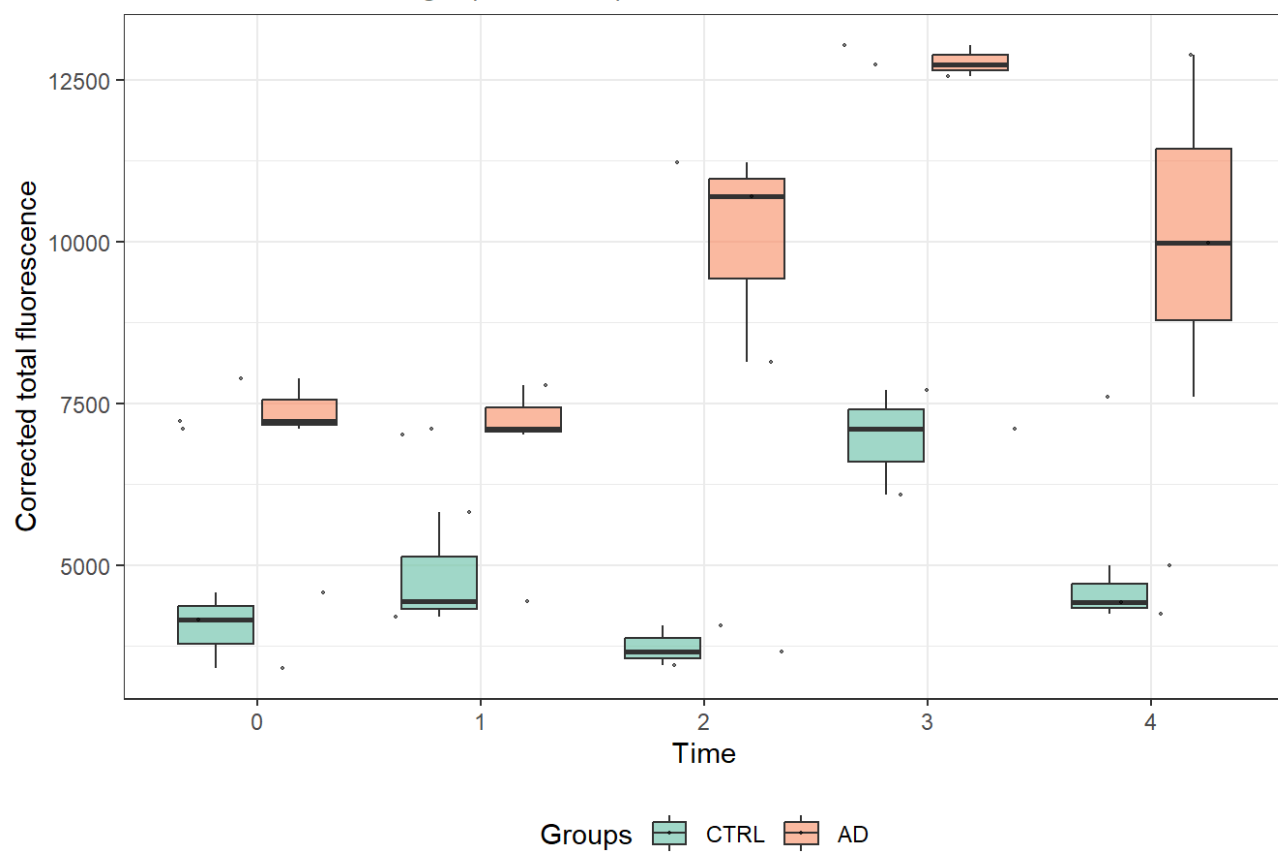
```
## Model has log-transformed response. Back-transforming predictions to
## original response scale. Standard errors are still on the transformed
## scale.
```



Personalizzazione

```
db %>%
  drop_na(time2, group) %>%
  ggplot(aes(y = oxo, x = time2, fill = group)) +
  geom_boxplot(alpha = 0.6, varwidth = TRUE, na.rm = TRUE) +
  geom_jitter(color = "black", alpha = 0.5, size = 0.3, na.rm = TRUE) +
  scale_fill_brewer(palette = "Set2") +
  scale_color_brewer(palette = "Set2") +
  theme_bw(base_family = "sans") +
  theme(legend.position = "bottom") +
  ggtitle("DNA oxidative damage (8-oxo-dG)") +
  xlab("Time") +
  ylab("Corrected total fluorescence") +
  labs(color = "GROUP", fill = "Groups")
```


DNA oxidative damage (8-oxo-dG)



- 1 pacchetti
- 2 Preparazione dati
- 3 Model DNA damage (%)
- 4 Model Tail length

Comet assay

RRF

2026-06-05

1 pacchetti

```
library(tidyverse)
```

```
## Warning: il pacchetto 'tidyverse' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'ggplot2' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tibble' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tidyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'readr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'purrr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'dplyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'stringr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'forcats' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'lubridate' è stato creato con R versione 4.5.1
```

```
## — Attaching core tidyverse packages — tidyverse
2.0.0 —
## ✓ dplyr      1.1.4      ✓ readr      2.1.5
## ✓ forcats    1.0.0      ✓ stringr    1.5.1
## ✓ ggplot2     3.5.2      ✓ tibble     3.3.0
## ✓ lubridate  1.9.4      ✓ tidyr      1.3.1
## ✓ purrr       1.0.4
## — Conflicts — tidyverse_confli
cts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force al
l conflicts to become errors
```

```
library(ggplot2)
library(lme4)
```

```
## Warning: il pacchetto 'lme4' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: Matrix
##
## Caricamento pacchetto: 'Matrix'
##
## I seguenti oggetti sono mascherati da 'package:tidyr':
##
##     expand, pack, unpack
```

```
library(lmerTest)
```

```
## Warning: il pacchetto 'lmerTest' è stato creato con R versione 4.5.1
```

```
##
## Caricamento pacchetto: 'lmerTest'
##
## Il seguente oggetto è mascherato da 'package:lme4':
##
##     lmer
##
## Il seguente oggetto è mascherato da 'package:stats':
##
##     step
```

```
library(sjPlot)
```

```
## Warning: il pacchetto 'sjPlot' è stato creato con R versione 4.5.1
```

```
library(car)
```

```
## Warning: il pacchetto 'car' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: carData
```

```
## Warning: il pacchetto 'carData' è stato creato con R versione 4.5.1
```

```
##  
## Caricamento pacchetto: 'car'  
##  
## Il seguente oggetto è mascherato da 'package:dplyr':  
##  
##      recode  
##  
## Il seguente oggetto è mascherato da 'package:purrr':  
##  
##      some
```

2 Preparazione dati

```
db <- read.csv2("C:/Users/utente2/Desktop/Report PNRR/Comet dataset csv.csv")  
  
db$group <- factor(db$GROUP, levels=c('CTRL','AD'), ordered=FALSE)  
  
db$time2 <- factor(db$TIME, exclude = c("", NA))
```

3 Model DNA damage (%)

```
Model_1 <- lmer(log(X.DNA) ~ time2 + group + group*time2 + (1 | ID),  
               data = db)  
summary(Model_1)
```

```

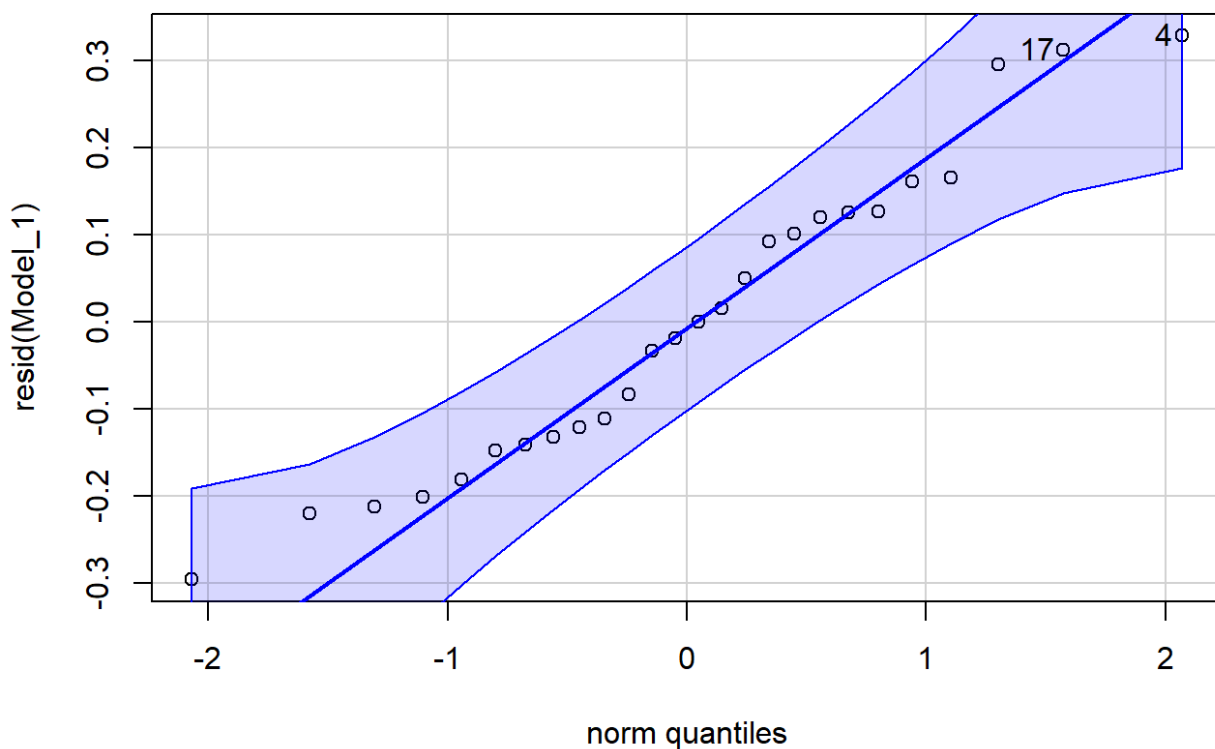
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(X.DNA) ~ time2 + group + group * time2 + (1 | ID)
## Data: db
##
## REML criterion at convergence: 19.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.1897 -0.5583 -0.0369  0.5001  1.3224
##
## Random effects:
## Groups   Name                Variance Std.Dev.
## ID       (Intercept) 0.16290   0.4036
## Residual                  0.06168   0.2484
## Number of obs: 26, groups: ID, 6
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   -1.41686    0.27361   6.10797  -5.178  0.00195 **
## time21         0.10527    0.20278  11.92800   0.519  0.61317
## time22         0.06941    0.20278  11.92800   0.342  0.73810
## time23         0.18857    0.30270  12.16373   0.623  0.54481
## time24         0.65331    0.20278  11.92800   3.222  0.00738 **
## groupAD        -0.45506    0.38694   6.10797  -1.176  0.28336
## time21:groupAD  0.20323    0.28677  11.92800   0.709  0.49214
## time22:groupAD  0.56348    0.28677  11.92800   1.965  0.07315 .
## time23:groupAD  0.99393    0.38150  12.11936   2.605  0.02284 *
## time24:groupAD  0.15978    0.30827  11.99376   0.518  0.61366
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) time21 time22 time23 time24 gropAD t21:AD t22:AD t23:
AD
## time21      -0.371
## time22      -0.371  0.500
## time23      -0.248  0.335  0.335
## time24      -0.371  0.500  0.500  0.335
## groupAD     -0.707  0.262  0.262  0.176  0.262
## tim21:grpAD  0.262 -0.707 -0.354 -0.237 -0.354 -0.371
## tim22:grpAD  0.262 -0.354 -0.707 -0.237 -0.354 -0.371  0.500
## tim23:grpAD  0.197 -0.266 -0.266 -0.793 -0.266 -0.279  0.376  0.376
## tim24:grpAD  0.244 -0.329 -0.329 -0.220 -0.658 -0.345  0.465  0.465  0.3
37

```

```
confint(Model_1, level = 0.95, method = "Wald")
```

```
##                2.5 %    97.5 %  
## .sig01          NA        NA  
## .sigma          NA        NA  
## (Intercept)    -1.953119092 -0.8805992  
## time21         -0.292170769  0.5027024  
## time22         -0.328030607  0.4668425  
## time23         -0.404710495  0.7818577  
## time24          0.255870388  1.0507435  
## groupAD        -1.213450832  0.3033214  
## time21:groupAD -0.358832946  0.7652874  
## time22:groupAD  0.001419594  1.1255400  
## time23:groupAD  0.246209836  1.7416567  
## time24:groupAD -0.444417901  0.7639862
```

```
qqPlot(resid(Model_1))
```



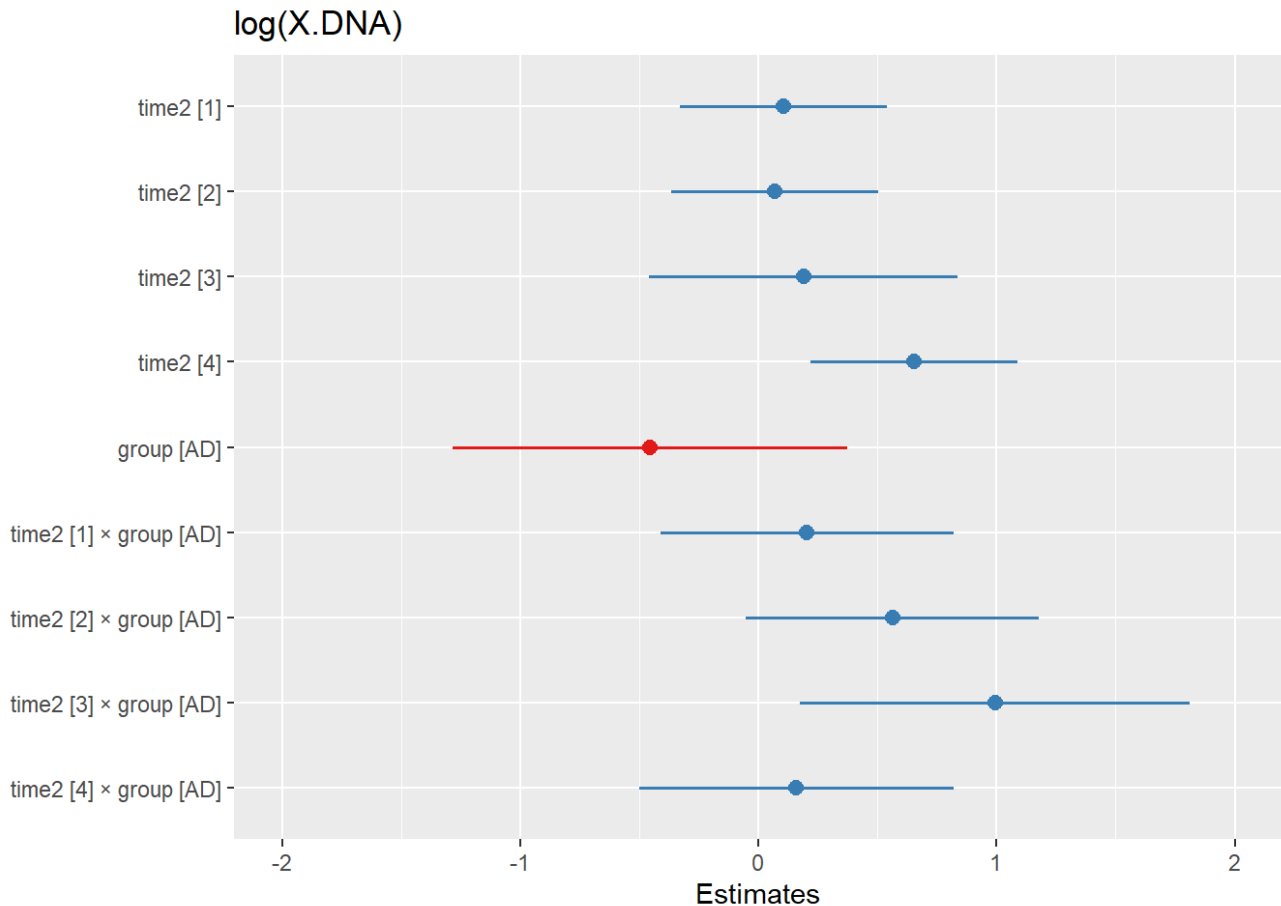
```
## 4 17  
## 4 15
```

```
shapiro.test(resid(Model_1))
```

```
##
## Shapiro-Wilk normality test
##
## data: resid(Model_1)
## W = 0.95794, p-value = 0.3525
```

3.1 Visualizzazione del modello

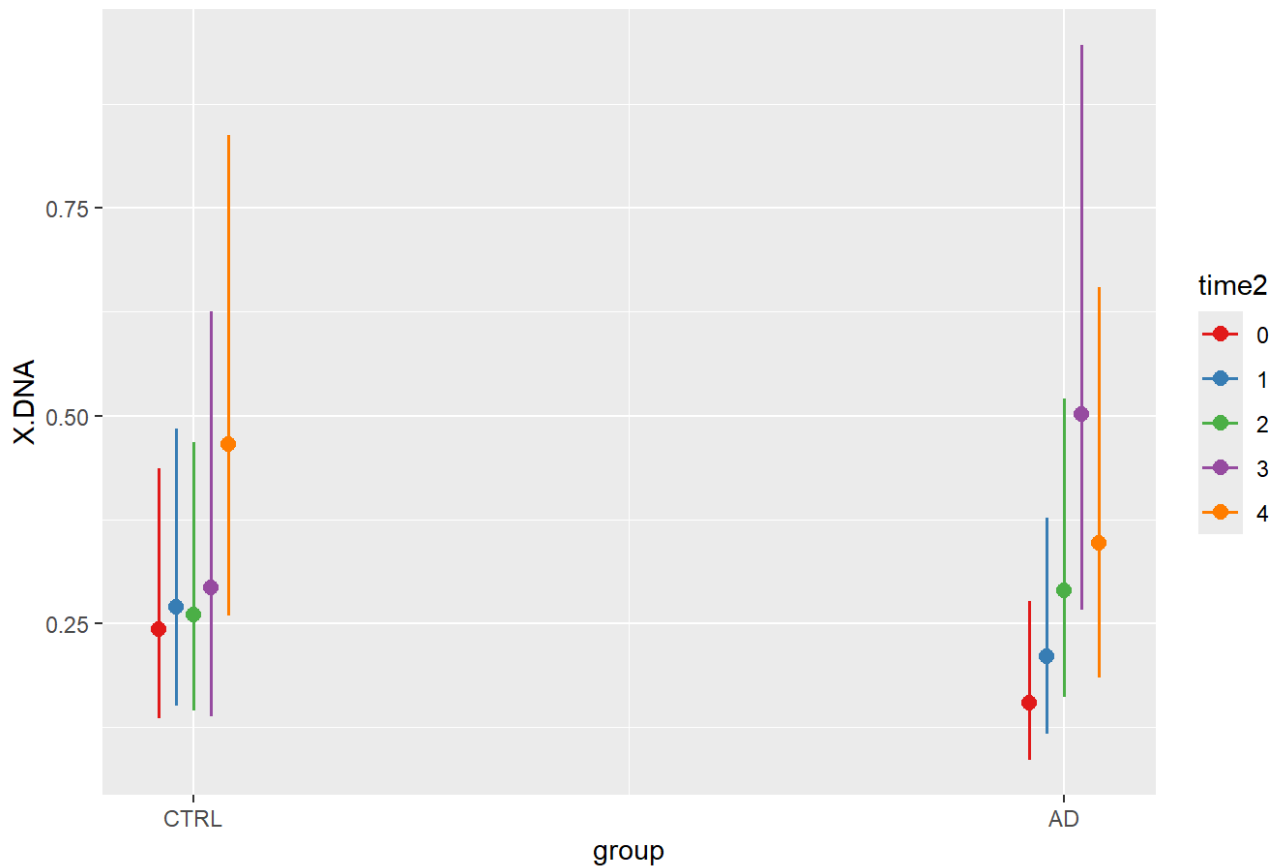
```
plot_model(Model_1)
```



```
plot_model(Model_1,type="pred",
            terms=c("group", "time2"),
            pred.type="fe")
```

```
## Model has log-transformed response. Back-transforming predictions to
## original response scale. Standard errors are still on the transformed
## scale.
```

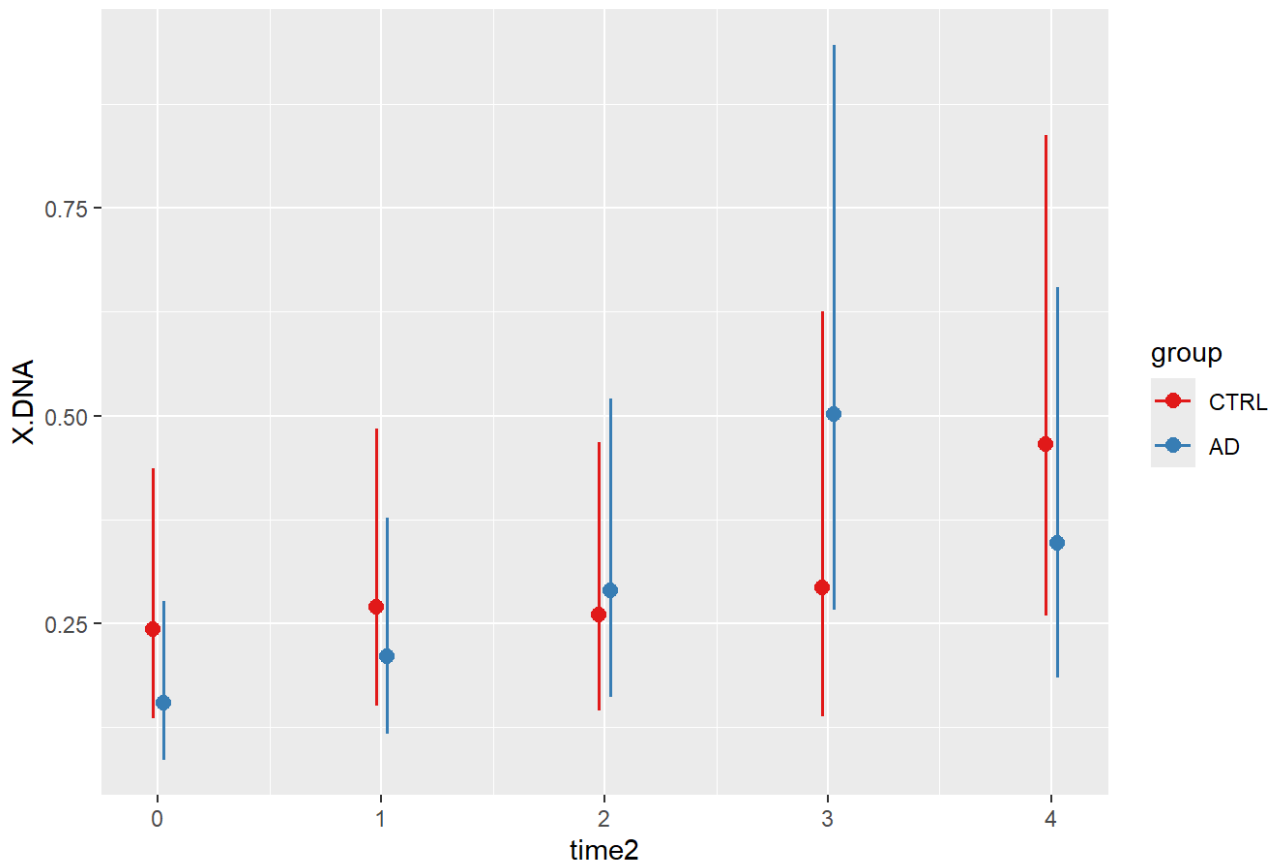
Predicted values of X.DNA



```
plot_model(Model_1,type="pred",
            terms=c("time2", "group"),
            pred.type="fe")
```

```
## Model has log-transformed response. Back-transforming predictions to
## original response scale. Standard errors are still on the transformed
## scale.
```

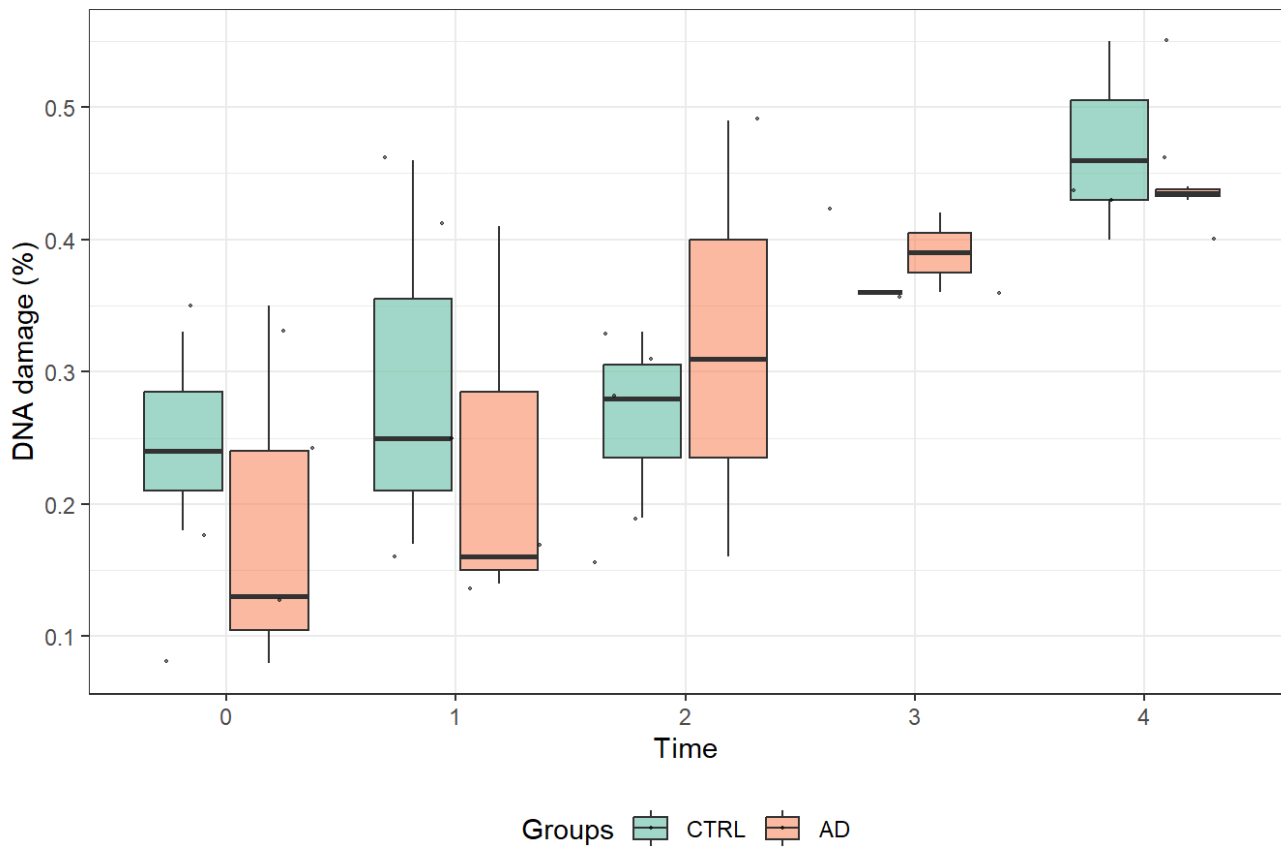

Predicted values of X.DNA



Personalizzazione

```
db %>%
  drop_na(time2, group) %>%
  ggplot(aes(y = X.DNA, x = time2, fill = group)) +
  geom_boxplot(alpha = 0.6, varwidth = TRUE, na.rm = TRUE) +
  geom_jitter(color = "black", alpha = 0.5, size = 0.3, na.rm = TRUE) +
  scale_fill_brewer(palette = "Set2") +
  scale_color_brewer(palette = "Set2") +
  theme_bw(base_family = "sans") +
  theme(legend.position = "bottom") +
  ggtitle("Comet assay") +
  xlab("Time") +
  ylab("DNA damage (%)") +
  labs(color = "GROUP", fill = "Groups")
```

Comet assay



4 Model Tail length

```
Model_1 <- lmer(log(tail.length) ~ time2 + group + group*time2 + (1 | ID),  
                data = db)  
summary(Model_1)
```

```

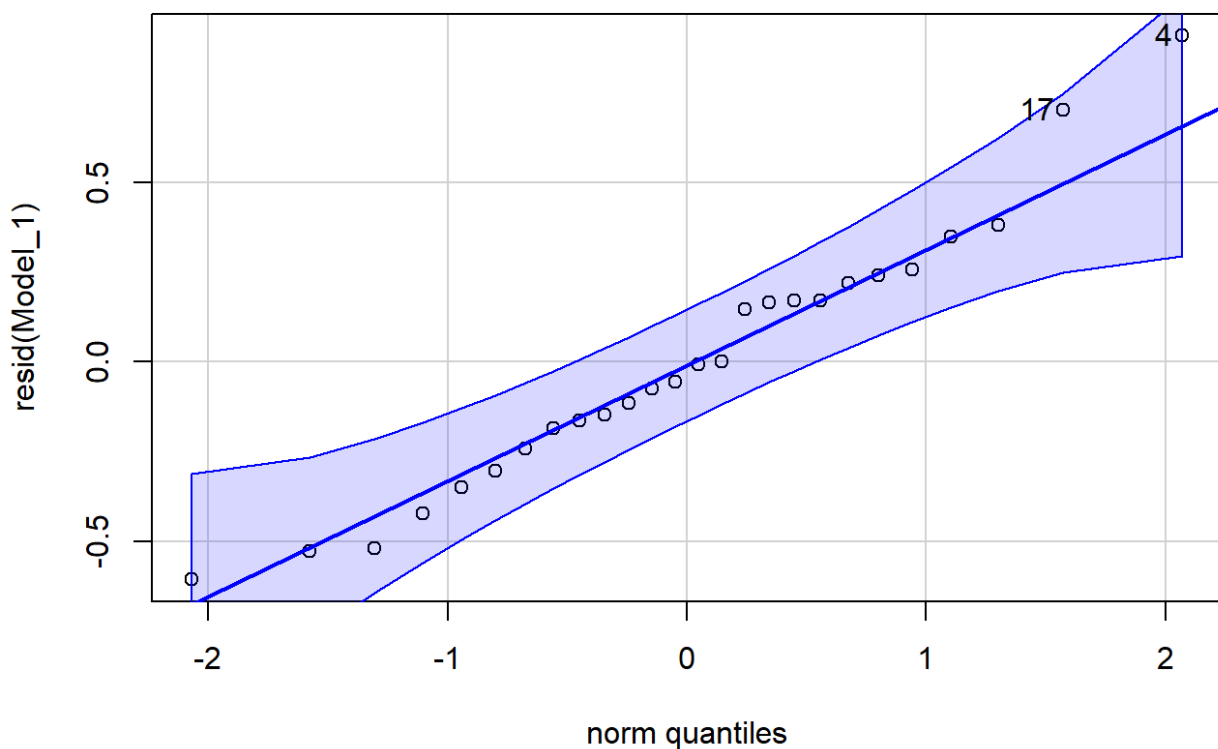
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(tail.length) ~ time2 + group + group * time2 + (1 | ID)
## Data: db
##
## REML criterion at convergence: 49.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.15313 -0.43047 -0.05901  0.39667  1.72993
##
## Random effects:
## Groups Name Variance Std.Dev.
## ID      (Intercept) 3.6642  1.9142
## Residual          0.2764  0.5257
## Number of obs: 26, groups: ID, 6
##
## Fixed effects:
##              Estimate Std. Error    df t value Pr(>|t|)
## (Intercept)    2.74172    1.14609  4.44779    2.392  0.06853 .
## time21         1.00771    0.42923 11.99734    2.348  0.03687 *
## time22         0.98256    0.42923 11.99734    2.289  0.04100 *
## time23         0.98104    0.64319 12.04682    1.525  0.15301
## time24         1.53847    0.42923 11.99734    3.584  0.00375 **
## groupAD         0.96729    1.62081  4.44779    0.597  0.57971
## time21:groupAD -0.37407    0.60703 11.99734   -0.616  0.54925
## time22:groupAD  0.40417    0.60703 11.99734    0.666  0.51812
## time23:groupAD  1.82992    0.81004 12.03768    2.259  0.04322 *
## time24:groupAD  0.03667    0.65324 12.01140    0.056  0.95616
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) time21 time22 time23 time24 gropAD t21:AD t22:AD t23:
AD
## time21      -0.187
## time22      -0.187  0.500
## time23      -0.125  0.334  0.334
## time24      -0.187  0.500  0.500  0.334
## groupAD     -0.707  0.132  0.132  0.088  0.132
## tim21:grpAD  0.132 -0.707 -0.354 -0.236 -0.354 -0.187
## tim22:grpAD  0.132 -0.354 -0.707 -0.236 -0.354 -0.187  0.500
## tim23:grpAD  0.099 -0.265 -0.265 -0.794 -0.265 -0.140  0.375  0.375
## tim24:grpAD  0.123 -0.329 -0.329 -0.219 -0.657 -0.174  0.465  0.465  0.3
35

```

```
confint(Model_1, level = 0.95, method = "Wald")
```

```
##                2.5 %    97.5 %  
## .sig01          NA      NA  
## .sigma          NA      NA  
## (Intercept)    0.4954214 4.9880088  
## time21         0.1664294 1.8489922  
## time22         0.1412777 1.8238405  
## time23        -0.2795861 2.2416620  
## time24         0.6971875 2.3797503  
## groupAD        -2.2094536 4.1440243  
## time21:groupAD -1.5638248 0.8156784  
## time22:groupAD -0.7855806 1.5939226  
## time23:groupAD  0.2422637 3.4175743  
## time24:groupAD -1.2436565 1.3169926
```

```
qqPlot(resid(Model_1))
```



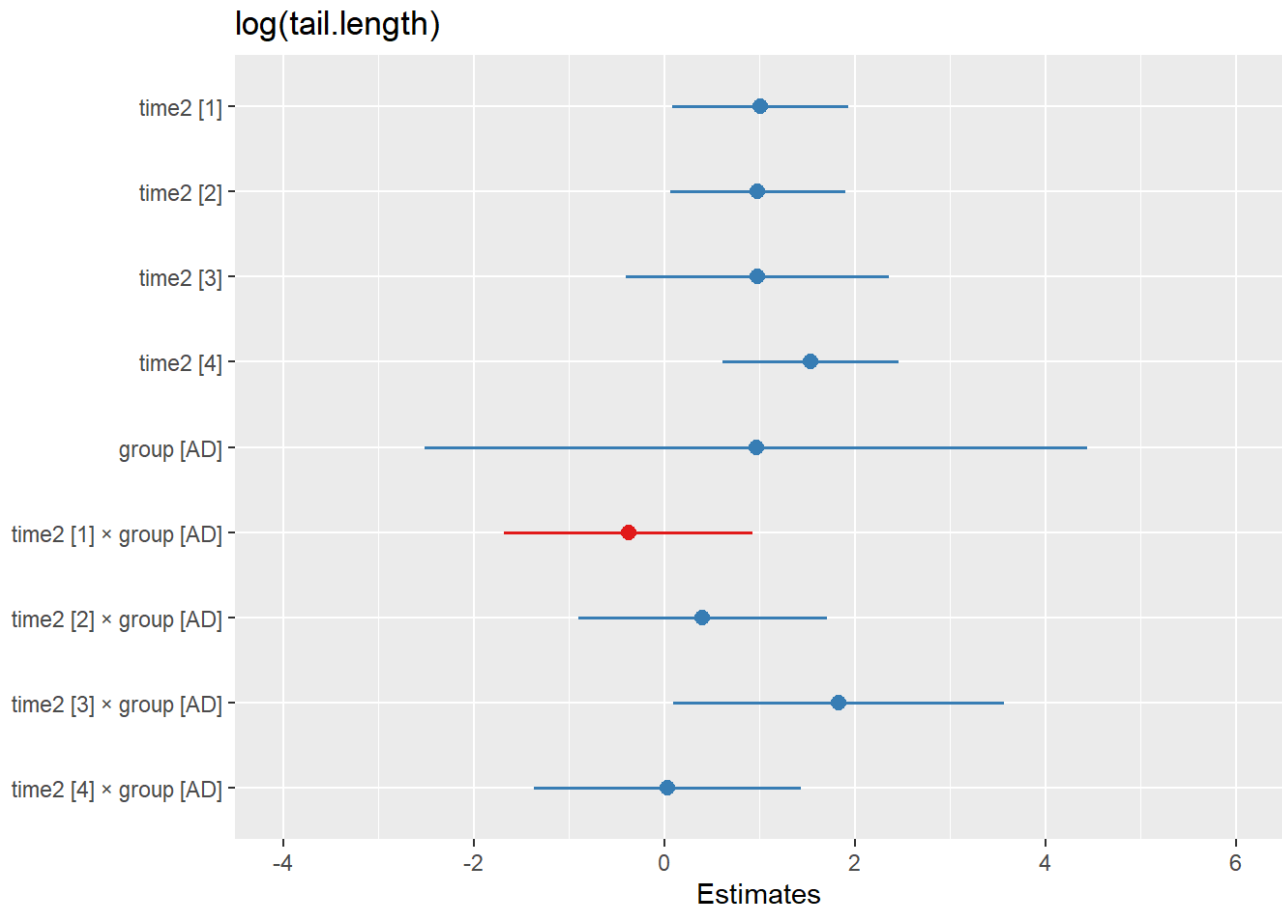
```
## 4 17  
## 4 15
```

```
shapiro.test(resid(Model_1))
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: resid(Model_1)  
## W = 0.96895, p-value = 0.5962
```

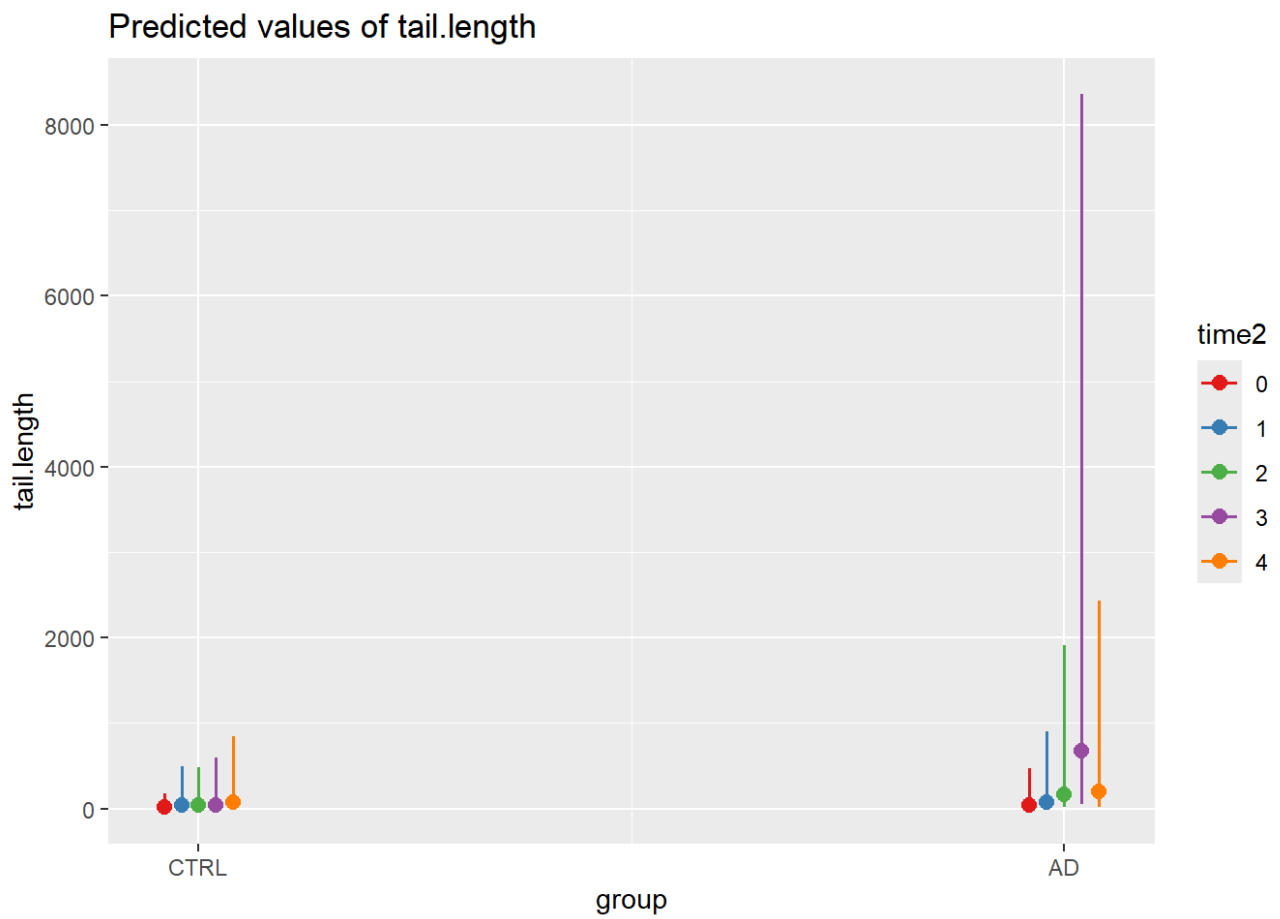
4.1 Visualizzazione del modello

```
plot_model(Model_1)
```



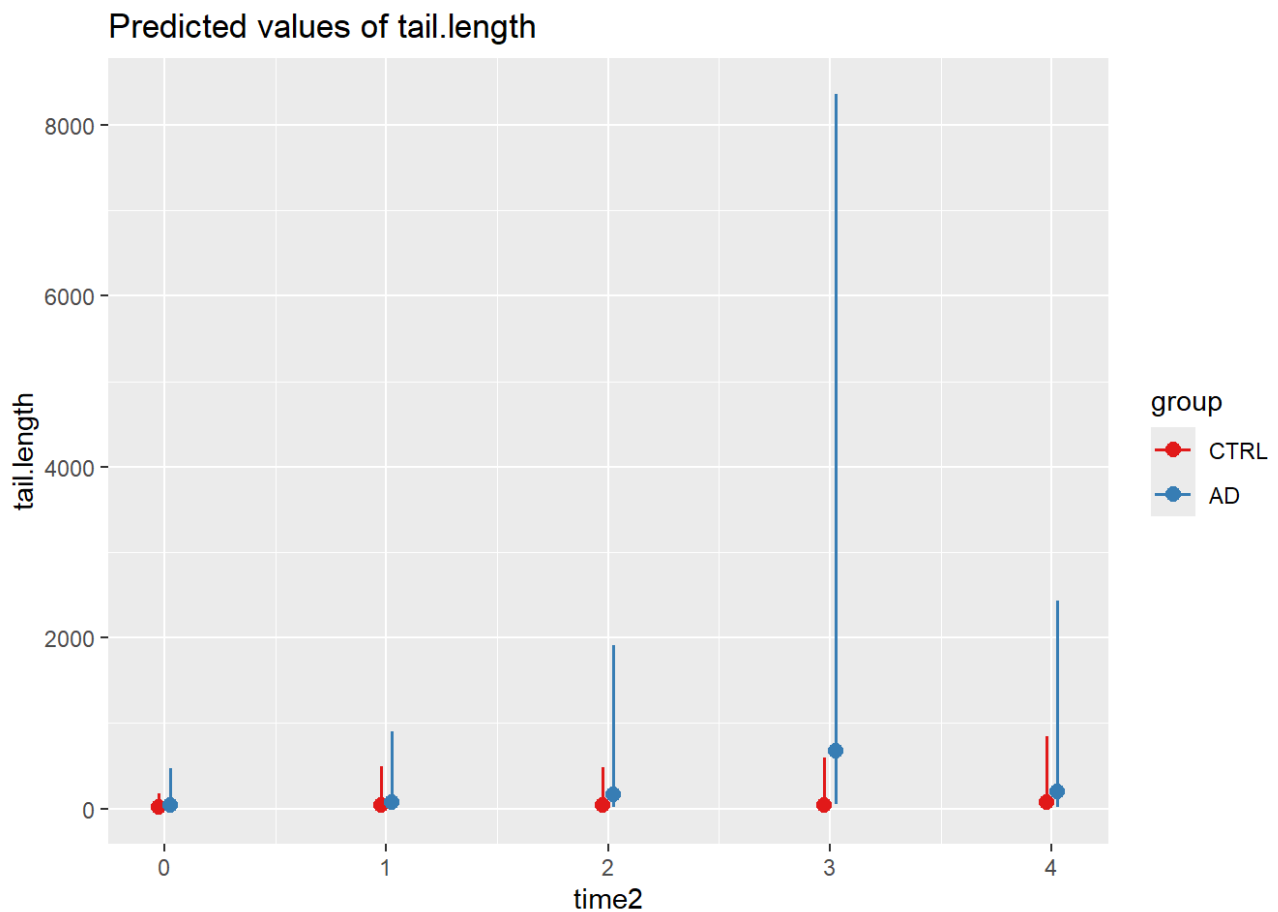
```
plot_model(Model_1,type="pred",  
            terms=c("group", "time2"),  
            pred.type="fe")
```

```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```



```
plot_model(Model_1,type="pred",  
            terms=c("time2", "group"),  
            pred.type="fe")
```

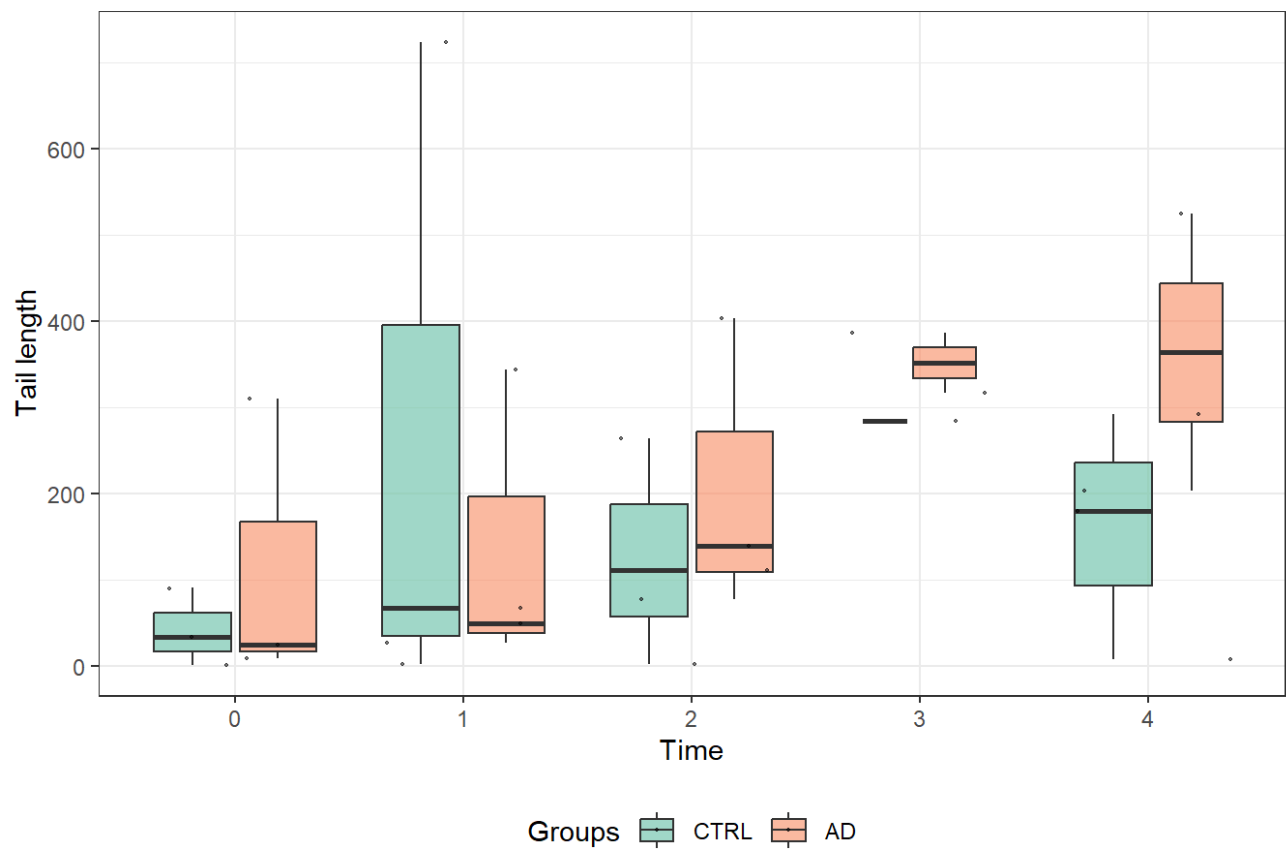
```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```



Personalizzazione

```
db %>%
  drop_na(time2, group) %>%
  ggplot(aes(y = tail.length, x = time2, fill = group)) +
  geom_boxplot(alpha = 0.6, varwidth = TRUE, na.rm = TRUE) +
  geom_jitter(color = "black", alpha = 0.5, size = 0.3, na.rm = TRUE) +
  scale_fill_brewer(palette = "Set2") +
  scale_color_brewer(palette = "Set2") +
  theme_bw(base_family = "sans") +
  theme(legend.position = "bottom") +
  ggtitle("Comet assay") +
  xlab("Time") +
  ylab("Tail length") +
  labs(color = "GROUP", fill = "Groups")
```

Comet assay



- 1 pacchetti
- 2 Preparazione dati
- 3 Model plasmatic mtDNA

Plasmatic mtDNA

RRF

2026-06-05

1 pacchetti

```
library(tidyverse)
```

```
## Warning: il pacchetto 'tidyverse' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'ggplot2' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tibble' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'tidyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'readr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'purrr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'dplyr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'stringr' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'forcats' è stato creato con R versione 4.5.1
```

```
## Warning: il pacchetto 'lubridate' è stato creato con R versione 4.5.1
```

```
## — Attaching core tidyverse packages — tidyverse
2.0.0 —
## ✓ dplyr      1.1.4      ✓ readr      2.1.5
## ✓ forcats   1.0.0      ✓ stringr    1.5.1
## ✓ ggplot2    3.5.2      ✓ tibble     3.3.0
## ✓ lubridate  1.9.4      ✓ tidyr      1.3.1
## ✓ purrr      1.0.4
## — Conflicts — tidyverse_confli
cts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force al
l conflicts to become errors
```

```
library(ggplot2)
library(lme4)
```

```
## Warning: il pacchetto 'lme4' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: Matrix
##
## Caricamento pacchetto: 'Matrix'
##
## I seguenti oggetti sono mascherati da 'package:tidyr':
##
##     expand, pack, unpack
```

```
library(lmerTest)
```

```
## Warning: il pacchetto 'lmerTest' è stato creato con R versione 4.5.1
```

```
##
## Caricamento pacchetto: 'lmerTest'
##
## Il seguente oggetto è mascherato da 'package:lme4':
##
##     lmer
##
## Il seguente oggetto è mascherato da 'package:stats':
##
##     step
```

```
library(sjPlot)
```

```
## Warning: il pacchetto 'sjPlot' è stato creato con R versione 4.5.1
```

```
library(car)
```

```
## Warning: il pacchetto 'car' è stato creato con R versione 4.5.1
```

```
## Caricamento del pacchetto richiesto: carData
```

```
## Warning: il pacchetto 'carData' è stato creato con R versione 4.5.1
```

```
##  
## Caricamento pacchetto: 'car'  
##  
## Il seguente oggetto è mascherato da 'package:dplyr':  
##  
##      recode  
##  
## Il seguente oggetto è mascherato da 'package:purrr':  
##  
##      some
```

2 Preparazione dati

```
db <- read.csv2("C:/Users/utente2/Desktop/Report PNRR/Dataset mtdna csv.csv")  
  
db$group <- factor(db$group, levels=c('CTRL','MCI','AD'), ordered=FALSE)  
  
db$time2 <- factor(db$time, exclude = c("", NA))
```

3 Model plasmatic mtDNA

```
Model_1 <- lmer(log(mtdna) ~ time2 + group + group*time2 + (1 | id),  
               data = db)  
summary(Model_1)
```

```

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: log(mtdna) ~ time2 + group + group * time2 + (1 | id)
## Data: db
##
## REML criterion at convergence: 492.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.1573 -0.5459  0.1409  0.6993  1.8482
##
## Random effects:
## Groups   Name                Variance Std.Dev.
## id       (Intercept) 0.06848  0.2617
## Residual                    1.01824  1.0091
## Number of obs: 168, groups: id, 84
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    4.1134    0.1903 161.3593  21.613  <2e-16 ***
## time21         0.3600    0.2605  81.0000   1.382   0.171
## groupMCI       0.2618    0.2793 161.3593   0.937   0.350
## groupAD        0.2189    0.2739 161.3593   0.799   0.425
## time21:groupMCI -0.2377    0.3824  81.0000  -0.622   0.536
## time21:groupAD  -0.3951    0.3750  81.0000  -1.054   0.295
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) time21 grpMCI gropAD t21:MC
## time21      -0.684
## groupMCI     -0.681  0.466
## groupAD      -0.695  0.476  0.473
## tm21:grpMCI  0.466 -0.681 -0.684 -0.324
## tim21:grpAD  0.476 -0.695 -0.324 -0.684  0.473

```

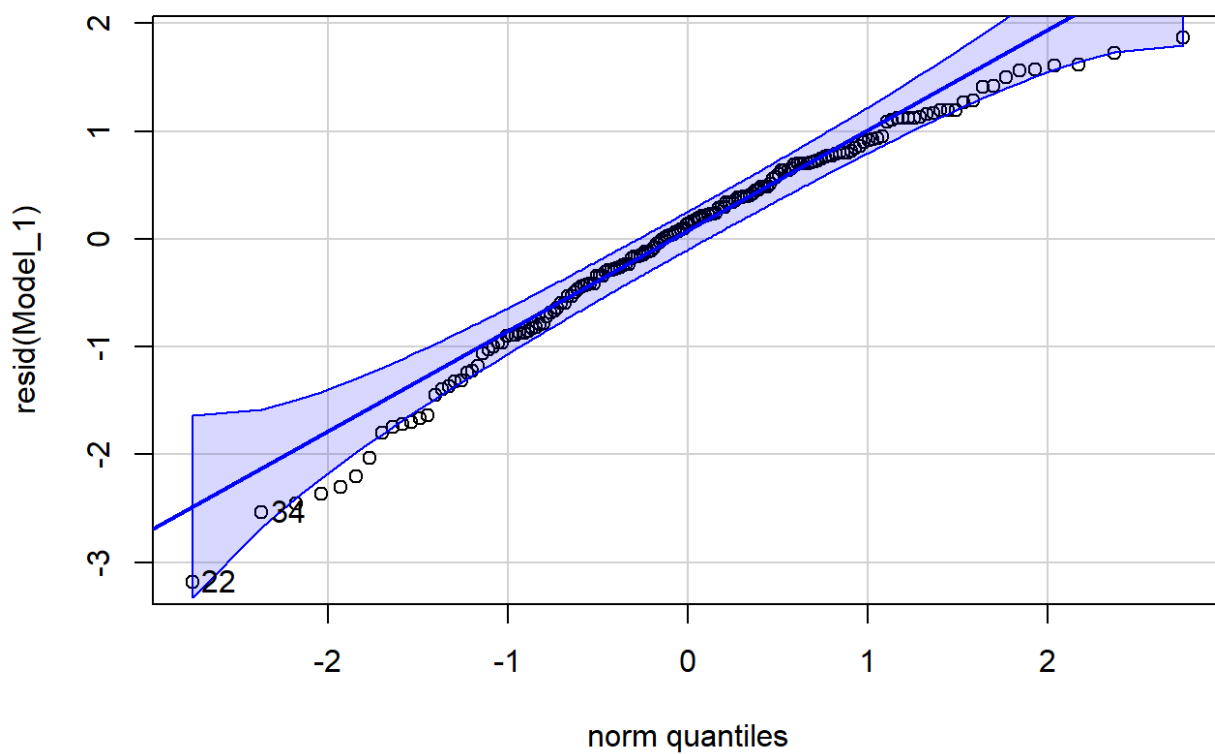
```

confint(Model_1, level = 0.95, method = "Wald")

```

```
##                2.5 %    97.5 %
## .sig01          NA      NA
## .sigma          NA      NA
## (Intercept)    3.7404135 4.4864758
## time21         -0.1506589 0.8706502
## groupMCI        -0.2856432 0.8092769
## groupAD         -0.3179733 0.7557938
## time21:groupMCI -0.9871065 0.5117655
## time21:groupAD -1.1300586 0.3398563
```

```
qqPlot(resid(Model_1))
```



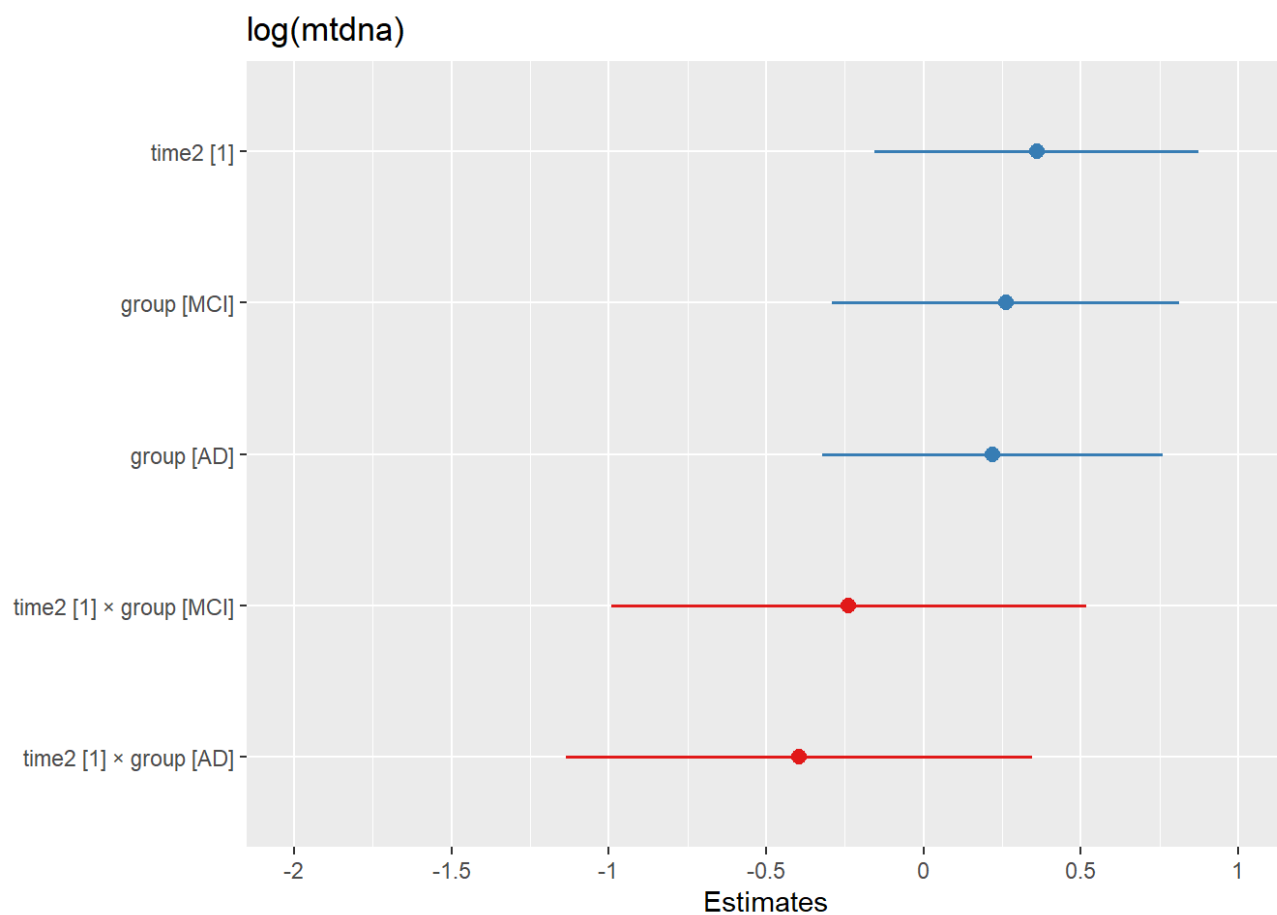
```
## [1] 22 34
```

```
shapiro.test(resid(Model_1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(Model_1)
## W = 0.97143, p-value = 0.001531
```

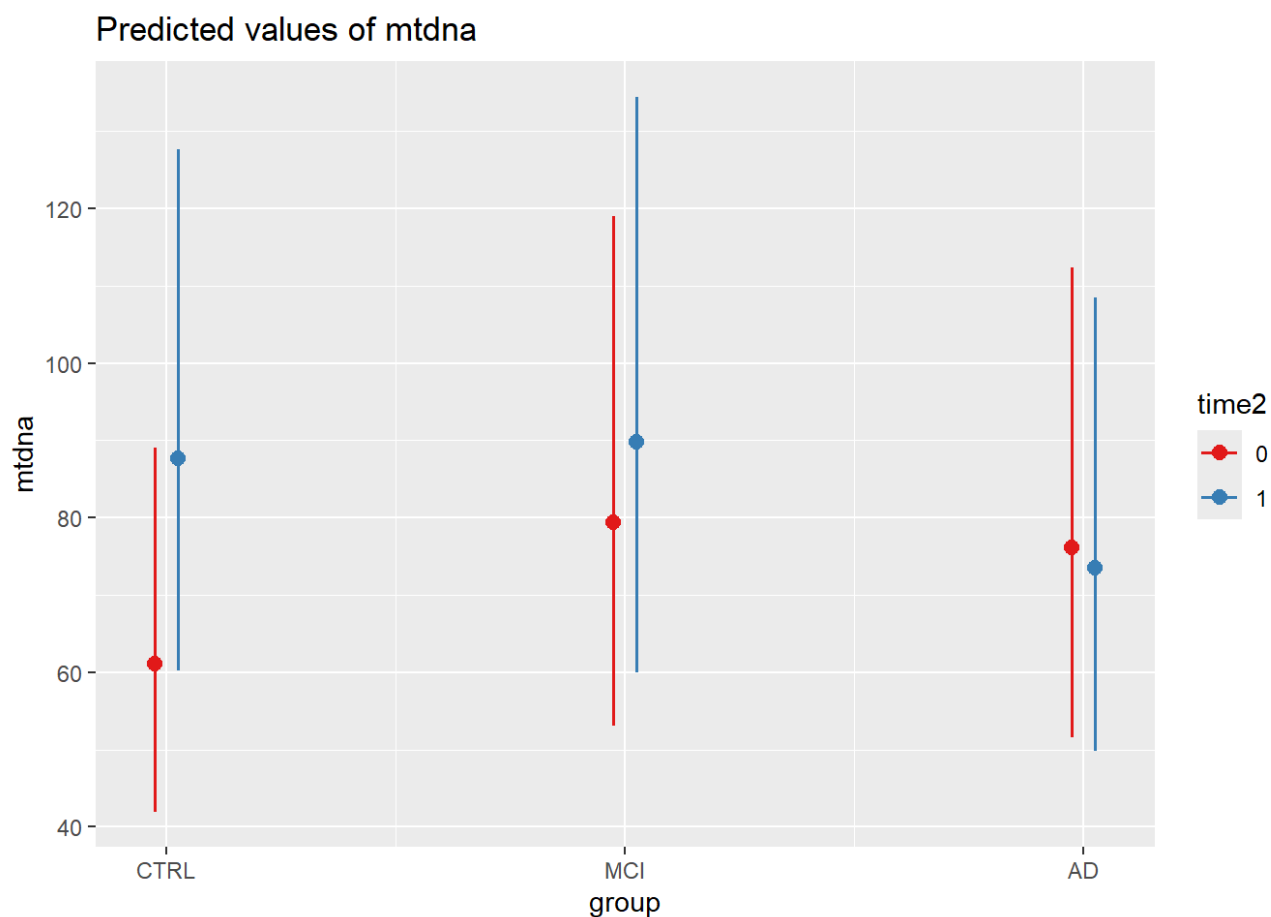
3.1 Visualizzazione del modello

```
plot_model(Model_1)
```



```
plot_model(Model_1,type="pred",  
           terms=c("group", "time2"),  
           pred.type="fe")
```

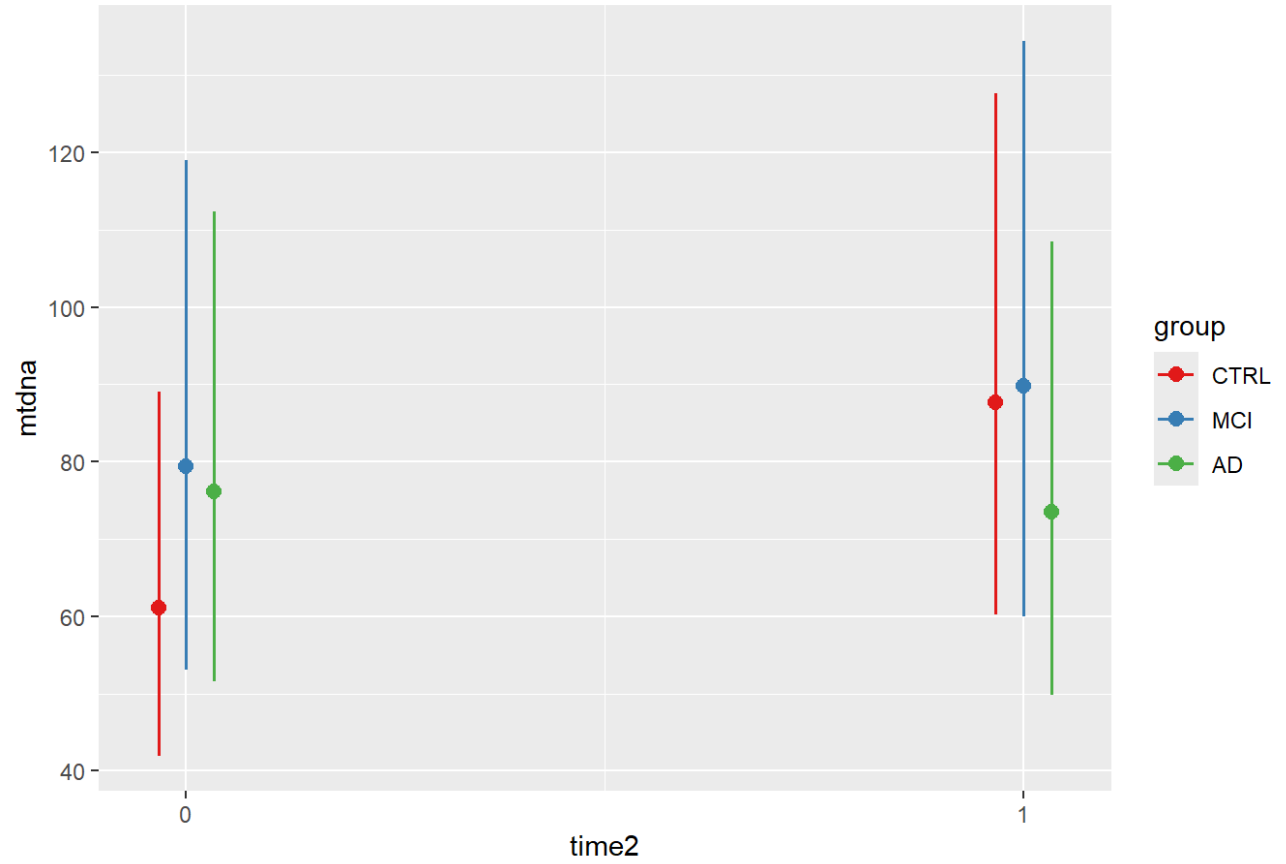
```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```



```
plot_model(Model_1,type="pred",  
            terms=c("time2", "group"),  
            pred.type="fe")
```

```
## Model has log-transformed response. Back-transforming predictions to  
## original response scale. Standard errors are still on the transformed  
## scale.
```

Predicted values of mtdna



- 1 pacchetti
- 2 Preparazione dati
- 3 Model C5a
- 4 Model ccl3
- 5 Model ccl4
- 6 Model cx3cl1
- 7 Model cxcl10
- 8 Model cxcl16
- 9 Model ifny
- 10 Model il1a
- 11 Model il1b
- 12 Model il33
- 13 Model il6
- 14 Model il10
- 15 Model tnfa
- 16 Model FABP3
- 17 Model Ferritin
- 18 Model Neurogranin
- 19 Model TREM2

AnalisiLuminexAdvancedExtreme

MR

2026-03-06

1 pacchetti

```
library(readxl)
library(tableone)
```

```
## Warning: il pacchetto 'tableone' è stato creato con R versione 4.5.2
```

```
library(dplyr)
```

```
##  
## Caricamento pacchetto: 'dplyr'
```

```
## I seguenti oggetti sono mascherati da 'package:stats':  
##  
##      filter, lag
```

```
## I seguenti oggetti sono mascherati da 'package:base':  
##  
##      intersect, setdiff, setequal, union
```

```
library(tibble)  
library(openxlsx)
```

```
## Warning: il pacchetto 'openxlsx' è stato creato con R versione 4.5.2
```

```
library(car)
```

```
## Caricamento del pacchetto richiesto: carData
```

```
##  
## Caricamento pacchetto: 'car'
```

```
## Il seguente oggetto è mascherato da 'package:dplyr':  
##  
##      recode
```

```
library(ggplot2)
```

```
## Warning: il pacchetto 'ggplot2' è stato creato con R versione 4.5.2
```

```
library(tidyr)  
library(tidyverse)
```

```
## Warning: il pacchetto 'readr' è stato creato con R versione 4.5.2
```

```
## Warning: il pacchetto 'stringr' è stato creato con R versione 4.5.2
```

```
## Warning: il pacchetto 'forcats' è stato creato con R versione 4.5.2
```

```
## — Attaching core tidyverse packages — tidyverse
2.0.0 —
## ✓ forcats 1.0.1 ✓ readr 2.1.6
## ✓ lubridate 1.9.4 ✓ stringr 1.6.0
## ✓ purrr 1.0.4
```

```
## — Conflicts — tidyverse_confli
cts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag() masks stats::lag()
## ✗ car::recode() masks dplyr::recode()
## ✗ purrr::some() masks car::some()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force al
l conflicts to become errors
```

```
library(lme4)
```

```
## Caricamento del pacchetto richiesto: Matrix
##
## Caricamento pacchetto: 'Matrix'
##
## I seguenti oggetti sono mascherati da 'package:tidyr':
##
## expand, pack, unpack
```

```
library(lmerTest)
```

```
##
## Caricamento pacchetto: 'lmerTest'
##
## Il seguente oggetto è mascherato da 'package:lme4':
##
## lmer
##
## Il seguente oggetto è mascherato da 'package:stats':
##
## step
```

```
library(sjPlot)
```

```
## Warning: il pacchetto 'sjPlot' è stato creato con R versione 4.5.2
```

```
##  
## Caricamento pacchetto: 'sjPlot'  
##  
## Il seguente oggetto è mascherato da 'package:ggplot2':  
##  
##      set_theme
```

```
library(emmeans)
```

```
## Warning: il pacchetto 'emmeans' è stato creato con R versione 4.5.2
```

```
## Welcome to emmeans.  
## Caution: You lose important information if you filter this package's results.  
## See '? untidy'
```

```
library(clubSandwich)
```

```
## Warning: il pacchetto 'clubSandwich' è stato creato con R versione 4.5.2
```

```
## Registered S3 method overwritten by 'clubSandwich':  
##   method      from  
##   bread.mlm    sandwich
```

```
library(ggeffects)
```

```
## Warning: il pacchetto 'ggeffects' è stato creato con R versione 4.5.2
```

```
library(rmcorr)
```

```
## Warning: il pacchetto 'rmcorr' è stato creato con R versione 4.5.2
```

```
library(ggrepel)  
library(patchwork)
```

```
## Warning: il pacchetto 'patchwork' è stato creato con R versione 4.5.2
```

```
library(ggpubr)
```

```
## Warning: il pacchetto 'ggpubr' è stato creato con R versione 4.5.2
```

```
library(lattice)
library(readxl)
library(igraph)
```

```
## Warning: il pacchetto 'igraph' è stato creato con R versione 4.5.2
```

```
##
## Caricamento pacchetto: 'igraph'
##
## I seguenti oggetti sono mascherati da 'package:lubridate':
##
##    %--%, union
##
## I seguenti oggetti sono mascherati da 'package:purrr':
##
##    compose, simplify
##
## Il seguente oggetto è mascherato da 'package:tidyr':
##
##    crossing
##
## Il seguente oggetto è mascherato da 'package:tibble':
##
##    as_data_frame
##
## I seguenti oggetti sono mascherati da 'package:dplyr':
##
##    as_data_frame, groups, union
##
## I seguenti oggetti sono mascherati da 'package:stats':
##
##    decompose, spectrum
##
## Il seguente oggetto è mascherato da 'package:base':
##
##    union
```

```
library(qgraph)
library(psych)
```

```
## Warning: il pacchetto 'psych' è stato creato con R versione 4.5.2
```

```
##
## Caricamento pacchetto: 'psych'
##
## I seguenti oggetti sono mascherati da 'package:ggplot2':
##
##      %+%, alpha
##
## Il seguente oggetto è mascherato da 'package:car':
##
##      logit
```

```
library(bootnet)
```

```
## Warning: il pacchetto 'bootnet' è stato creato con R versione 4.5.2
```

```
## This is bootnet 1.6
## For questions and issues, please see github.com/SachaEpskamp/bootnet.
```

```
library(glmmTMB)
```

```
## Warning: il pacchetto 'glmmTMB' è stato creato con R versione 4.5.2
```

```
## Warning in check_dep_version(dep_pkg = "TMB"): package version mismatch:
## glmmTMB was built with TMB package version 1.9.19
## Current TMB package version is 1.9.17
## Please re-install glmmTMB from source or restore original 'TMB' package
## (see '?reinstalling' for more information)
```



```
library(DHARMa)
```

```
## This is DHARMa 0.4.7. For overview type '?DHARMa'. For recent changes, t
## ype news(package = 'DHARMa')
```

```
library(nlme)
```

```
##
## Caricamento pacchetto: 'nlme'
##
## Il seguente oggetto è mascherato da 'package:lme4':
##
##      lmlist
##
## Il seguente oggetto è mascherato da 'package:dplyr':
##
##      collapse
```

```
library(lmtest)
```

```
## Caricamento del pacchetto richiesto: zoo
```

```
## Warning: il pacchetto 'zoo' è stato creato con R versione 4.5.2
```

```
##
## Caricamento pacchetto: 'zoo'
##
## I seguenti oggetti sono mascherati da 'package:base':
##
##      as.Date, as.Date.numeric
```

2 Preparazione dati

```
db <- read_excel("W:/LAB NEUROBIOLOGIA/Statistica MR/Statistica paper Lumin  
ex Riccardo/Analisi Luminex/Dataset paper_3T_anni.xlsx", sheet = "alfredo  
3")

db$GROUP <- factor(db$GROUP, levels=c('0','1','2','3'), ordered=FALSE)
db$t <- factor(db$t, exclude = c("", NA))
#db$xxx <- as.numeric(db$MMSE)
#db$xxx <- as.numeric(db$Met)
#db$xxx <- as.numeric(db$xxx)

db_t0 <- db %>%
  filter(t == "0")
head(db_t0)
```

```
## # A tibble: 6 × 25
##       id GROUP Sex      Age t      C5a  ccl3  ccl4 cx3cl1 cxcl10 cxcl16
ifny
##   <dbl> <fct> <chr> <dbl> <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
<dbl>
## 1    17 1      M      75 0    4152.  14.9  108.  284.  11.2  851.
0.696
## 2    22 1      F      71 0      NA    NA    NA    NA    NA    NA  NA  N
A
## 3    34 1      F      79 0    2659.  33.1  107.  405.  48.1 1652.
1.18
## 4    37 1      F      71 0      NA    NA    NA    NA    NA    NA  NA  N
A
## 5    55 1      F      75 0    7938.  14.5  103.  278.  11.9 1385.
0.997
## 6   151 1      F      71 0      NA    NA    NA    NA    NA    NA  NA  N
A
## # i 13 more variables: il1a <dbl>, il1b <dbl>, il33 <dbl>, il6 <dbl>,
## #   il10 <dbl>, tnfa <dbl>, FABP3 <dbl>, Ferritin <dbl>, Neurogranin <db
l>,
## #   TREM2 <dbl>, wave <dbl>, CIRS_comorb_0 <dbl>, CIRS_severity_0 <dbl>
```

```
db_t1 <- db %>%
  filter(t == "1")
head(db_t1)
```



```
## # A tibble: 6 × 25
##       id GROUP Sex      Age t          C5a  ccl3  ccl4 cx3cl1 cxcl10 cxcl16
ifny
##   <dbl> <fct> <chr> <dbl> <fct>  <dbl> <dbl> <dbl>  <dbl>  <dbl>  <dbl>
<dbl>
## 1    17 1      M       79 1      6360.  26.0 132.    619.  39.5   767.
9.31
## 2    22 1      F       75 1      1083.  30.0 104.    513.   7.33  780.
NA
## 3    34 1      F       83 1      2299.  28.2  77.1    556.  20.2  1319.
3.81
## 4    37 1      F       75 1     12088.  38.5 122.    742.  23.9   698.
1.31
## 5    55 1      F       79 1     10255.  26.0 138.    613.  32.3  1190.
11.0
## 6   151 1      F       75 1      1092.  47.4 143.    786.  NA    1130.
16.6
## # i 13 more variables: il1a <dbl>, il1b <dbl>, il33 <dbl>, il6 <dbl>,
## #   il10 <dbl>, tnfa <dbl>, FABP3 <dbl>, Ferritin <dbl>, Neurogranin <db
l>,
## #   TREM2 <dbl>, wave <dbl>, CIRS_comorb_0 <dbl>, CIRS_severity_0 <dbl>
```

```
db_t2 <- db %>%
  filter(t == "2")
head(db_t2)
```

```
## # A tibble: 6 × 25
##      id GROUP Sex      Age t          C5a  ccl3  ccl4 cx3cl1 cxcl10 cxcl16
ifny
##    <dbl> <fct> <chr> <dbl> <fct>  <dbl> <dbl> <dbl>  <dbl>  <dbl>  <dbl>
<dbl>
## 1    17  1      M      83  2      2000.  24.0  72.3   251.   28.1   421.
15.7
## 2    22  1      F      79  2      1199.  31.7 104.    500.    8.53   608.
NA
## 3    34  1      F      87  2      2280.  34.5  78.1   679.   15.7  1196.
15.7
## 4    37  1      F      79  2      12001.  37.3 125.    658.   59.8   680.
6.60
## 5    55  1      F      83  2      12870.  24.7  83.3   314.   28.0   578.
NA
## 6   151  1      F      79  2        919.  28.3  79.7   257.    NA   819.
3.30
## # i 13 more variables: il1a <dbl>, il1b <dbl>, il33 <dbl>, il6 <dbl>,
## #   il10 <dbl>, tnfa <dbl>, FABP3 <dbl>, Ferritin <dbl>, Neurogranin <db
l>,
## #   TREM2 <dbl>, wave <dbl>, CIRS_comorb_0 <dbl>, CIRS_severity_0 <dbl>
```

3 Model C5a

```
bptest(lm(C5a ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | i
d), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(C5a ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
(1 | id), data = db)
## BP = 30.2, df = 14, p-value = 0.007163
```

```
vars <- c("C5a","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

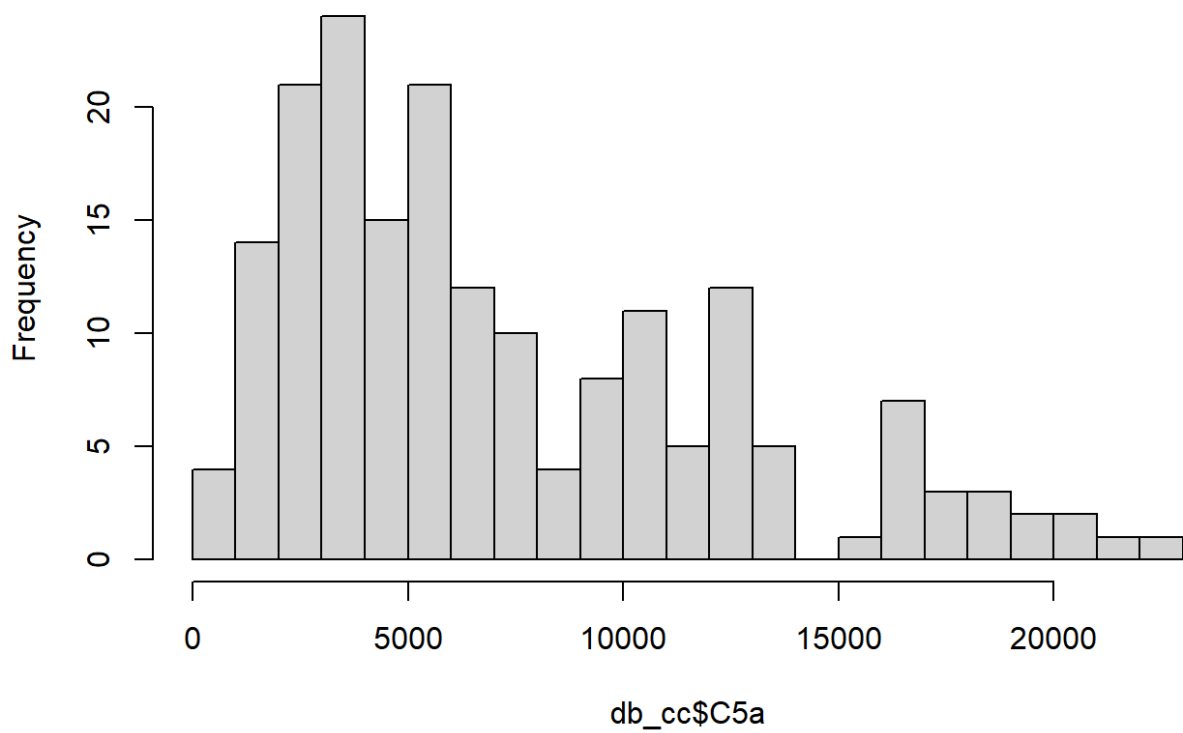
nrow(db); nrow(db_cc)
```

```
## [1] 243
```

```
## [1] 186
```

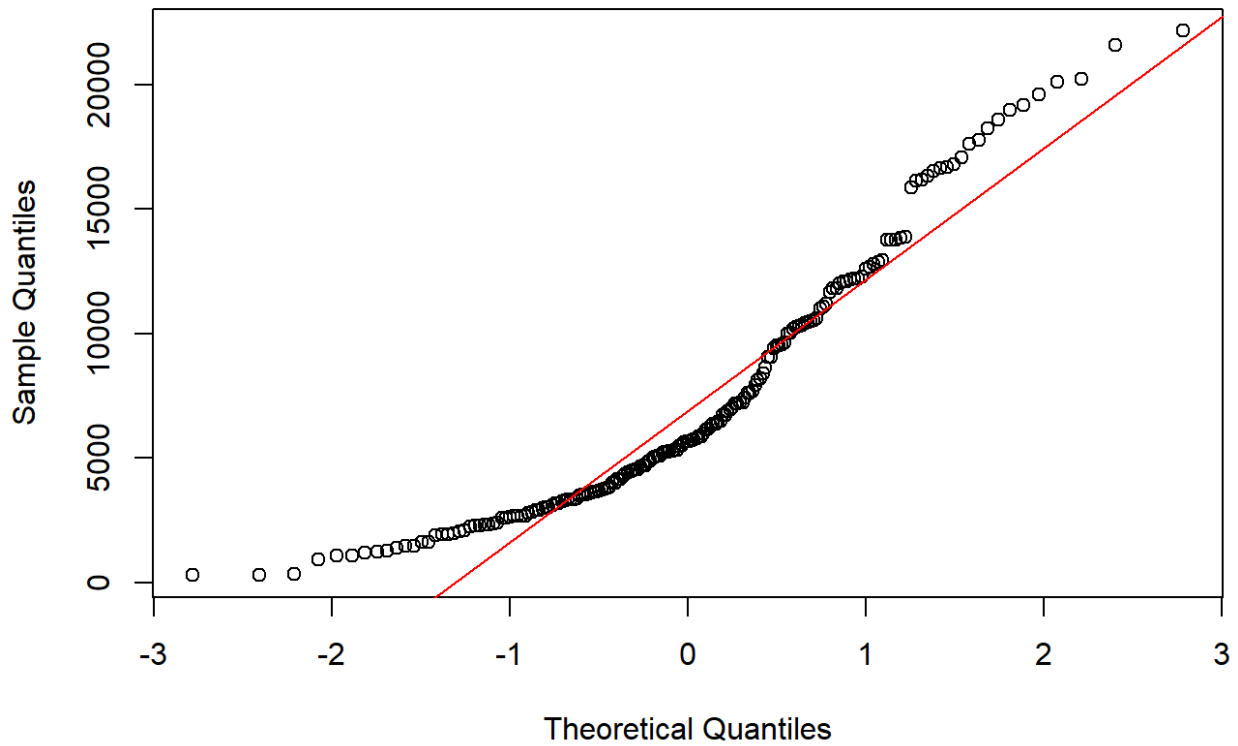
```
hist(db_cc$C5a, breaks=30)
```

Histogram of db_cc\$C5a



```
qqnorm(db_cc$C5a); qqline(db_cc$C5a, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$C5a)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##   306.4  3339.7   5678.0   7286.2 10449.5 22143.9
```

```
min(db_cc$C5a, na.rm=TRUE)
```

```
## [1] 306.354
```

```
db$C5a_z <- scale(log(db$C5a))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$C5a_z <- scale(log(db_cc$C5a))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

C5aModel_hetero <- glmmTMB(
  C5a_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(C5aModel_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## C5a_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion:                ~t
## Data: db_cc
##
##           AIC           BIC      logLik -2*log(L)  df.resid
##        457.9        519.2      -210.0      419.9       167
##
## Random effects:
##
## Conditional model:
##   Groups   Name              Variance Std.Dev.
##   id       (Intercept) 0.7076   0.8412
##   Residual                NA        NA
## Number of obs: 186, groups: id, 80
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.906581   3.200715   0.596   0.5514
## t1             -0.021067   0.212766  -0.099   0.9211
## t2             -0.062262   0.409797  -0.152   0.8792
## GROUP1         -0.582603   0.343428  -1.696   0.0898 .
## GROUP2         -0.072341   0.306343  -0.236   0.8133
## GROUP3          0.053974   0.409367   0.132   0.8951
## Age            -0.025268   0.041500  -0.609   0.5426
## SexM           -0.001548   0.213479  -0.007   0.9942
## CIRS_severity_0 0.203709   0.144530   1.410   0.1587
## t1:GROUP1       0.264145   0.258747   1.021   0.3073
## t2:GROUP1       0.130332   0.337594   0.386   0.6995
## t1:GROUP2      -0.236213   0.218197  -1.083   0.2790
## t2:GROUP2       0.072070   0.290377   0.248   0.8040
## t1:GROUP3      -0.350521   0.285878  -1.226   0.2202
## t2:GROUP3       0.158545   0.322566   0.492   0.6231
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.76733    0.15540  -4.938 7.9e-07 ***
## t1           -0.06338    0.25131  -0.252   0.801
## t2            0.16416    0.22150   0.741   0.459
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

C5aModel_homo <- glmmTMB(
  C5a_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(C5aModel_homo, C5aModel_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## C5aModel_homo: C5a_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## C5aModel_homo:      (1 | id), zi=~0, disp=~t
## C5aModel_hetero: C5a_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## C5aModel_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## C5aModel_homo   17 454.64 509.48 -210.32   420.64
## C5aModel_hetero  19 457.90 519.19 -209.95   419.90 0.7362      2      0.692

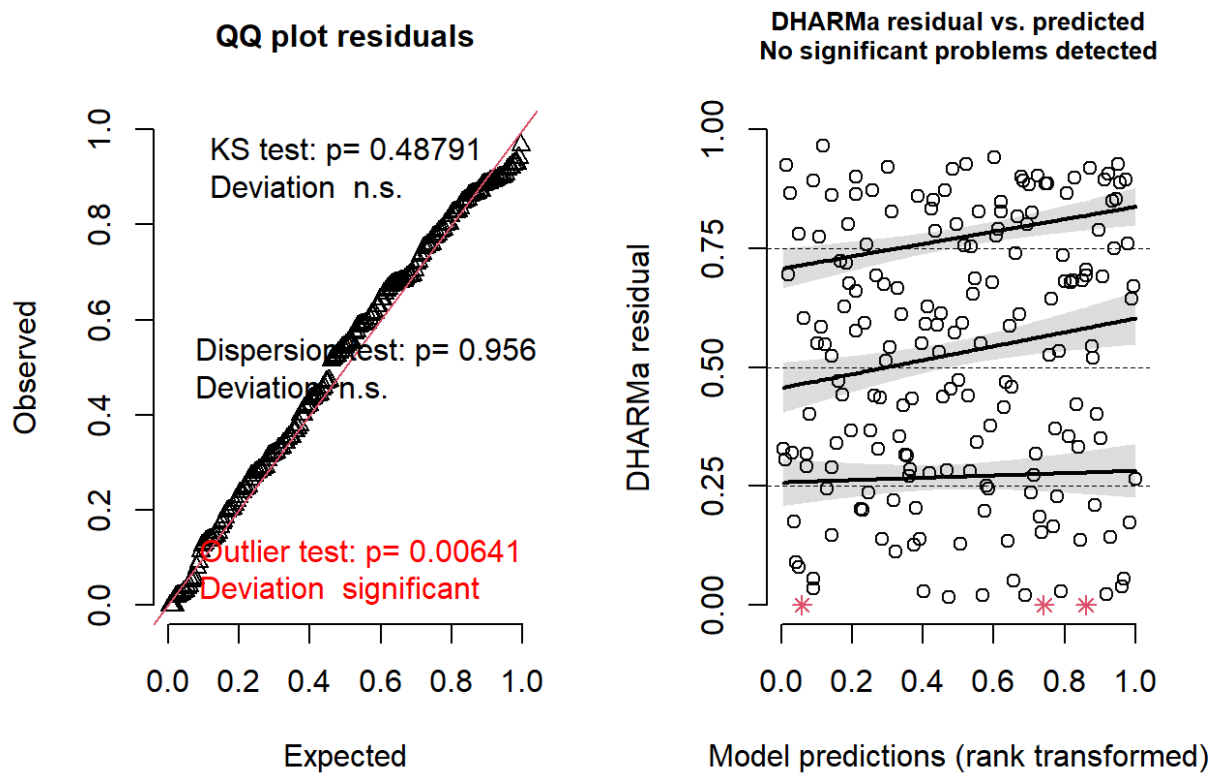
```

```

sim_res <- simulateResiduals(C5aModel_hetero, n = 1000)
plot(sim_res)

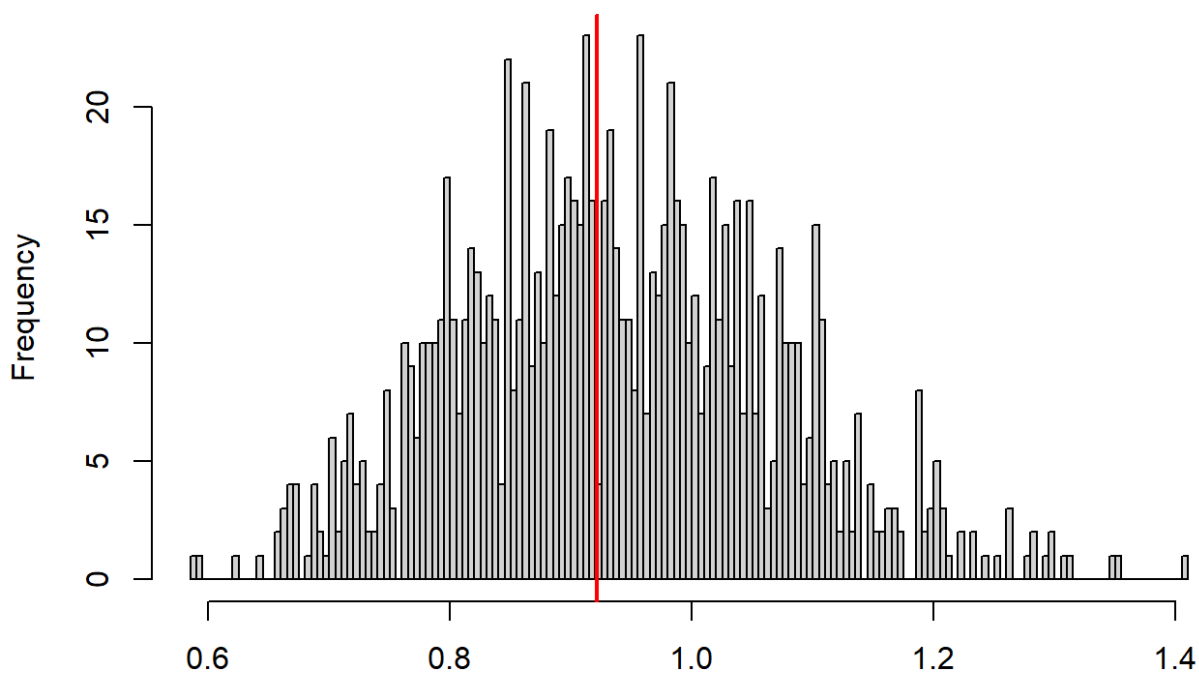
```

DHARMA residual



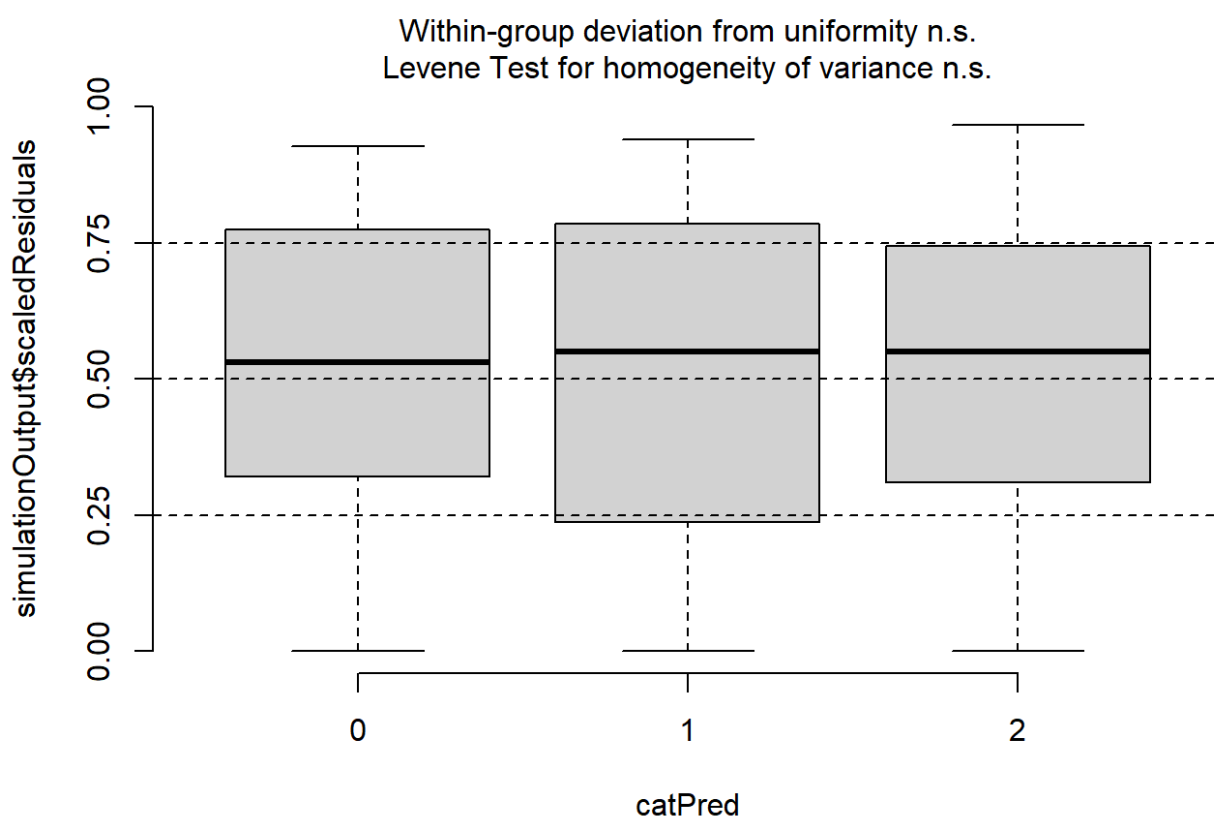
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated




```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 0.98352, p-value = 0.956
## alternative hypothesis: two.sided
```

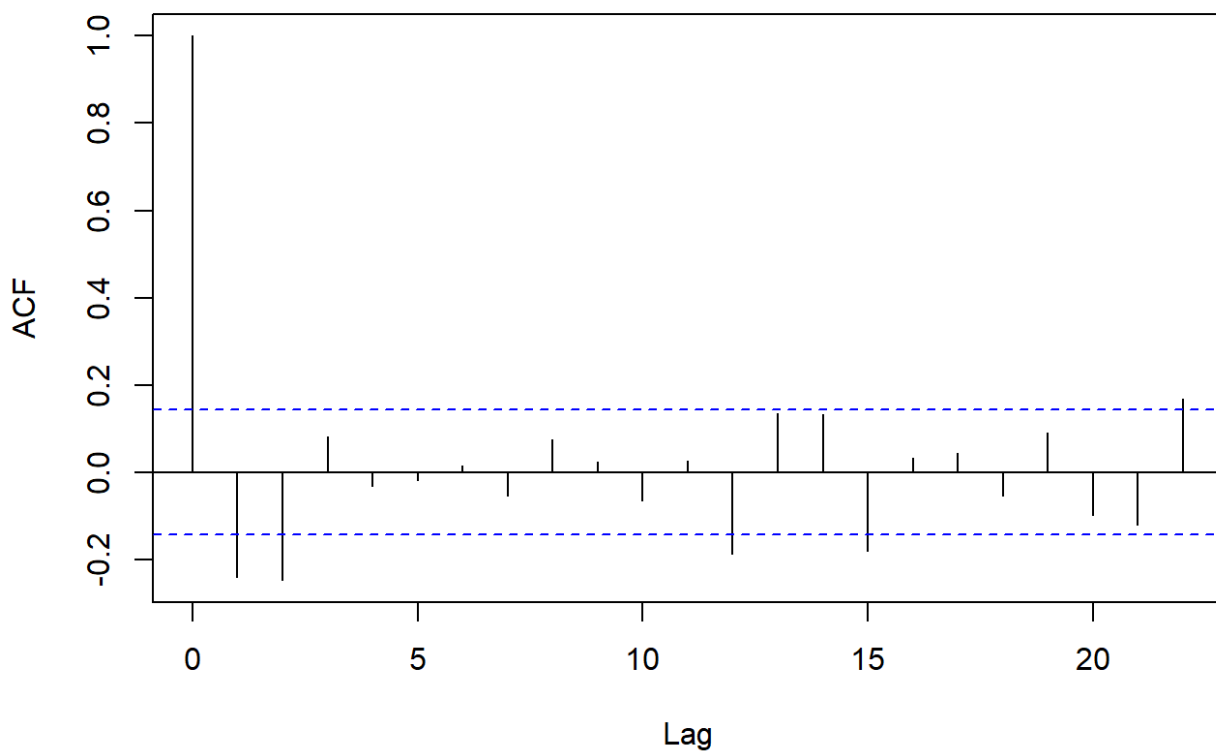
```
testQuantiles(sim_res, db_cc$t)
```



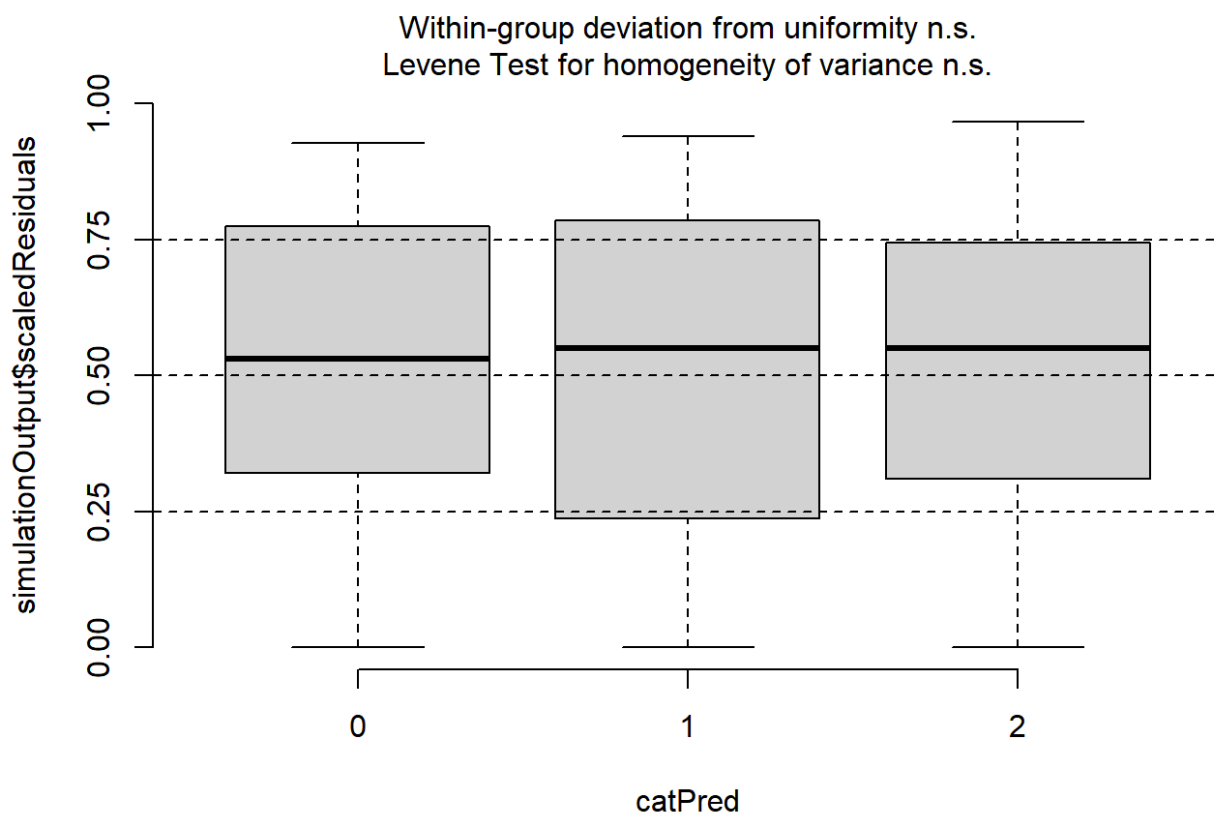
```
## NULL
```

```
acf(residuals(C5aModel_hetero))
```

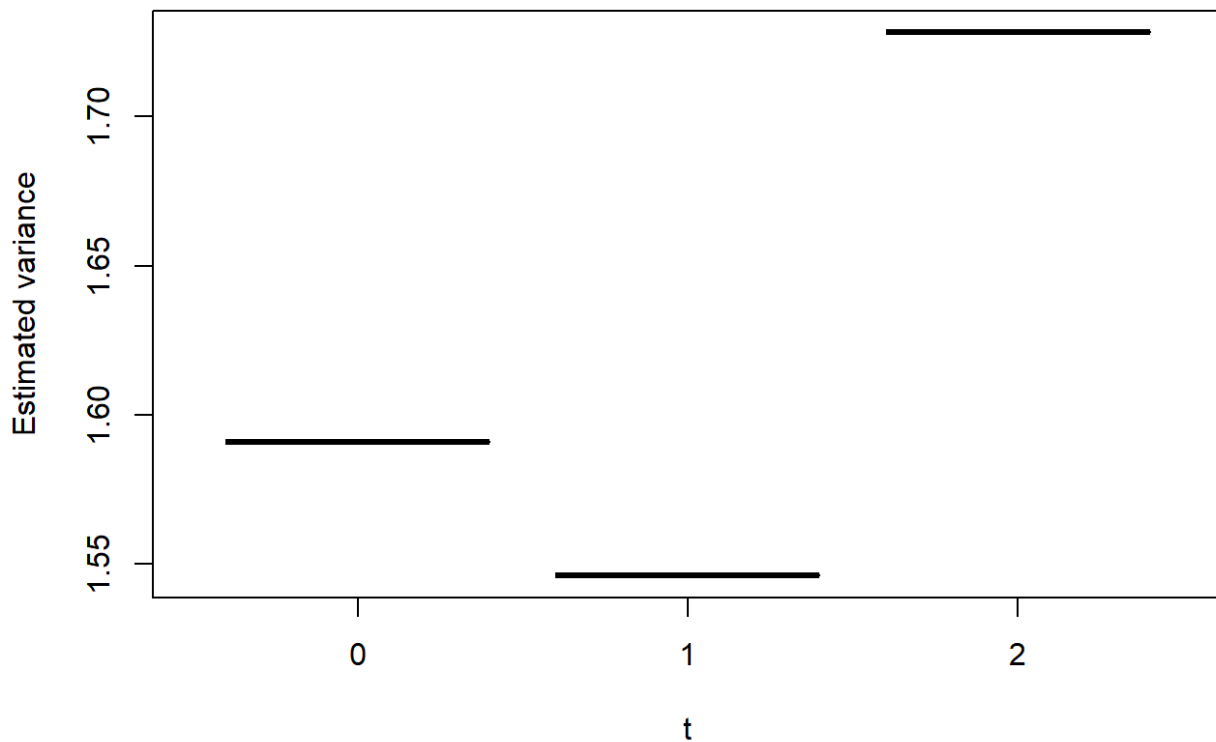
Series residuals(C5aModel_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

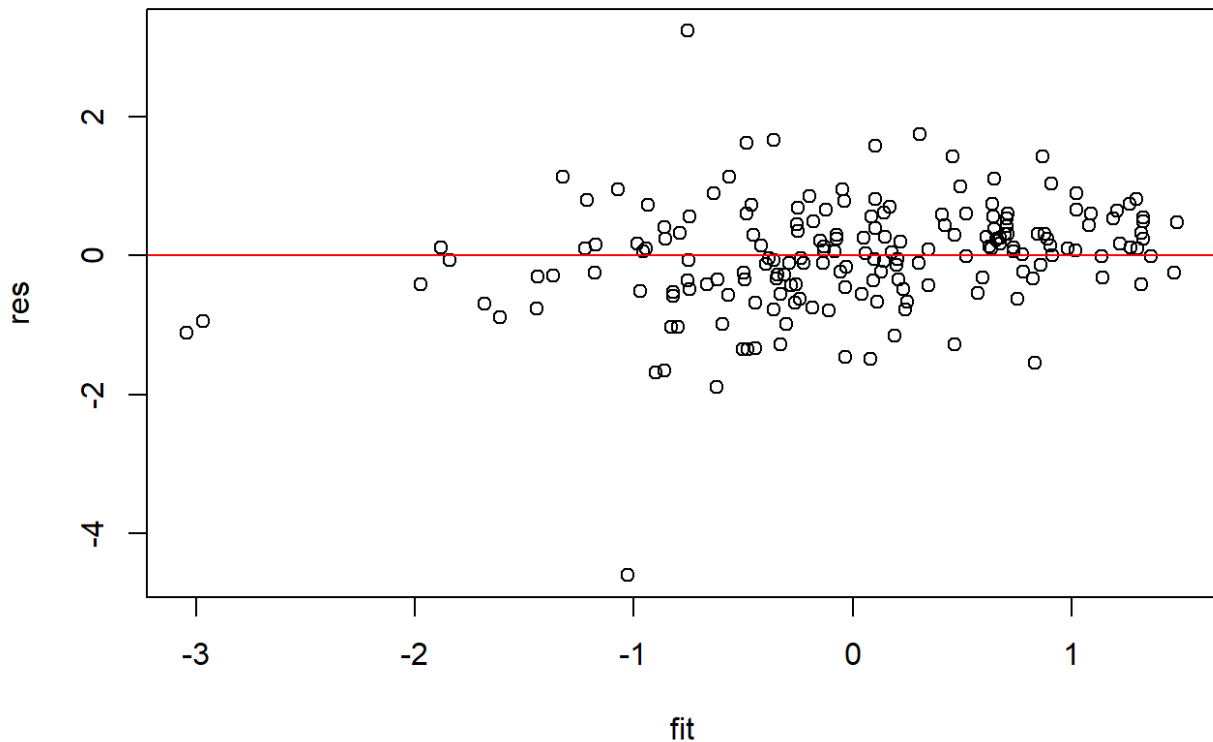


```
pred_disp <- predict(C5aModel_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(C5aModel_hetero, type="pearson")
fit <- fitted(C5aModel_hetero)

plot(fit, res)
abline(h=0, col="red")
```

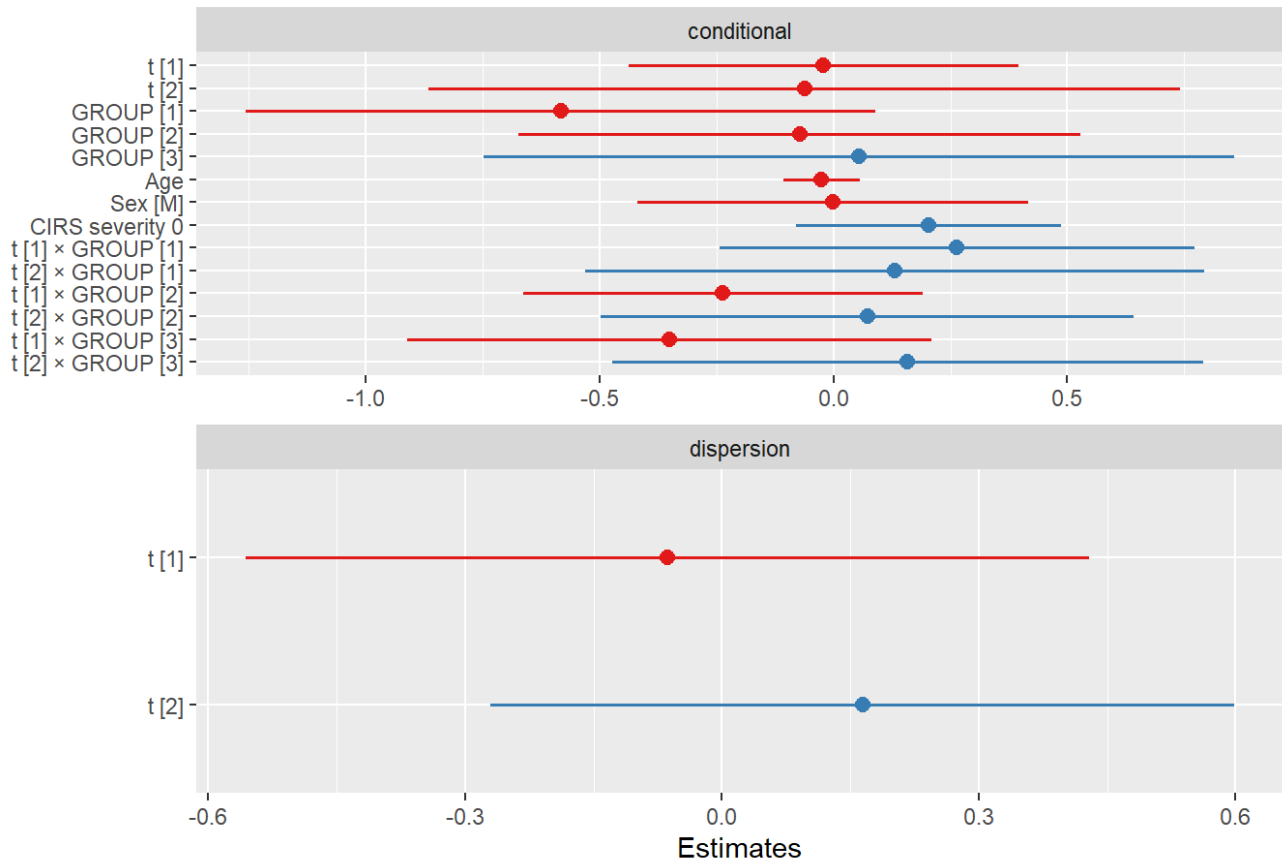


3.1 Visualizzazione del modello

```
plot_model(C5aModel_hetero)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## i Please use `linewidth` instead.  
## i The deprecated feature was likely used in the sjPlot package.  
## Please report the issue at <https://github.com/strengejacked/sjPlot/issues>.  
## This warning is displayed once per session.  
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was  
## generated.
```

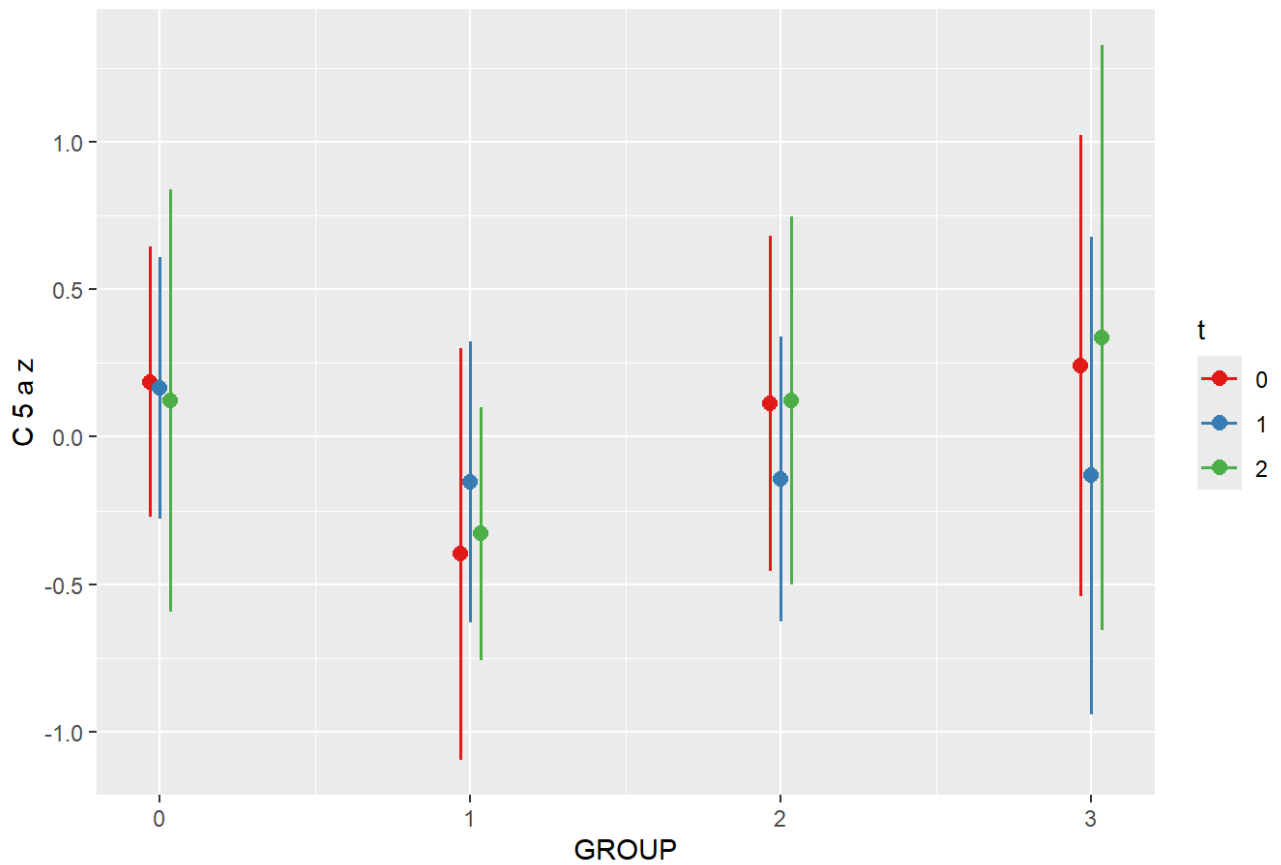
C 5 a z



```
plot_model(C5aModel_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

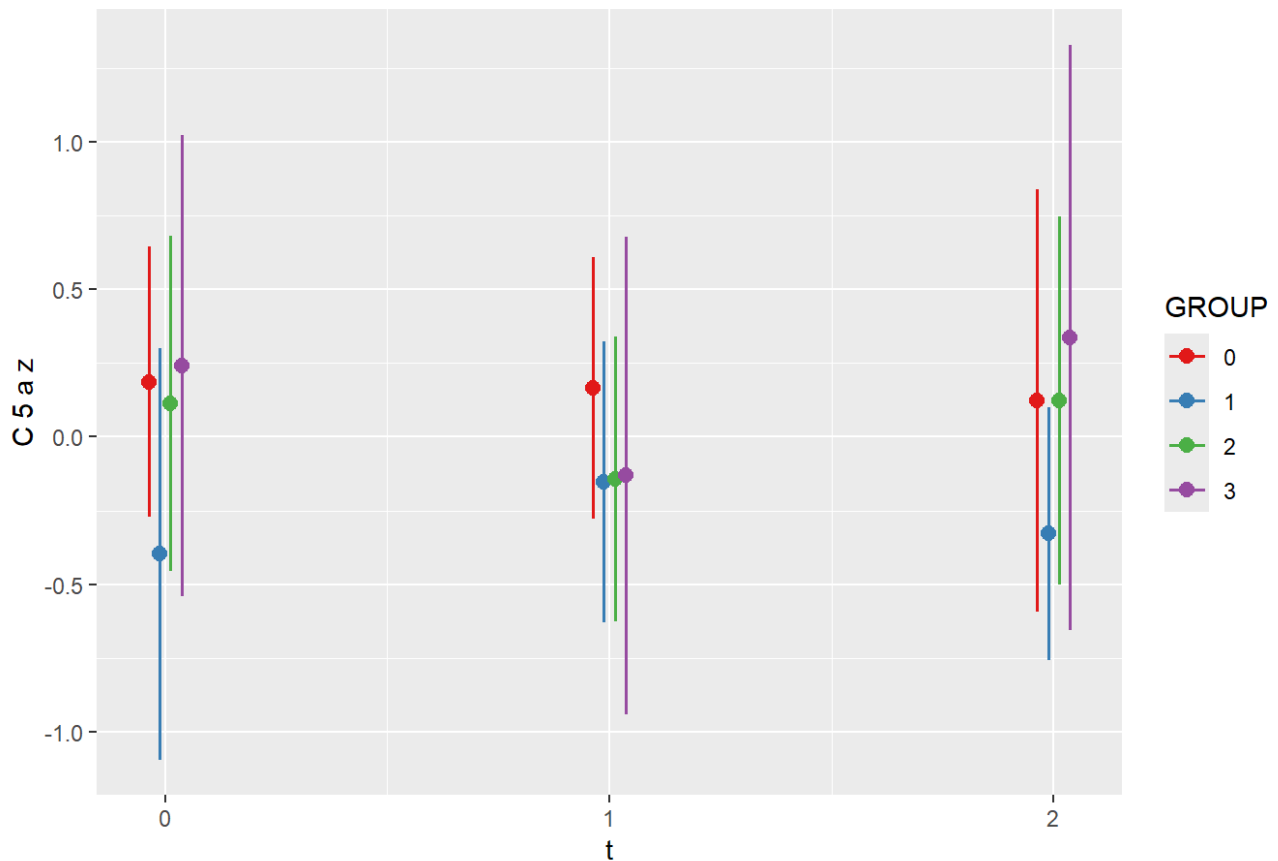
Predicted values of C 5 a z



```
plot_model(C5aModel_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of C 5 a z

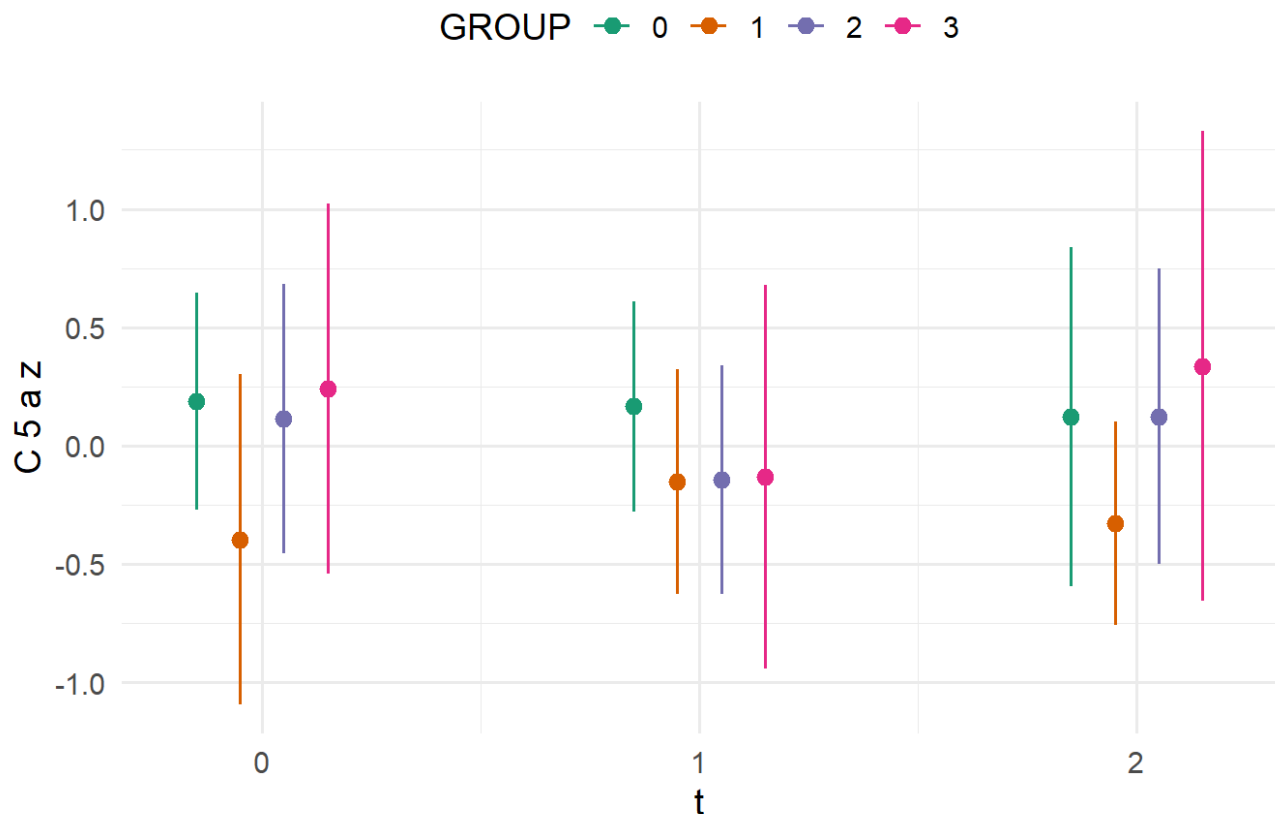


```
plot_model(C5aModel_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

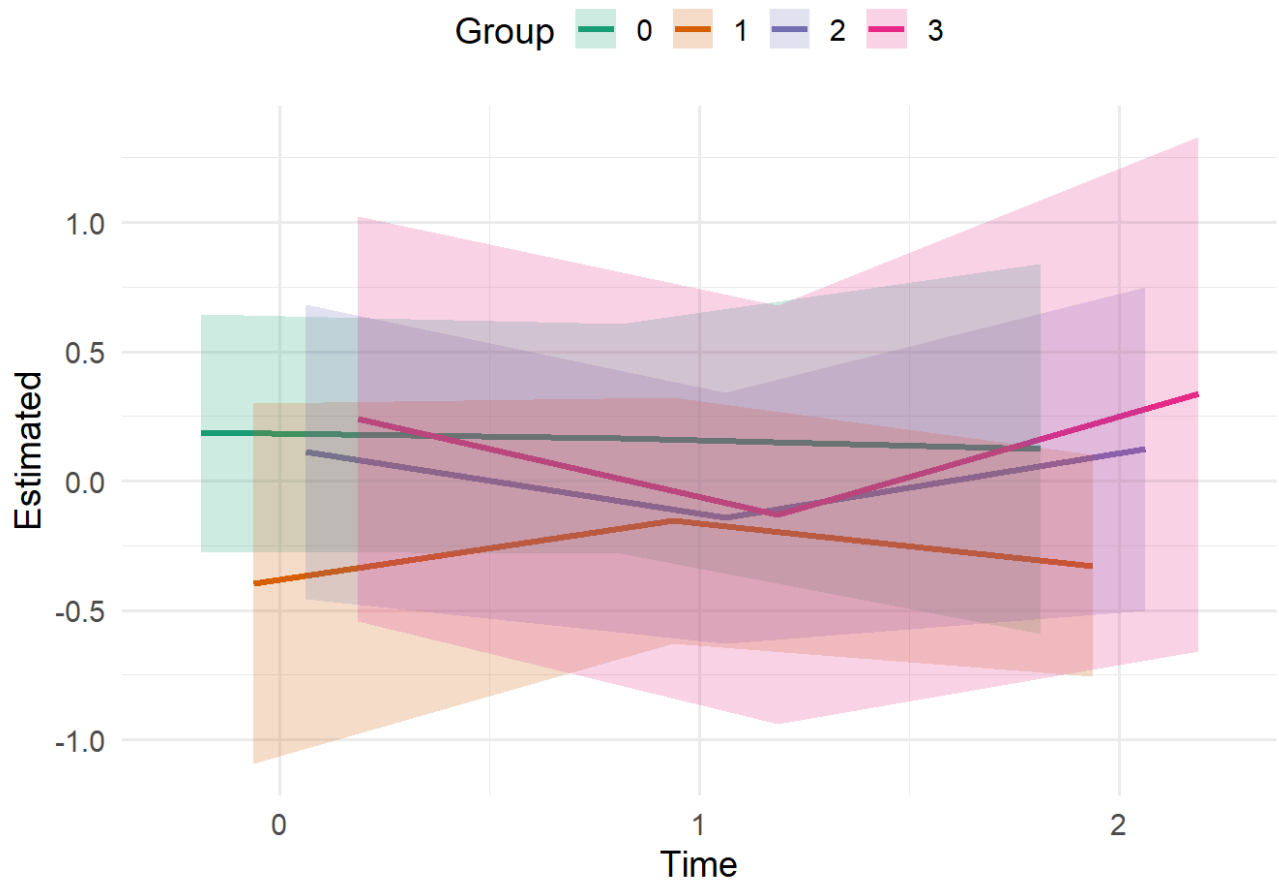
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of C 5 a z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(C5aModel_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```

3.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(C5aModel_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP emmean      SE  df asymp.LCL asymp.UCL
##  0      0.186 0.211 Inf    -0.228    0.5992
##  1     -0.397 0.360 Inf    -1.102    0.3081
##  2      0.113 0.290 Inf    -0.454    0.6808
##  3      0.240 0.374 Inf    -0.493    0.9718
##
## t = 1:
##  GROUP emmean      SE  df asymp.LCL asymp.UCL
##  0      0.165 0.195 Inf    -0.217    0.5462
##  1     -0.154 0.241 Inf    -0.626    0.3179
##  2     -0.144 0.240 Inf    -0.615    0.3269
##  3     -0.132 0.385 Inf    -0.886    0.6222
##
## t = 2:
##  GROUP emmean      SE  df asymp.LCL asymp.UCL
##  0      0.123 0.343 Inf    -0.549    0.7958
##  1     -0.329 0.210 Inf    -0.740    0.0826
##  2      0.123 0.307 Inf    -0.479    0.7248
##  3      0.336 0.481 Inf    -0.607    1.2788
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(C5aModel_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  contrast          estimate      SE  df z.ratio p.value
##  GROUP0 - GROUP1      0.451 0.314 Inf    1.436  0.4765
##  GROUP0 - GROUP2      0.127 0.284 Inf    0.448  0.9701
##  GROUP0 - GROUP3      0.010 0.380 Inf    0.026  1.0000
##  GROUP1 - GROUP2     -0.324 0.307 Inf   -1.055  0.7170
##  GROUP1 - GROUP3     -0.441 0.449 Inf   -0.982  0.7596
##  GROUP2 - GROUP3     -0.117 0.418 Inf   -0.280  0.9924
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(C5aModel_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.582604 0.343 Inf   1.696  0.3255
## GROUP0 - GROUP2  0.072341 0.306 Inf   0.236  0.9954
## GROUP0 - GROUP3 -0.053974 0.409 Inf  -0.132  0.9992
## GROUP1 - GROUP2 -0.510263 0.357 Inf  -1.431  0.4799
## GROUP1 - GROUP3 -0.636578 0.485 Inf  -1.313  0.5545
## GROUP2 - GROUP3 -0.126315 0.451 Inf  -0.280  0.9923
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.318458 0.335 Inf   0.952  0.7767
## GROUP0 - GROUP2  0.308553 0.302 Inf   1.022  0.7366
## GROUP0 - GROUP3  0.296547 0.413 Inf   0.718  0.8901
## GROUP1 - GROUP2 -0.009905 0.336 Inf  -0.029  1.0000
## GROUP1 - GROUP3 -0.021911 0.485 Inf  -0.045  1.0000
## GROUP2 - GROUP3 -0.012006 0.452 Inf  -0.027  1.0000
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.452271 0.394 Inf   1.147  0.6604
## GROUP0 - GROUP2  0.000271 0.356 Inf   0.001  1.0000
## GROUP0 - GROUP3 -0.212519 0.439 Inf  -0.484  0.9626
## GROUP1 - GROUP2 -0.452000 0.358 Inf  -1.262  0.5875
## GROUP1 - GROUP3 -0.664791 0.520 Inf  -1.277  0.5774
## GROUP2 - GROUP3 -0.212791 0.483 Inf  -0.440  0.9714
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(C5aModel_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    0.02107 0.213 Inf   0.099  0.9946
## t0 - t2    0.06226 0.410 Inf   0.152  0.9874
## t1 - t2    0.04120 0.300 Inf   0.137  0.9896
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1   -0.24308 0.252 Inf  -0.966  0.5984
## t0 - t2   -0.06807 0.367 Inf  -0.185  0.9812
## t1 - t2    0.17501 0.223 Inf   0.786  0.7115
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    0.25728 0.241 Inf   1.069  0.5334
## t0 - t2   -0.00981 0.391 Inf  -0.025  0.9997
## t1 - t2   -0.26709 0.270 Inf  -0.989  0.5838
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    0.37159 0.306 Inf   1.215  0.4441
## t0 - t2   -0.09628 0.455 Inf  -0.211  0.9757
## t1 - t2   -0.46787 0.368 Inf  -1.273  0.4107
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

4 Model ccl3

```
bptest(lm(ccl3 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(ccl3 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 22.305, df = 14, p-value = 0.07256
```

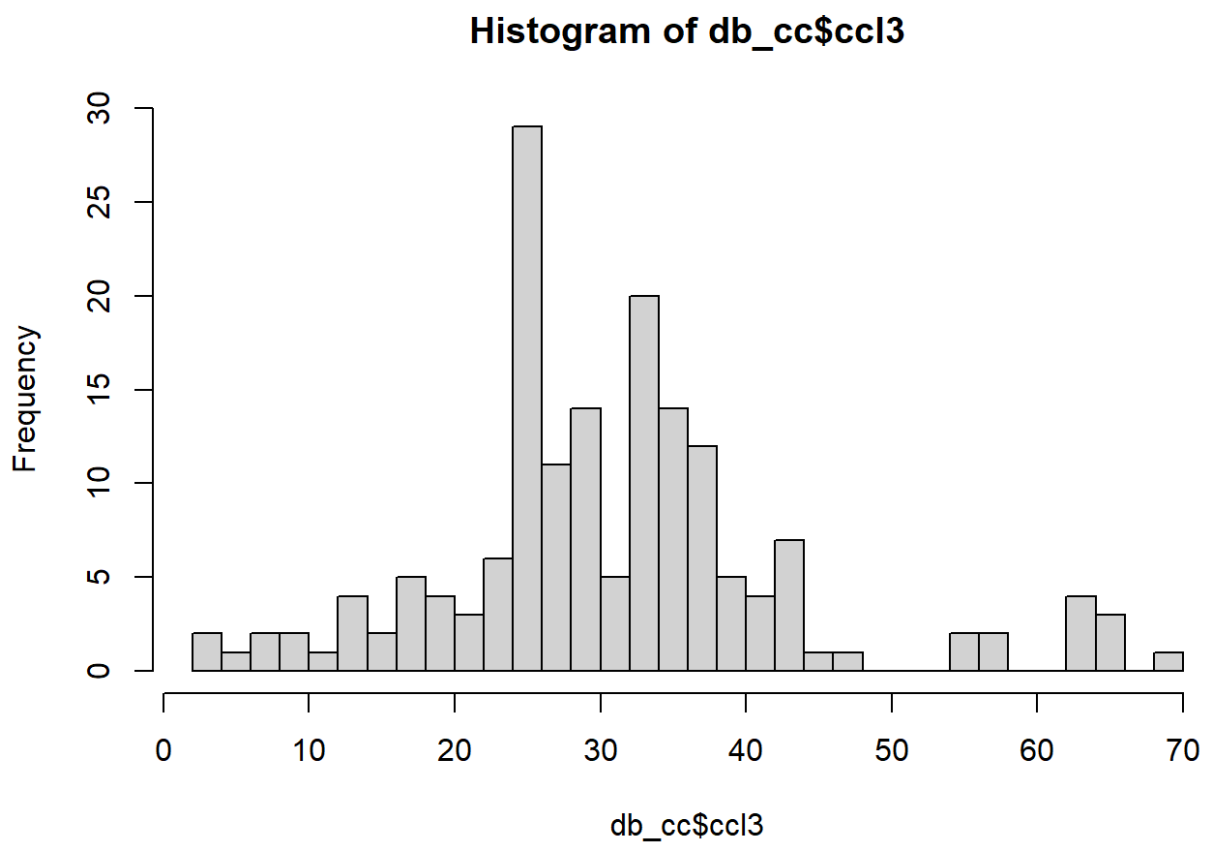
```
vars <- c("ccl3","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

nrow(db); nrow(db_cc)
```

```
## [1] 243
```

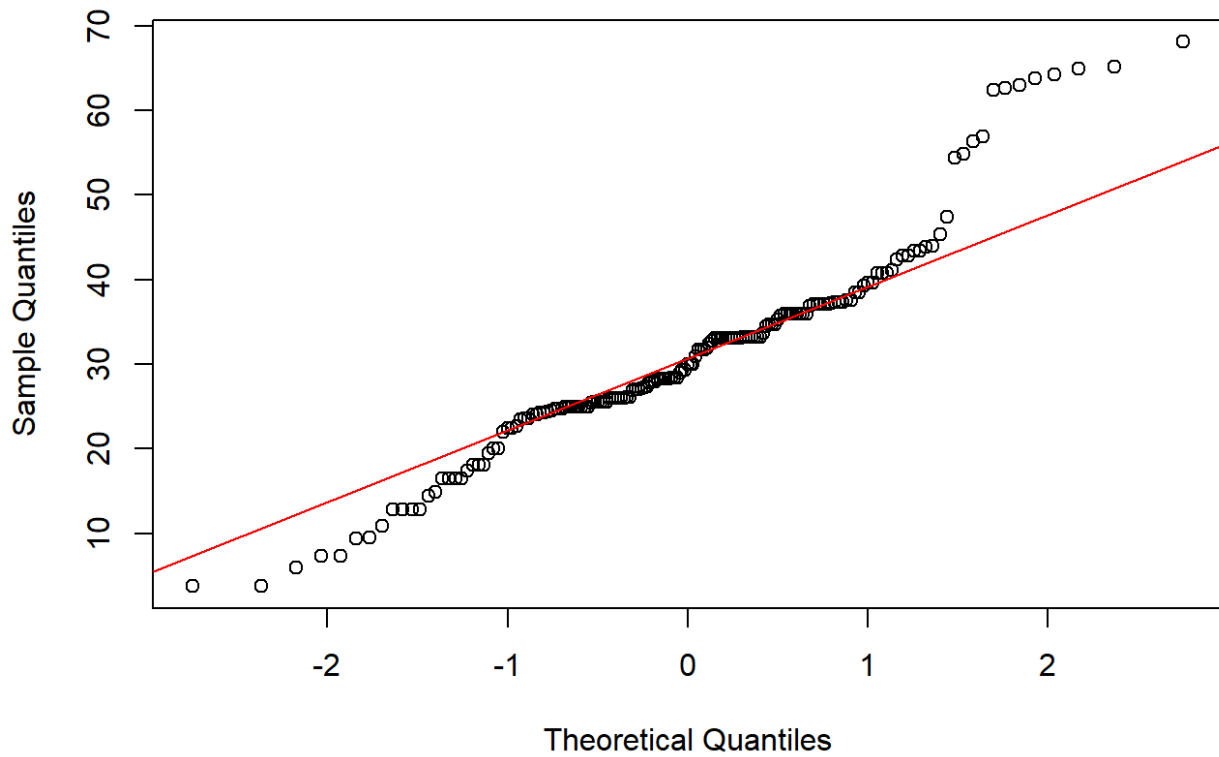
```
## [1] 167
```

```
hist(db_cc$ccl3, breaks=30)
```



```
qqnorm(db_cc$ccl3); qqline(db_cc$ccl3, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$ccl3)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##   3.763  24.989  29.971  31.031  36.426  68.095
```

```
min(db_cc$ccl3, na.rm=TRUE)
```

```
## [1] 3.762943
```

```
db$ccl3_z <- scale(log(db$ccl3))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$ccl3_z <- scale(log(db_cc$ccl3))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

ccl3Model_hetero <- glmmTMB(
  ccl3_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(ccl3Model_hetero)
```

```
## Family: gaussian ( identity )
## Formula:
## ccl3_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion: ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    468.6    527.9   -215.3    430.6      148
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id      (Intercept) 0.06353  0.2521
##   Residual              NA      NA
## Number of obs: 167, groups: id, 78
##
## Conditional model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.49369    2.31489  -0.645   0.5188
## t1            -0.04059    0.30214  -0.134   0.8931
## t2             0.24599    0.45329   0.543   0.5873
## GROUP1        -0.50647    0.43960  -1.152   0.2493
## GROUP2         0.06157    0.40455   0.152   0.8790
## GROUP3        -0.52349    0.57072  -0.917   0.3590
## Age           0.01272    0.02940   0.433   0.6653
## SexM          -0.15283    0.14343  -1.066   0.2866
## CIRS_severity_0 0.24274    0.18946   1.281   0.2001
## t1:GROUP1      1.03336    0.48871   2.114   0.0345 *
## t2:GROUP1      0.61420    0.57554   1.067   0.2859
## t1:GROUP2      0.55591    0.45685   1.217   0.2237
## t2:GROUP2      0.49983    0.55000   0.909   0.3635
## t1:GROUP3      1.13393    0.63650   1.782   0.0748 .
## t2:GROUP3      1.22817    0.69239   1.774   0.0761 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.1202     0.1057   1.138 0.255223
## t1           -0.5557     0.1557  -3.570 0.000357 ***
## t2           -0.2677     0.1499  -1.786 0.074053 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



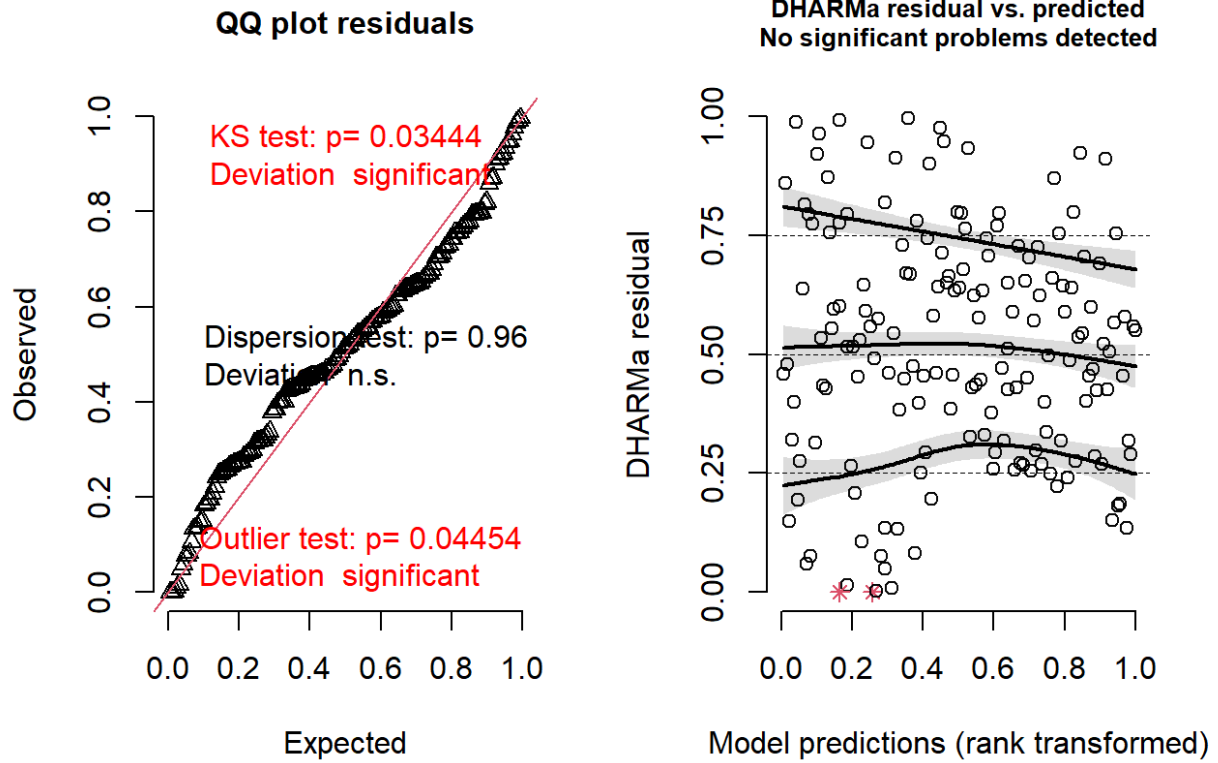
```
ccl3Model_homo <- glmmTMB(
  ccl3_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(ccl3Model_homo, ccl3Model_hetero, test="Chisq")
```

```
## Data: db_cc
## Models:
## ccl3Model_homo: ccl3_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severi
ty_0 + , zi=~0, disp=~1
## ccl3Model_homo:      (1 | id), zi=~0, disp=~t
## ccl3Model_hetero: ccl3_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## ccl3Model_hetero:      (1 | id), zi=~0, disp=~t
##
##          Df      AIC      BIC  logLik deviance Chisq Chi Df Pr(>Chis
q)
## ccl3Model_homo    17 479.32 532.33 -222.66   445.32
## ccl3Model_hetero  19 468.61 527.86 -215.31   430.61 14.71      2 0.00063
93 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

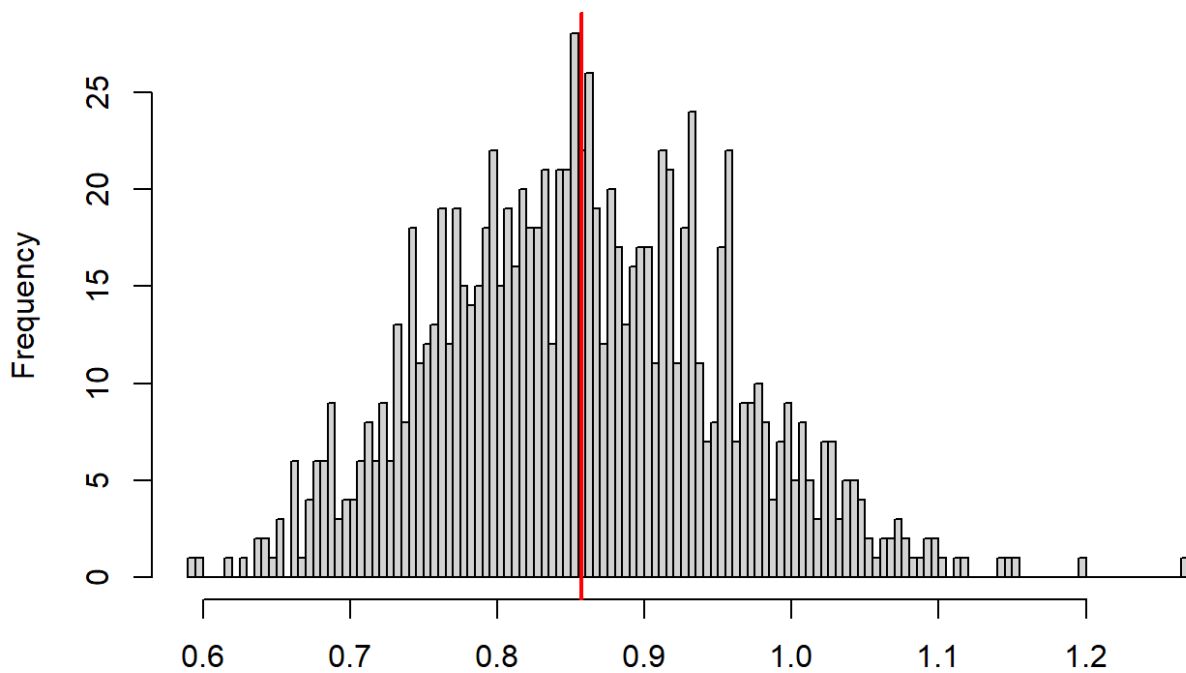
```
sim_res <- simulateResiduals(ccl3Model_hetero, n = 1000)
plot(sim_res)
```

DHARMA residual



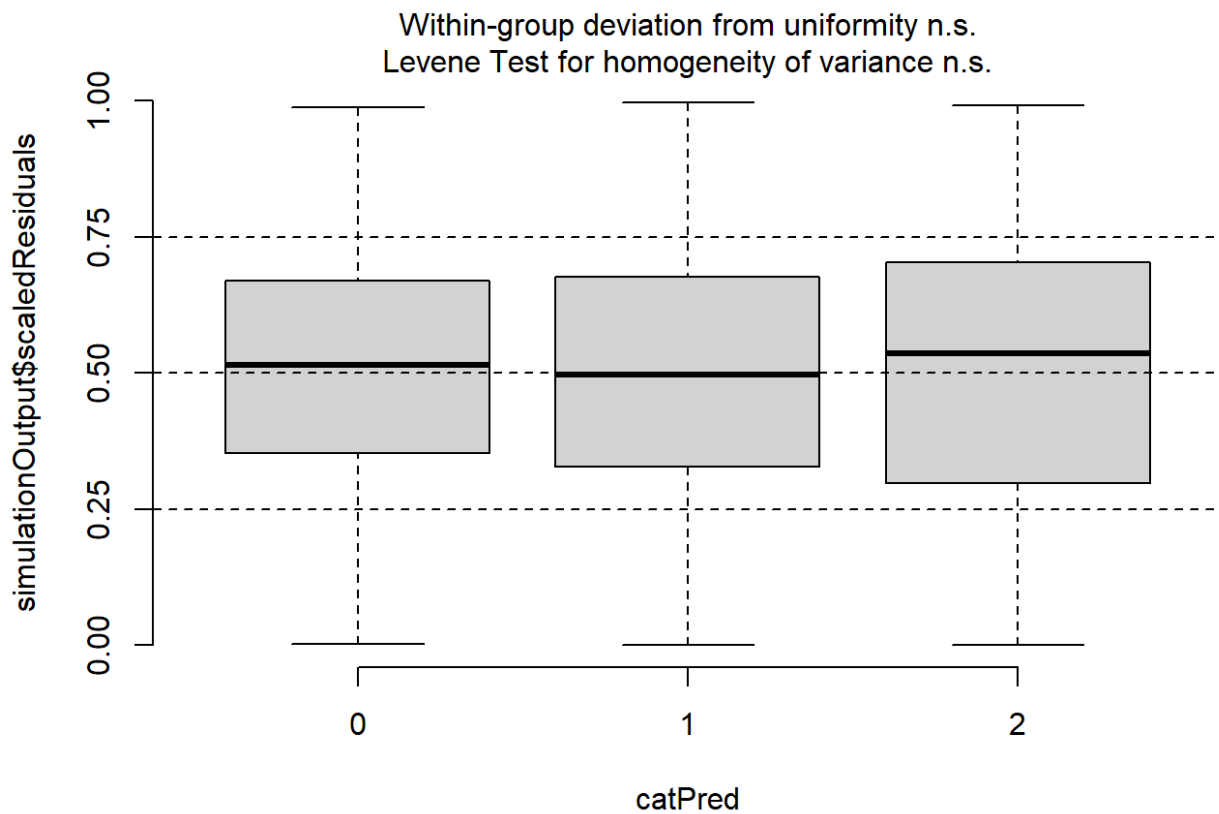
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data:  simulationOutput
## dispersion = 1.0005, p-value = 0.96
## alternative hypothesis: two.sided
```

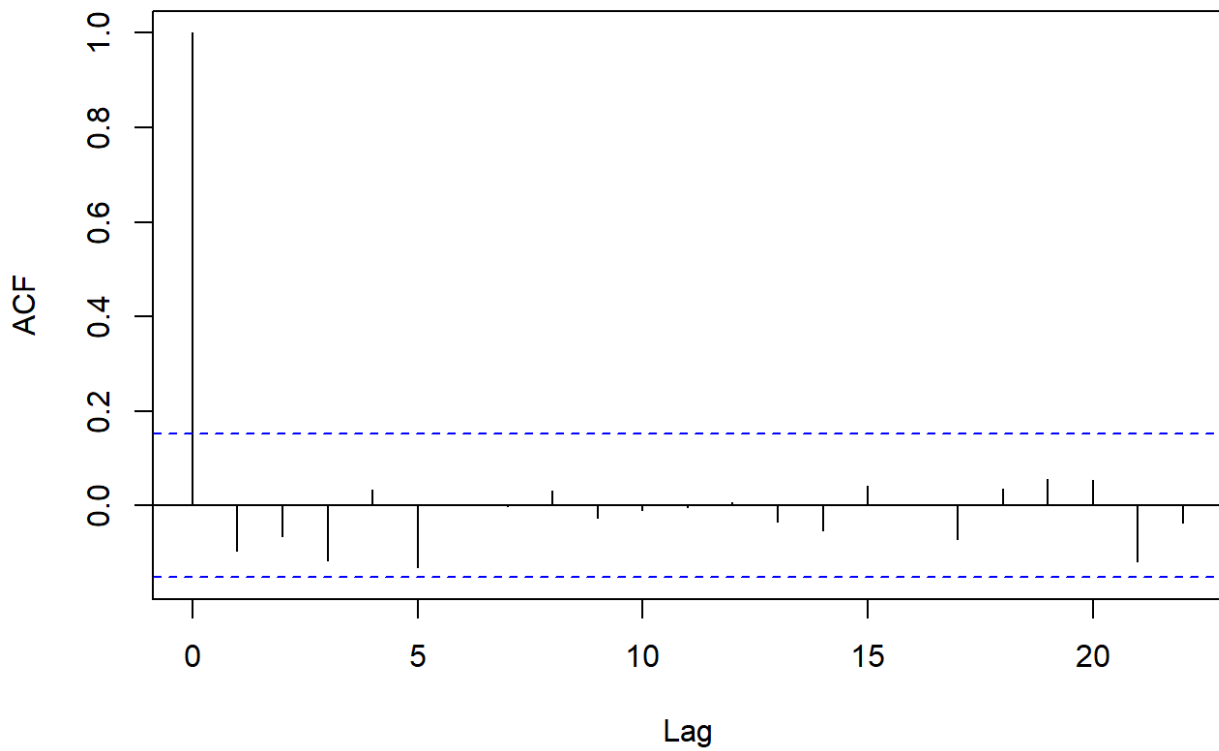
```
testQuantiles(sim_res, db_cc$t)
```



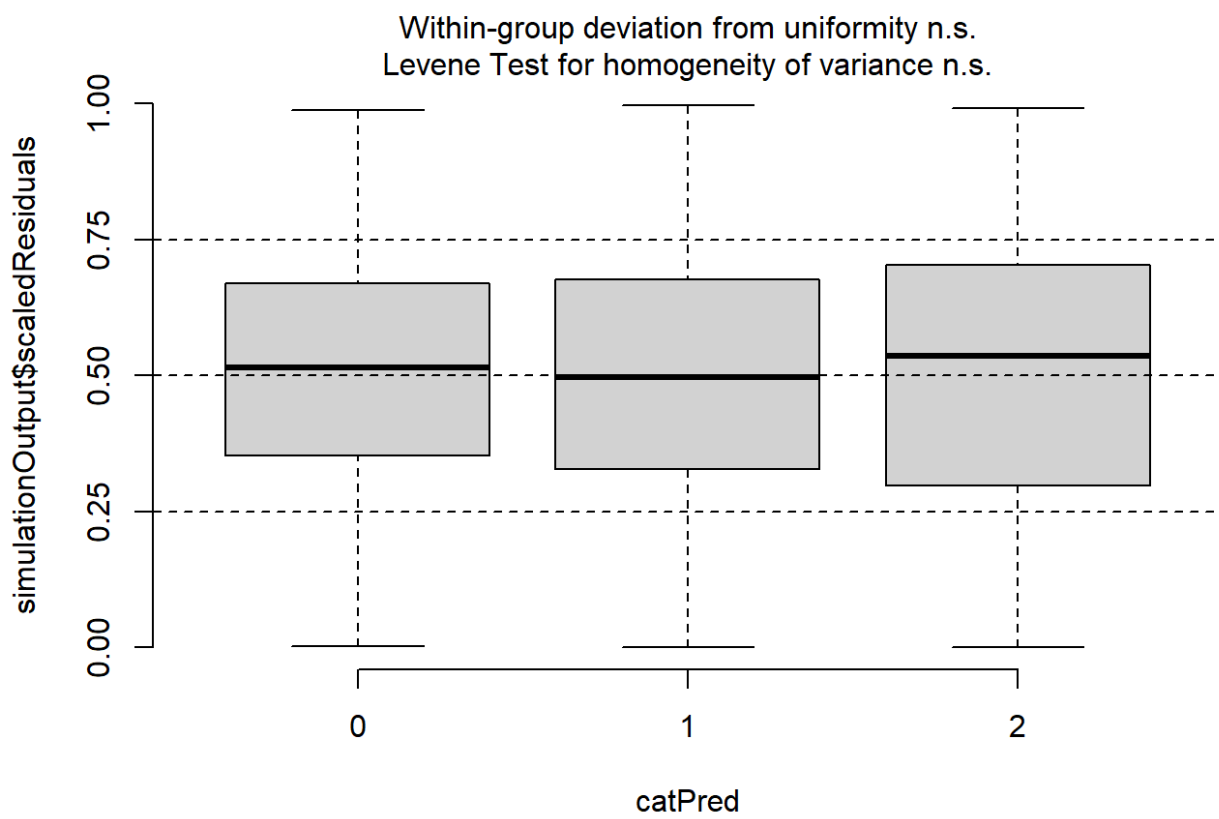
```
## NULL
```

```
acf(residuals(ccl3Model_hetero))
```

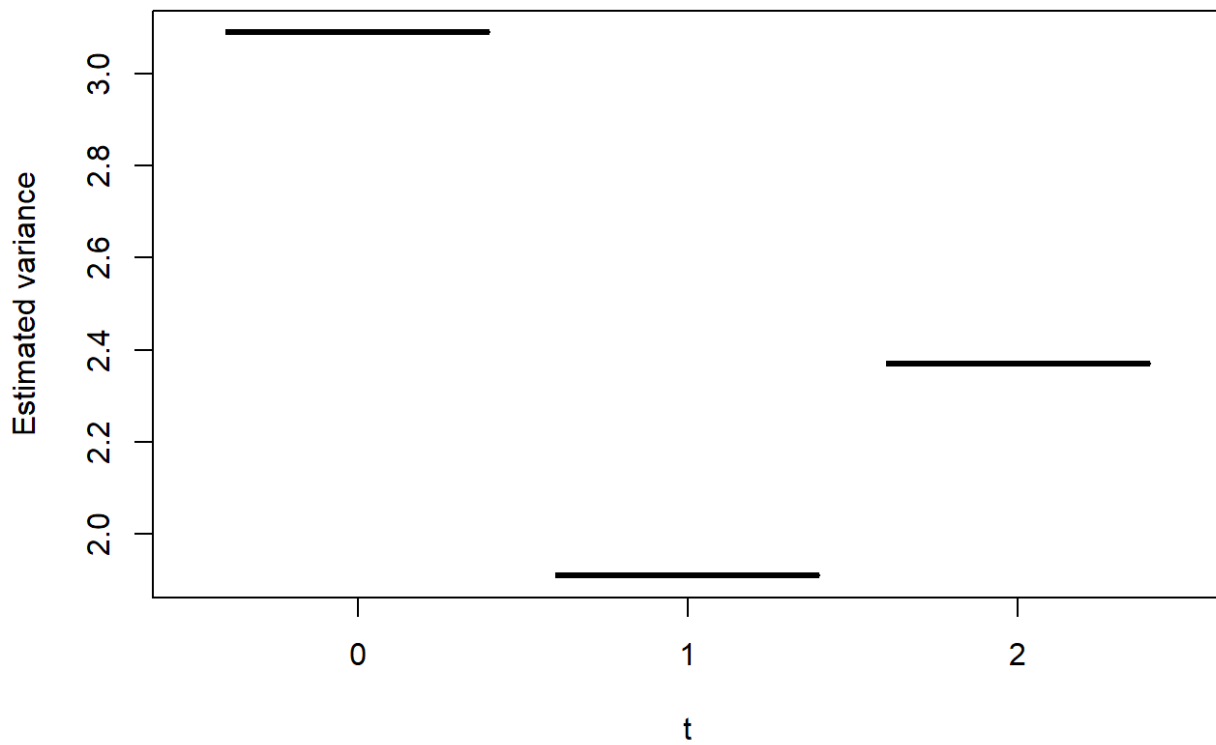
Series residuals(ccI3Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

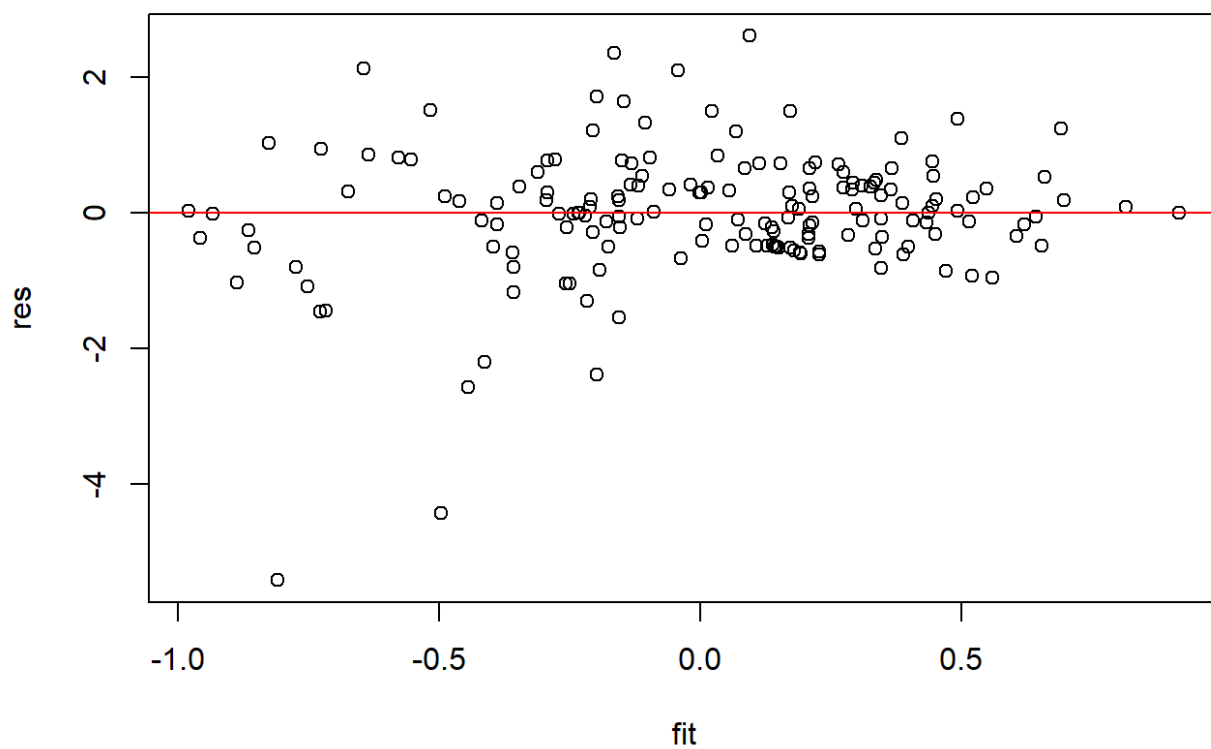


```
pred_disp <- predict(ccl3Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(ccl3Model_hetero, type="pearson")
fit <- fitted(ccl3Model_hetero)

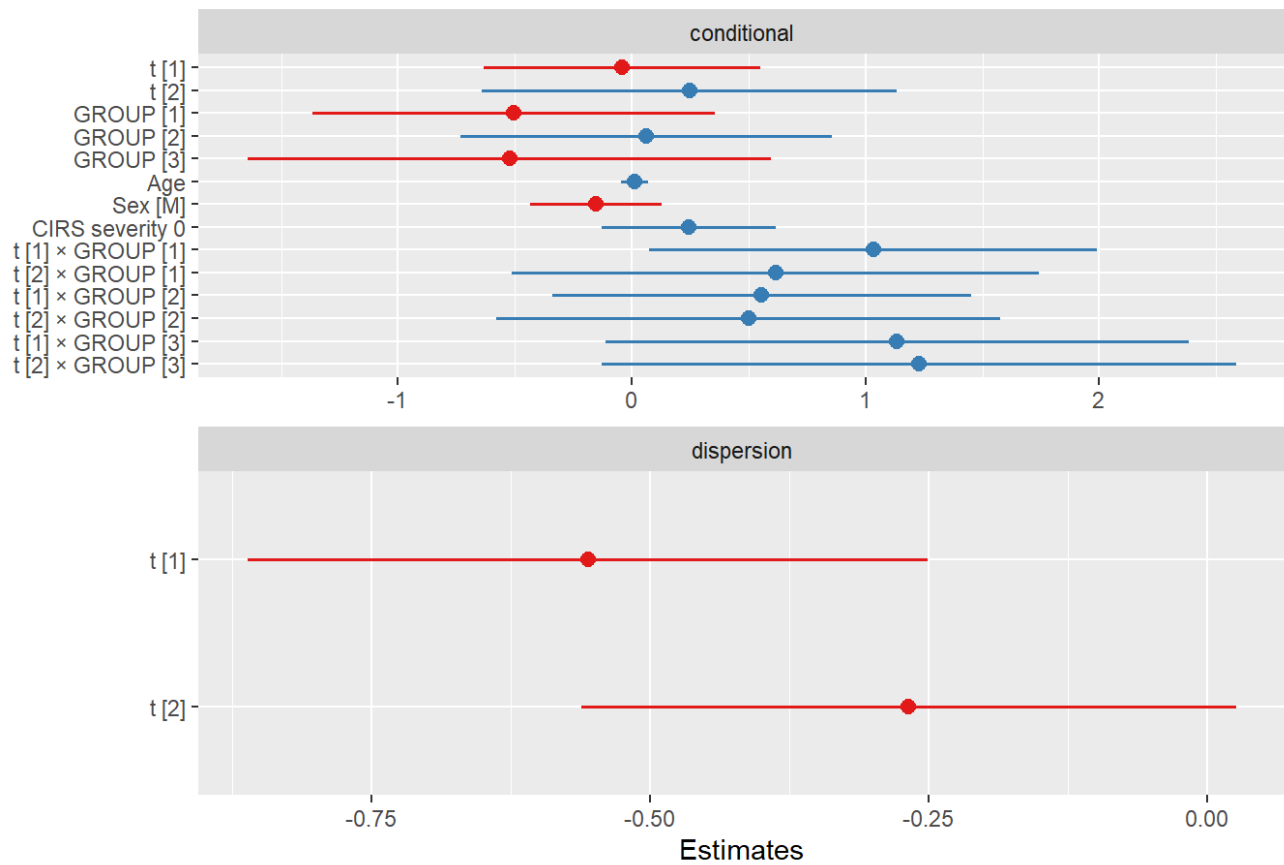
plot(fit, res)
abline(h=0, col="red")
```



4.1 Visualizzazione del modello

```
plot_model(cc13Model_hetero)
```

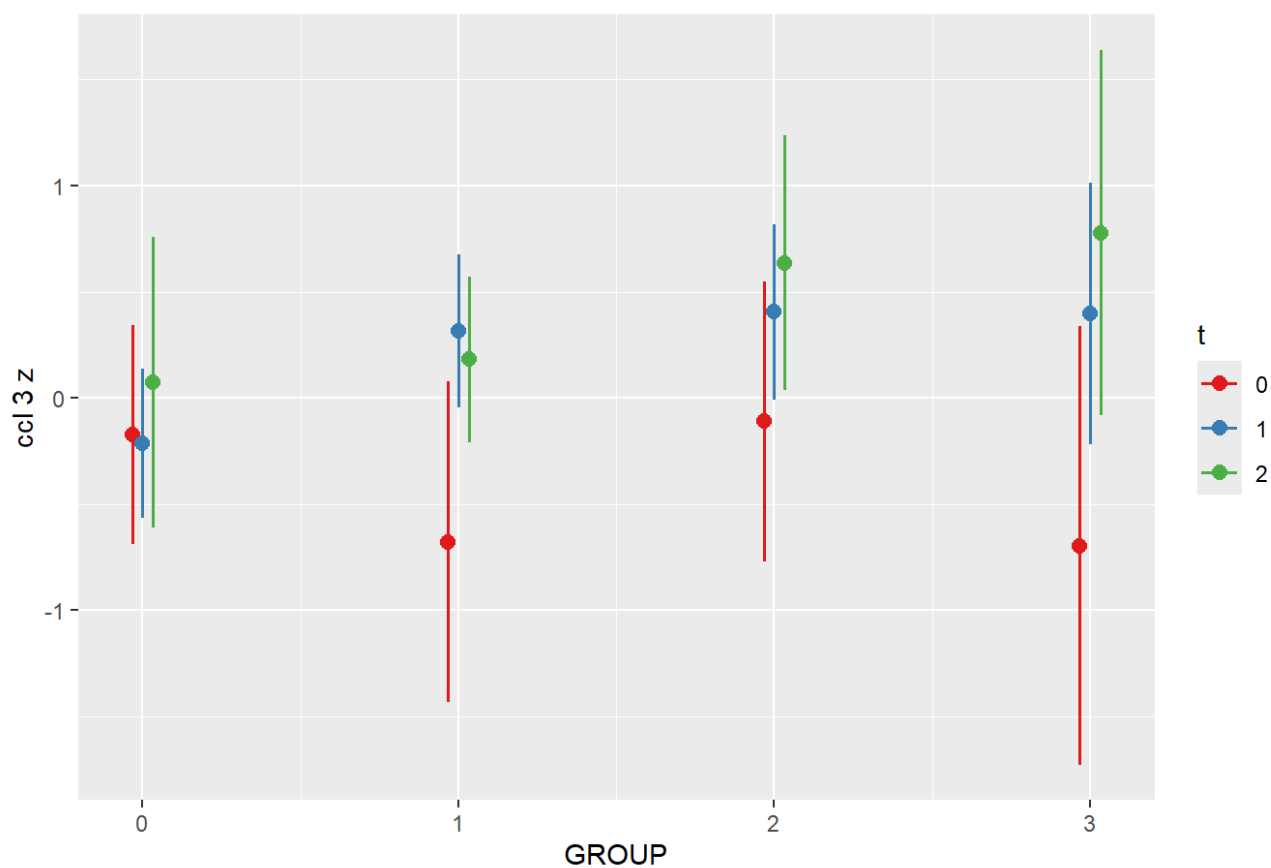
ccl 3 z



```
plot_model(ccl3Model_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

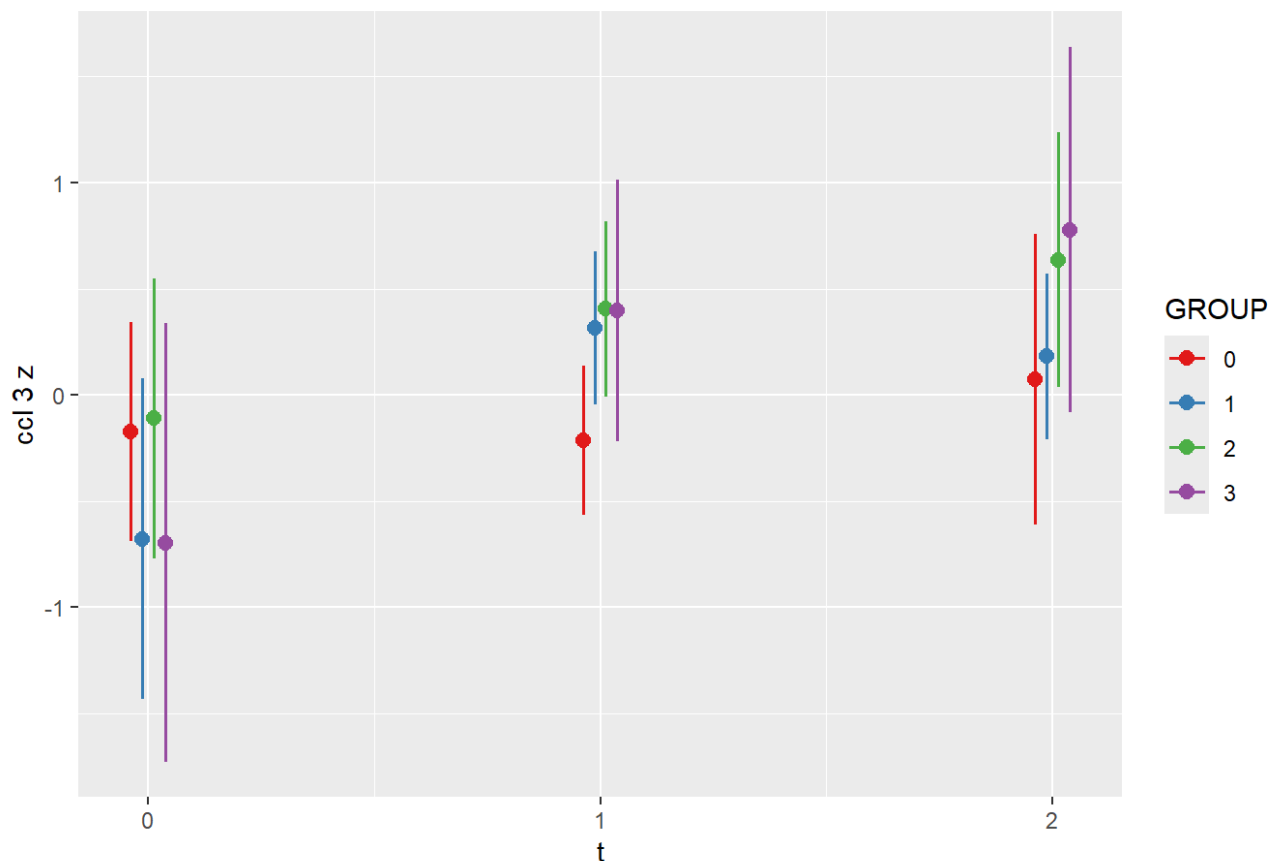
Predicted values of ccl 3 z



```
plot_model(ccl3Model_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```


Predicted values of ccl 3 z

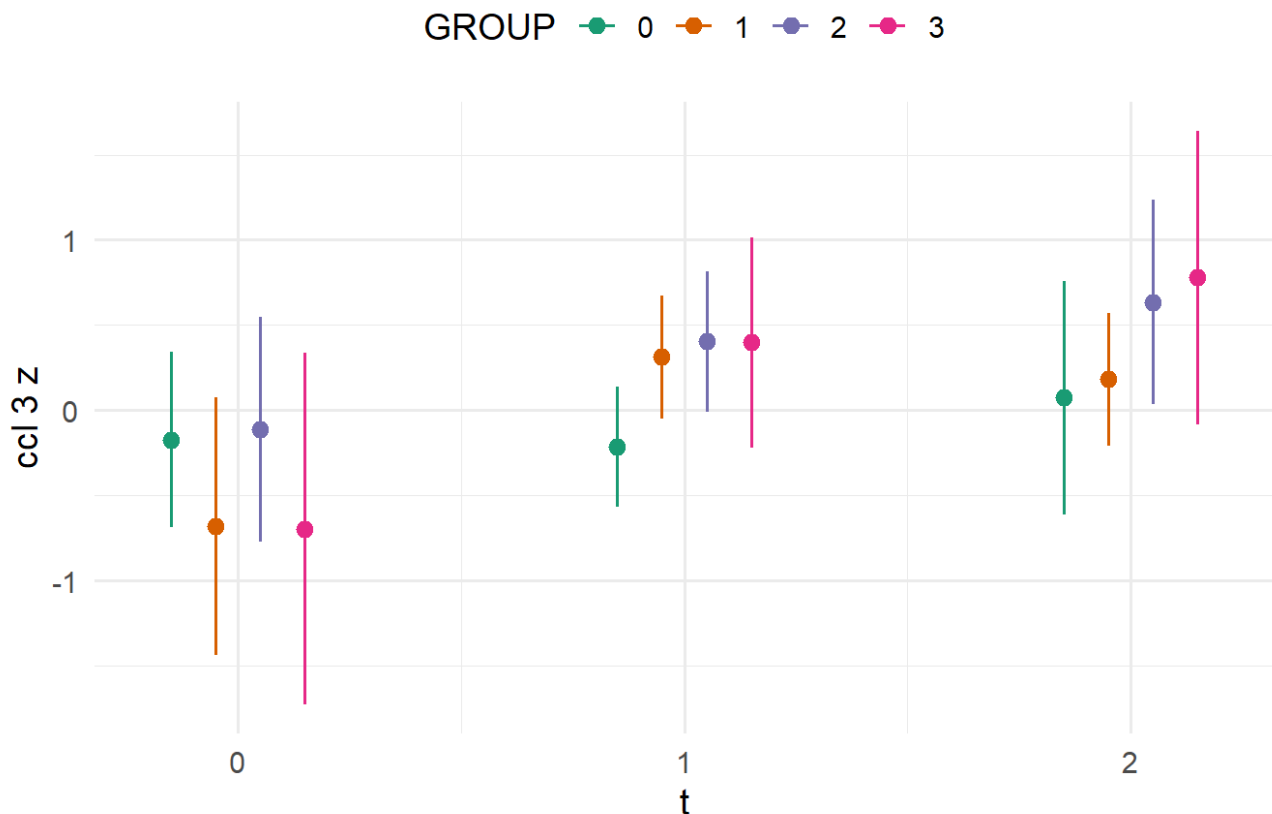


```
plot_model(ccl3Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

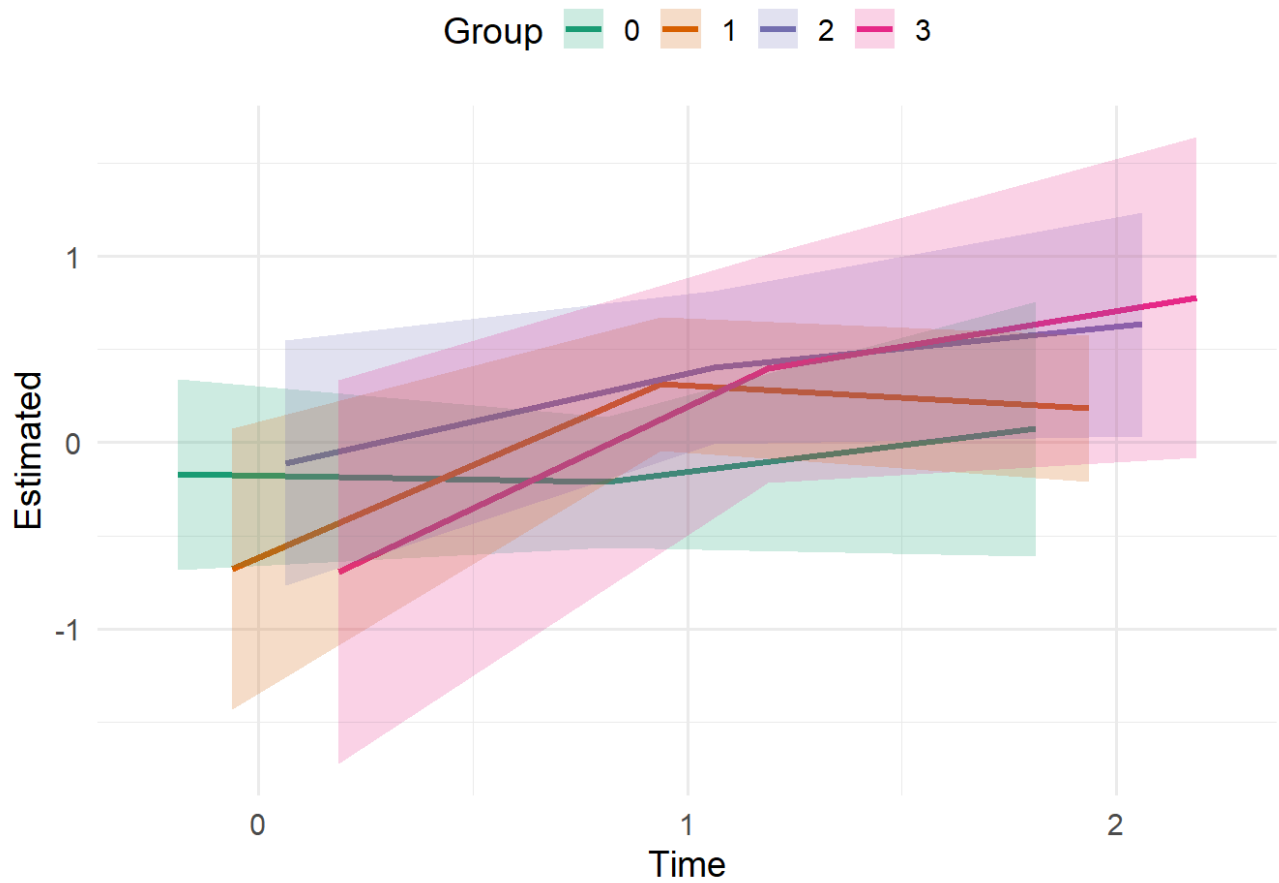
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of ccl 3 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(ccl3Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



4.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(cc13Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0      -0.24853  0.255 Inf   -0.7477  2.51e-01
##   1      -0.75500  0.385 Inf   -1.5101  6.28e-05
##   2      -0.18697  0.338 Inf   -0.8492  4.75e-01
##   3      -0.77202  0.519 Inf   -1.7900  2.46e-01
##
## t = 1:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0      -0.28913  0.163 Inf   -0.6087  3.04e-02
##   1       0.23777  0.181 Inf   -0.1161  5.92e-01
##   2       0.32835  0.206 Inf   -0.0764  7.33e-01
##   3       0.32131  0.298 Inf   -0.2637  9.06e-01
##
## t = 2:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0      -0.00254  0.340 Inf   -0.6693  6.64e-01
##   1       0.10520  0.195 Inf   -0.2776  4.88e-01
##   2       0.55886  0.298 Inf   -0.0247  1.14e+00
##   3       0.70214  0.427 Inf   -0.1345  1.54e+00
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(ccl3Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##   contrast          estimate      SE   df z.ratio p.value
##   GROUP0 - GROUP1  -0.0427  0.234 Inf   -0.182  0.9978
##   GROUP0 - GROUP2  -0.4135  0.217 Inf   -1.908  0.2247
##   GROUP0 - GROUP3  -0.2639  0.276 Inf   -0.955  0.7749
##   GROUP1 - GROUP2  -0.3708  0.236 Inf   -1.574  0.3936
##   GROUP1 - GROUP3  -0.2212  0.324 Inf   -0.683  0.9035
##   GROUP2 - GROUP3   0.1496  0.303 Inf    0.493  0.9606
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(ccl3Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.50647 0.440 Inf   1.152  0.6571
## GROUP0 - GROUP2 -0.06157 0.405 Inf  -0.152  0.9987
## GROUP0 - GROUP3  0.52349 0.571 Inf   0.917  0.7956
## GROUP1 - GROUP2 -0.56803 0.488 Inf  -1.165  0.6493
## GROUP1 - GROUP3  0.01703 0.637 Inf   0.027  1.0000
## GROUP2 - GROUP3  0.58506 0.612 Inf   0.955  0.7748
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.52689 0.257 Inf  -2.054  0.1685
## GROUP0 - GROUP2 -0.61747 0.246 Inf  -2.515  0.0577
## GROUP0 - GROUP3 -0.61044 0.323 Inf  -1.892  0.2314
## GROUP1 - GROUP2 -0.09058 0.277 Inf  -0.327  0.9879
## GROUP1 - GROUP3 -0.08355 0.370 Inf  -0.226  0.9959
## GROUP2 - GROUP3  0.00703 0.351 Inf   0.020  1.0000
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.10774 0.397 Inf  -0.271  0.9930
## GROUP0 - GROUP2 -0.56140 0.391 Inf  -1.436  0.4765
## GROUP0 - GROUP3 -0.70468 0.424 Inf  -1.661  0.3444
## GROUP1 - GROUP2 -0.45366 0.354 Inf  -1.281  0.5751
## GROUP1 - GROUP3 -0.59695 0.476 Inf  -1.254  0.5926
## GROUP2 - GROUP3 -0.14329 0.464 Inf  -0.309  0.9898
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(ccl3Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      0.0406 0.302 Inf   0.134  0.9901
## t0 - t2     -0.2460 0.453 Inf  -0.543  0.8502
## t1 - t2     -0.2866 0.379 Inf  -0.757  0.7293
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1     -0.9928 0.391 Inf  -2.542  0.0297
## t0 - t2     -0.8602 0.428 Inf  -2.009  0.1099
## t1 - t2      0.1326 0.257 Inf   0.516  0.8635
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1     -0.5153 0.386 Inf  -1.335  0.3755
## t0 - t2     -0.7458 0.471 Inf  -1.584  0.2526
## t1 - t2     -0.2305 0.352 Inf  -0.655  0.7894
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1     -1.0933 0.591 Inf  -1.850  0.1534
## t0 - t2     -1.4742 0.673 Inf  -2.189  0.0730
## t1 - t2     -0.3808 0.491 Inf  -0.775  0.7185
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

5 Model ccl4

```
bptest(lm(ccl4 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(ccl4 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 13.747, df = 14, p-value = 0.4687
```

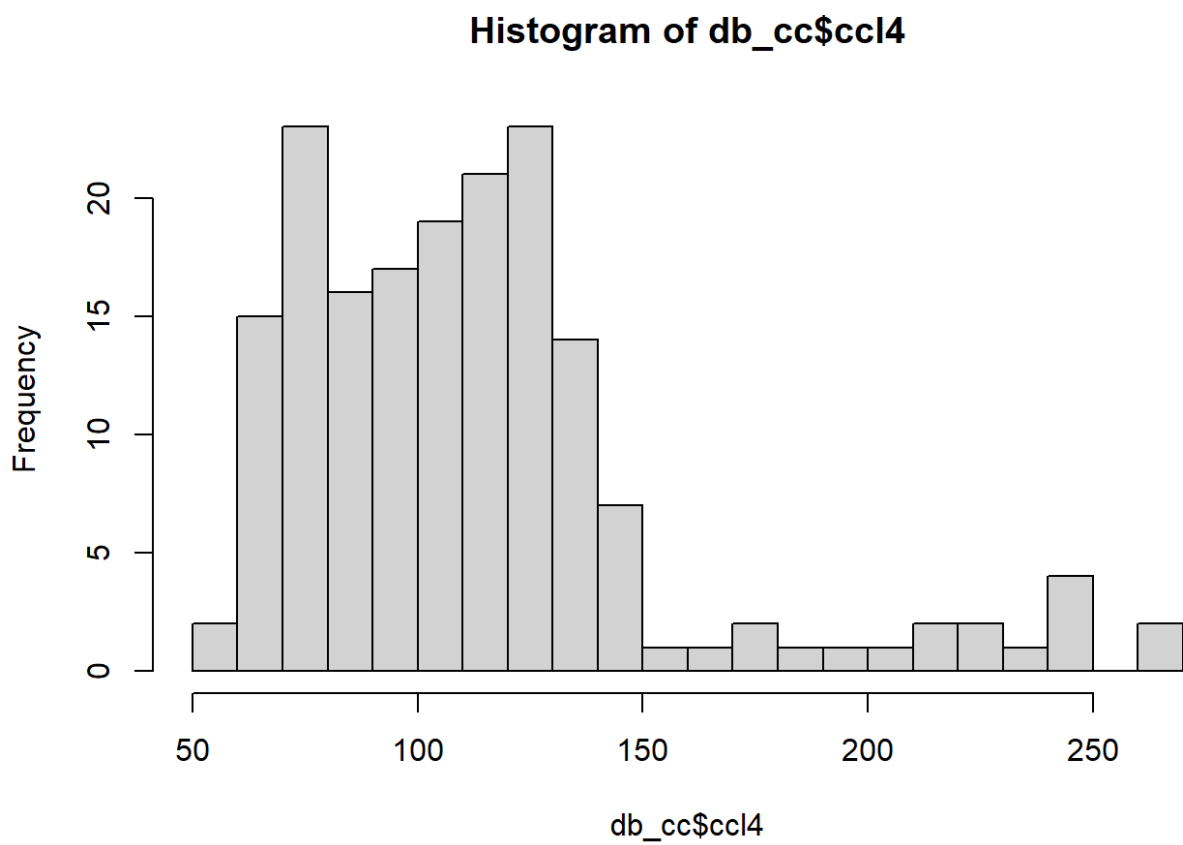
```
vars <- c("ccl4","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

nrow(db); nrow(db_cc)
```

```
## [1] 243
```

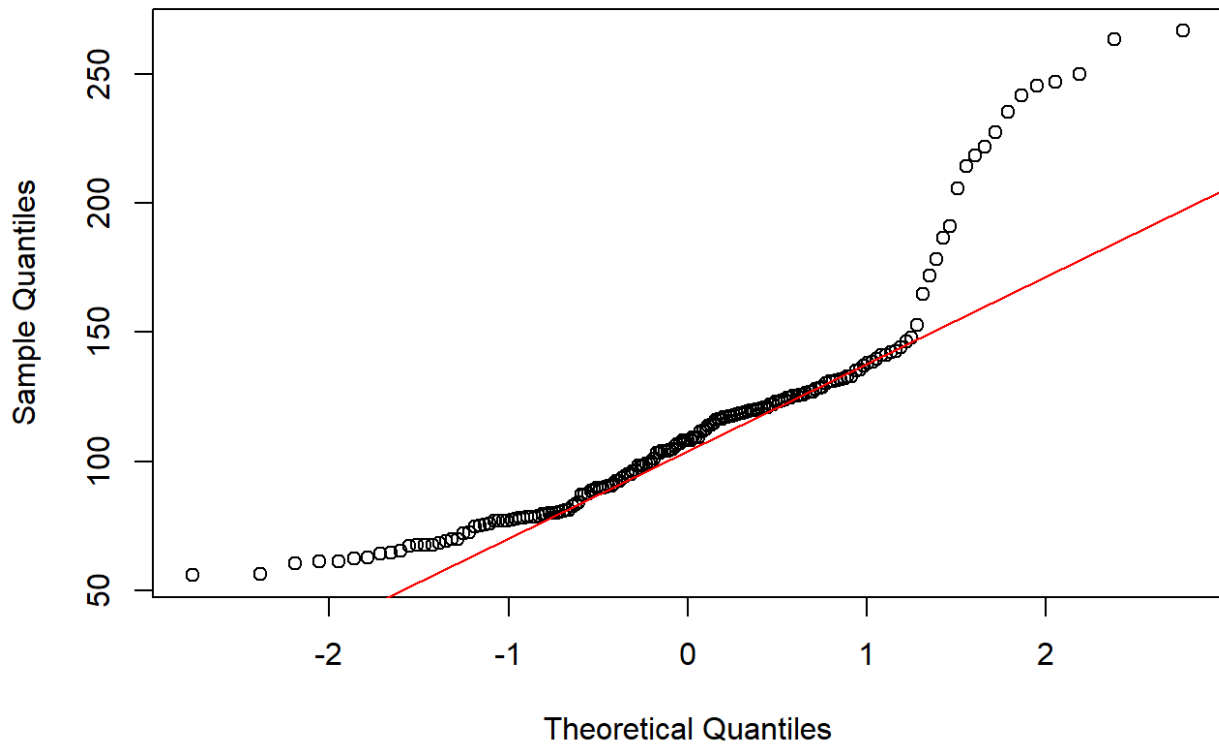
```
## [1] 175
```

```
hist(db_cc$cc14, breaks=30)
```



```
qqnorm(db_cc$cc14); qqline(db_cc$cc14, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$ccl4)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##   55.94   81.14  108.10  113.80  126.68  266.57
```

```
min(db_cc$ccl4, na.rm=TRUE)
```

```
## [1] 55.94387
```



```
db$ccl4_z <- scale(log(db$ccl4))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$ccl4_z <- scale(log(db_cc$ccl4))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

ccl4Model_hetero <- glmmTMB(
  ccl4_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(ccl4Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## ccl4_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion:                ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    458.7    518.8   -210.4    420.7      156
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id       (Intercept) 0.4349   0.6595
##   Residual              NA       NA
## Number of obs: 175, groups: id, 79
##
## Conditional model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    3.69759    2.80404   1.319 0.187281
## t1              1.12492    0.23667   4.753 2e-06 ***
## t2              1.28001    0.43312   2.955 0.003123 **
## GROUP1          0.66440    0.37104   1.791 0.073351 .
## GROUP2          0.82011    0.32753   2.504 0.012283 *
## GROUP3          0.57763    0.41722   1.384 0.166213
## Age            -0.05195    0.03620  -1.435 0.151268
## SexM            -0.21720    0.18729  -1.160 0.246176
## CIRS_severity_0 -0.18948    0.17531  -1.081 0.279789
## t1:GROUP1       -0.96437    0.35645  -2.705 0.006820 **
## t2:GROUP1       -1.70914    0.45881  -3.725 0.000195 ***
## t1:GROUP2       -1.00740    0.30364  -3.318 0.000907 ***
## t2:GROUP2       -1.21268    0.39862  -3.042 0.002348 **
## t1:GROUP3       -0.56258    0.38218  -1.472 0.141008
## t2:GROUP3       -1.05405    0.44551  -2.366 0.017984 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.34685    0.14547  -2.384  0.0171 *
## t1           -0.39983    0.28483  -1.404  0.1604
## t2           -0.04754    0.18651  -0.255  0.7988
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

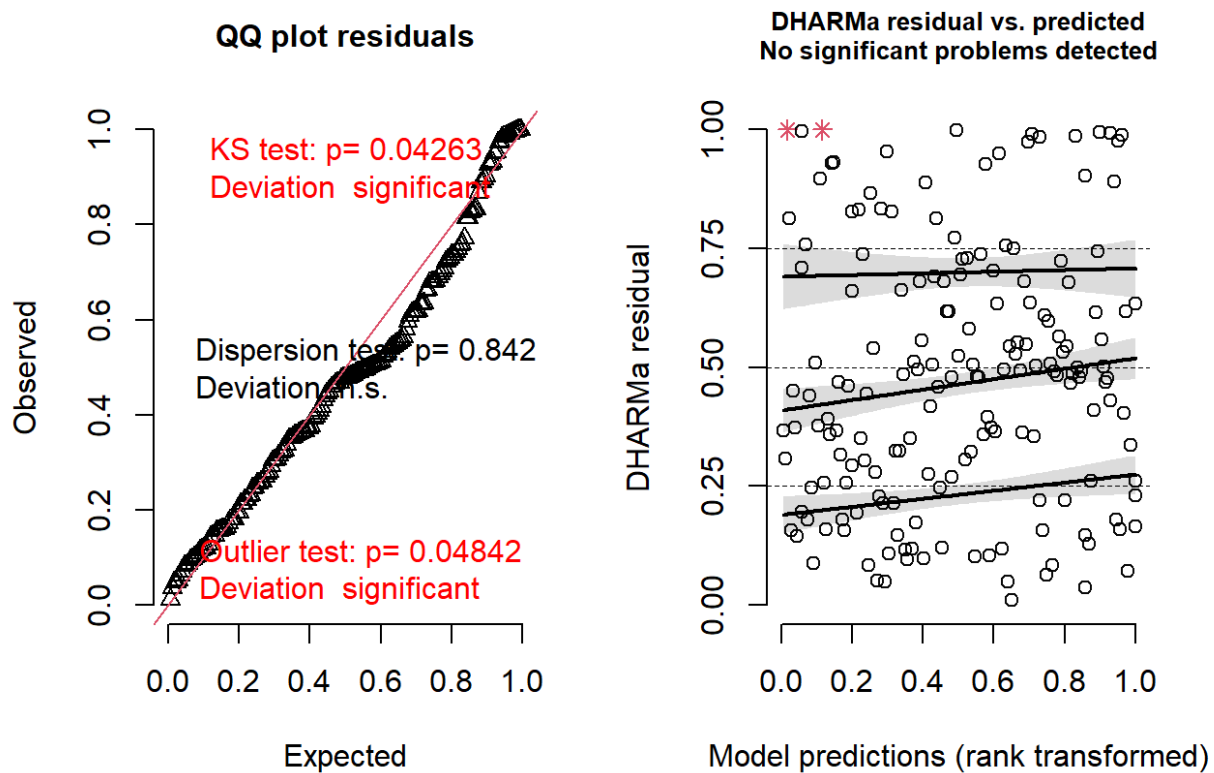
```
ccl4Model_homo <- glmmTMB(
  ccl4_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(ccl4Model_homo, ccl4Model_hetero, test="Chisq")
```

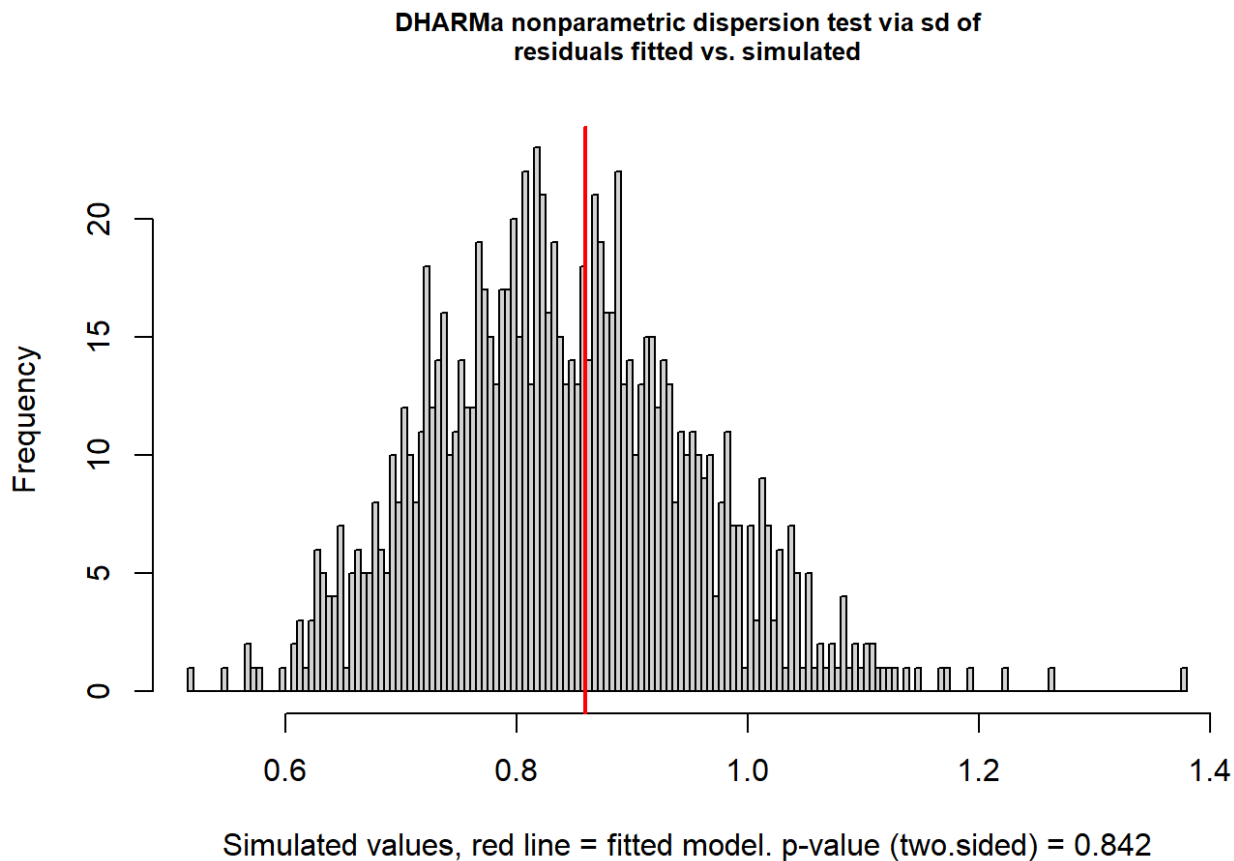
```
## Data: db_cc
## Models:
## ccl4Model_homo: ccl4_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severi
ty_0 + , zi=~0, disp=~1
## ccl4Model_homo:      (1 | id), zi=~0, disp=~t
## ccl4Model_hetero: ccl4_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## ccl4Model_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance Chisq Chi Df Pr(>Chis
q)
## ccl4Model_homo    17 457.21 511.01 -211.60   423.21
## ccl4Model_hetero  19 458.71 518.84 -210.36   420.71 2.494      2    0.28
74
```

```
sim_res <- simulateResiduals(ccl4Model_hetero, n = 1000)
plot(sim_res)
```

DHARMA residual

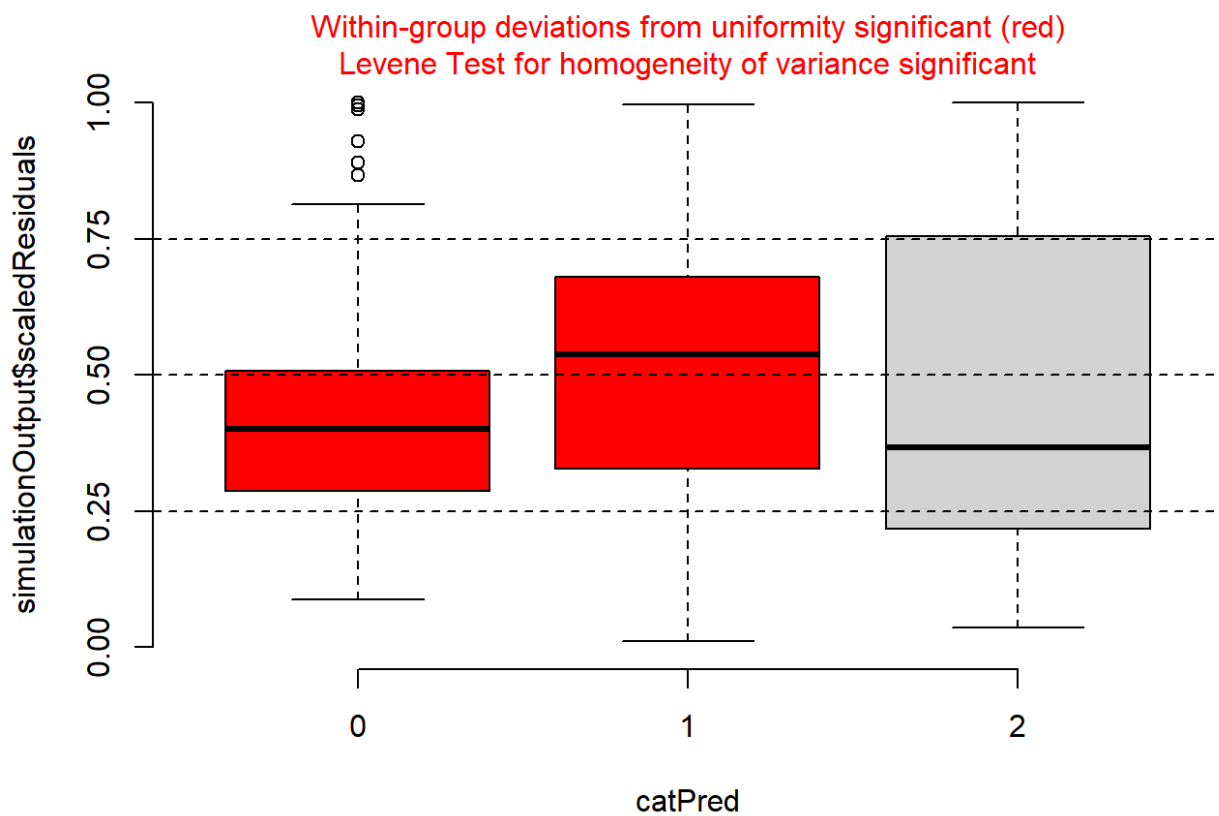


```
testDispersion(sim_res)
```



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 1.0234, p-value = 0.842
## alternative hypothesis: two.sided
```

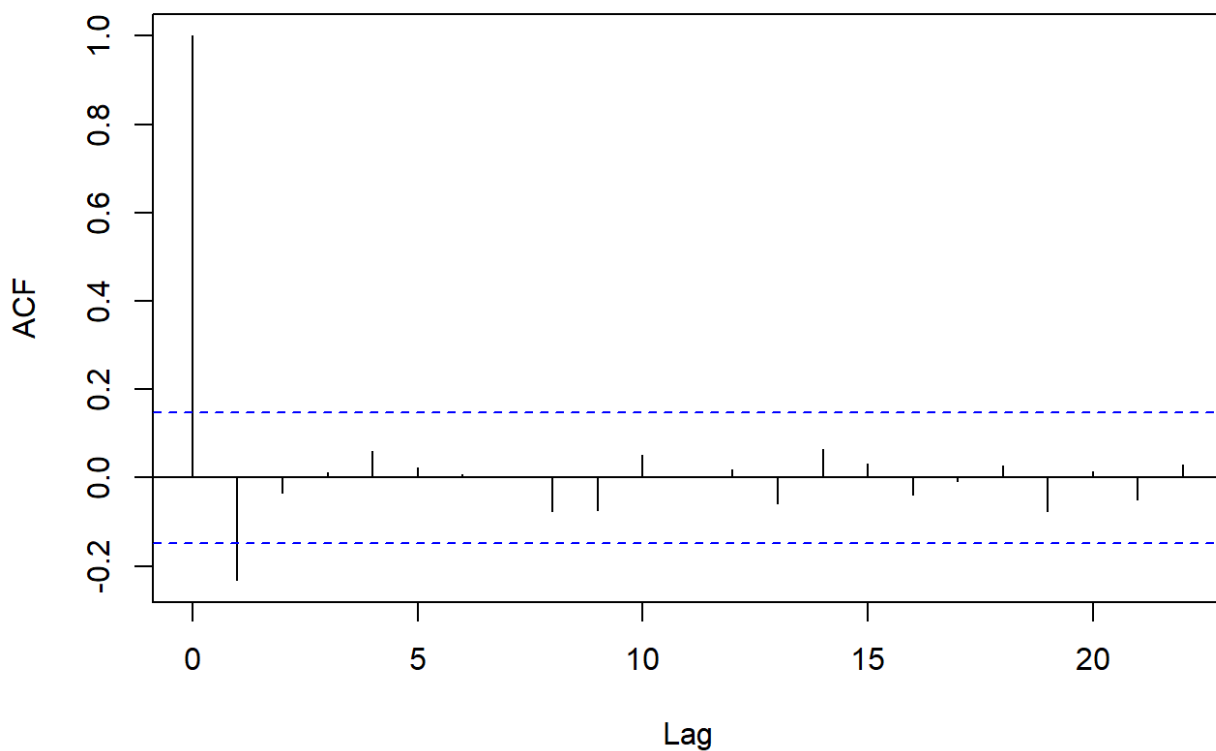
```
testQuantiles(sim_res, db_cc$t)
```



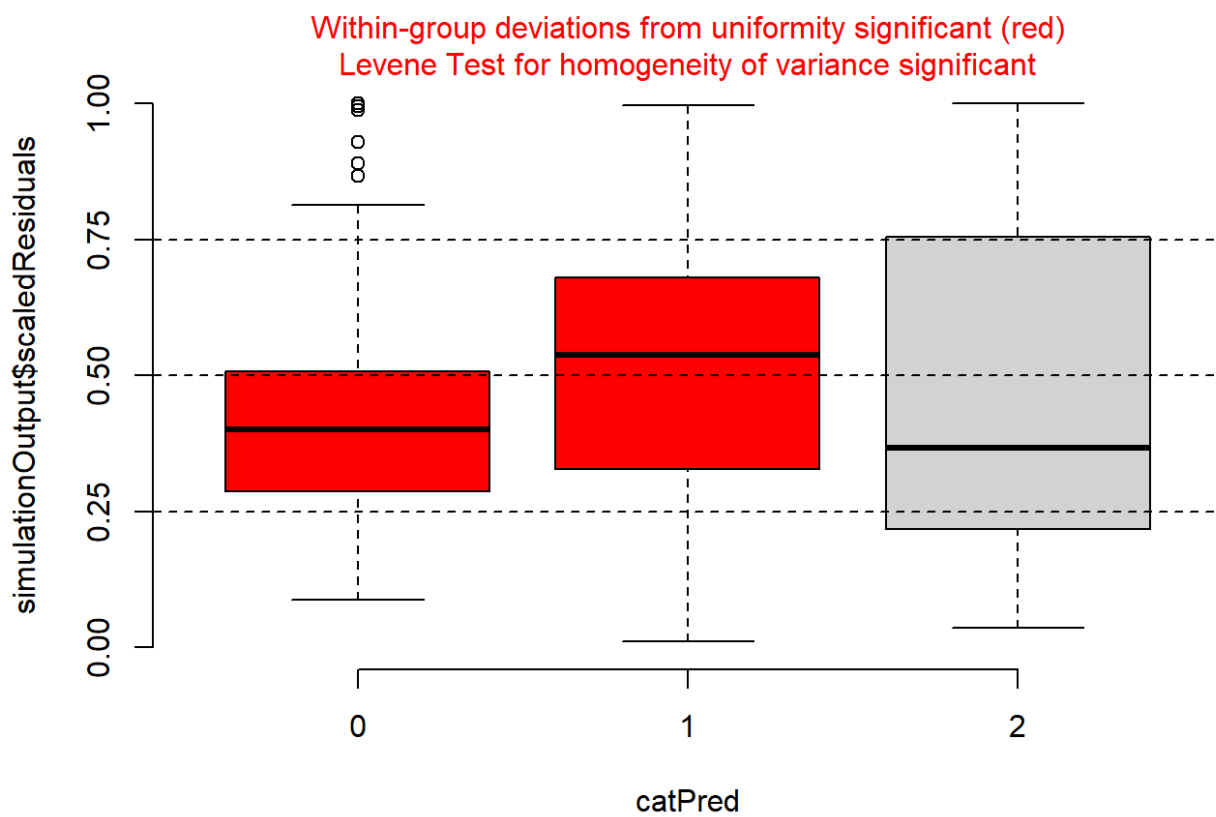
```
## NULL
```

```
acf(residuals(cc14Model_hetero))
```

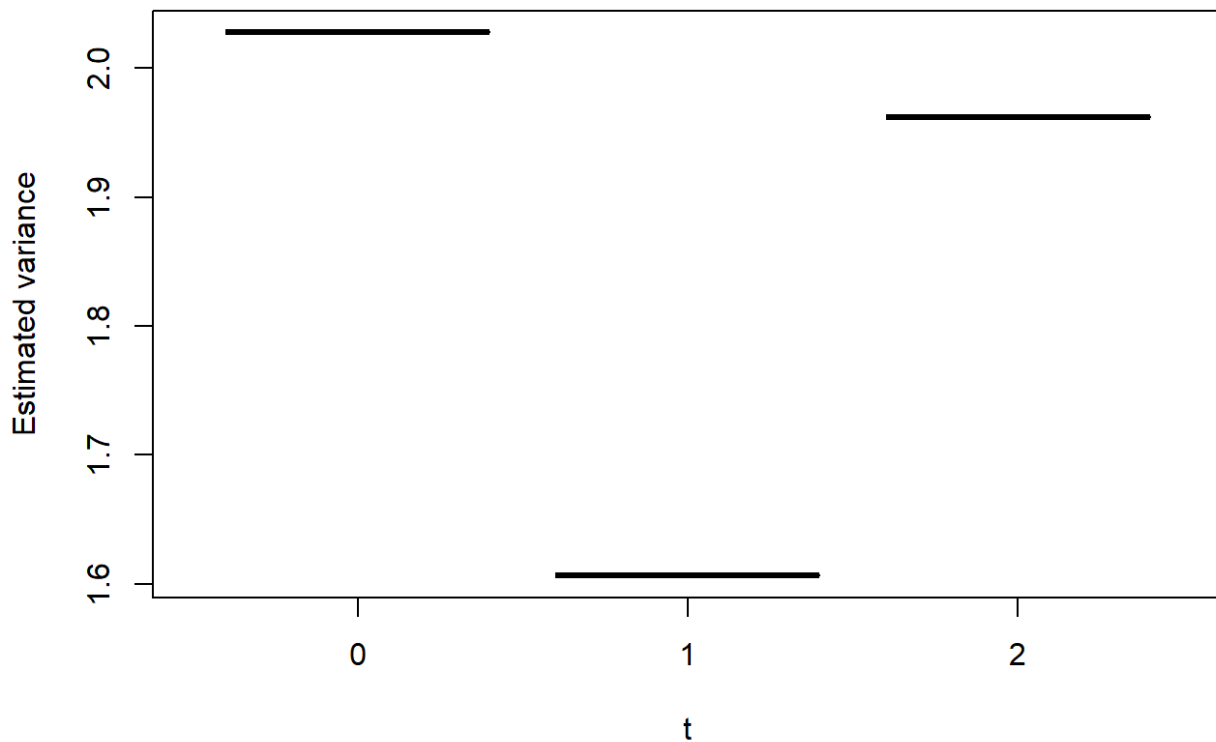
Series residuals(cc14Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

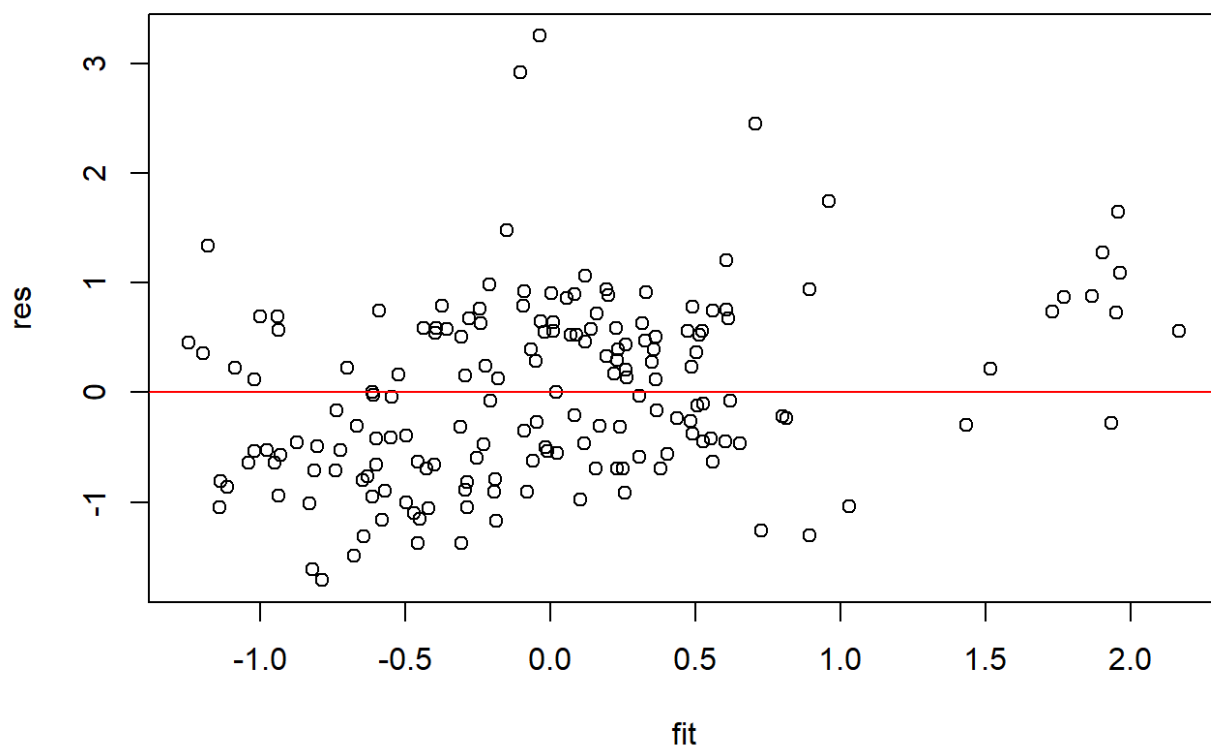


```
pred_disp <- predict(ccl4Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(ccl4Model_hetero, type="pearson")
fit <- fitted(ccl4Model_hetero)

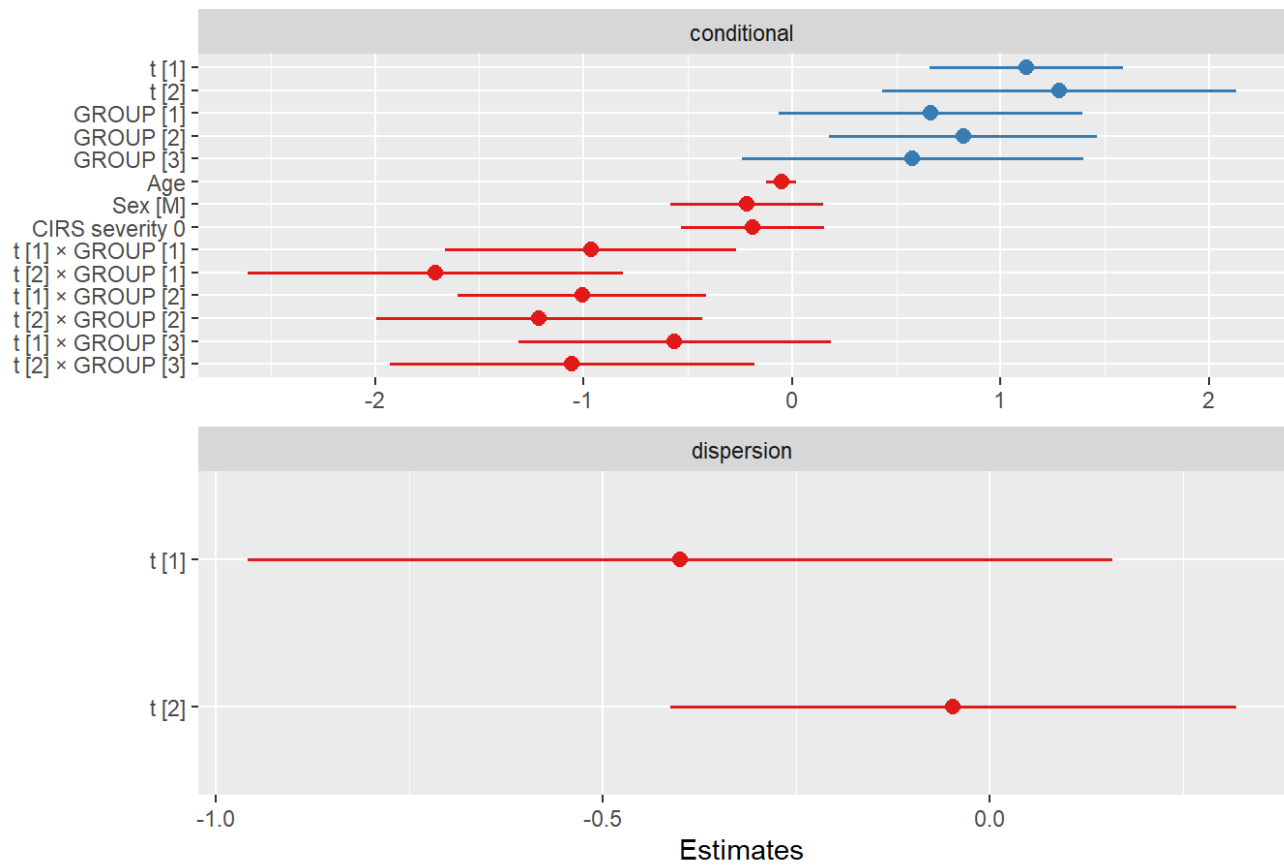
plot(fit, res)
abline(h=0, col="red")
```



5.1 Visualizzazione del modello

```
plot_model(ccl4Model_hetero)
```

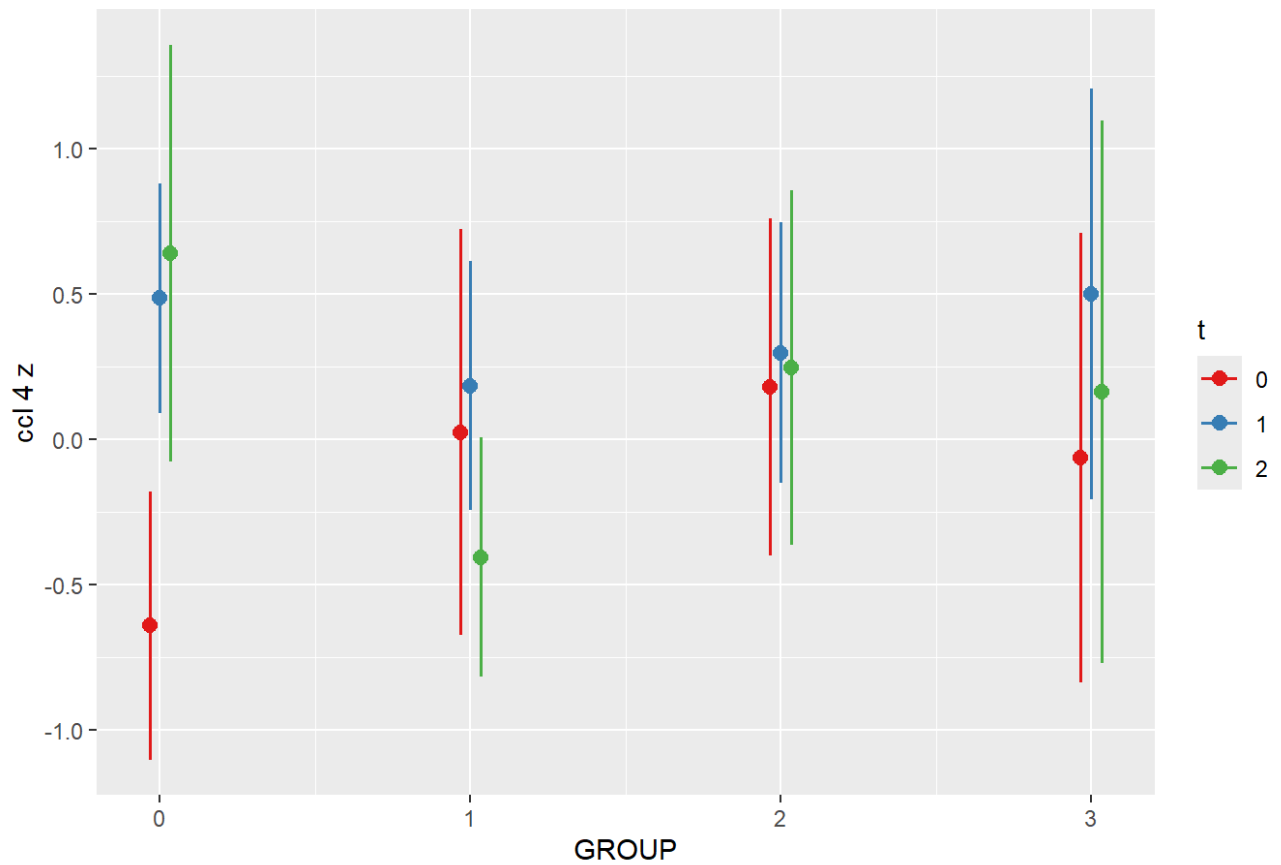

ccl 4 z



```
plot_model(ccl4Model_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

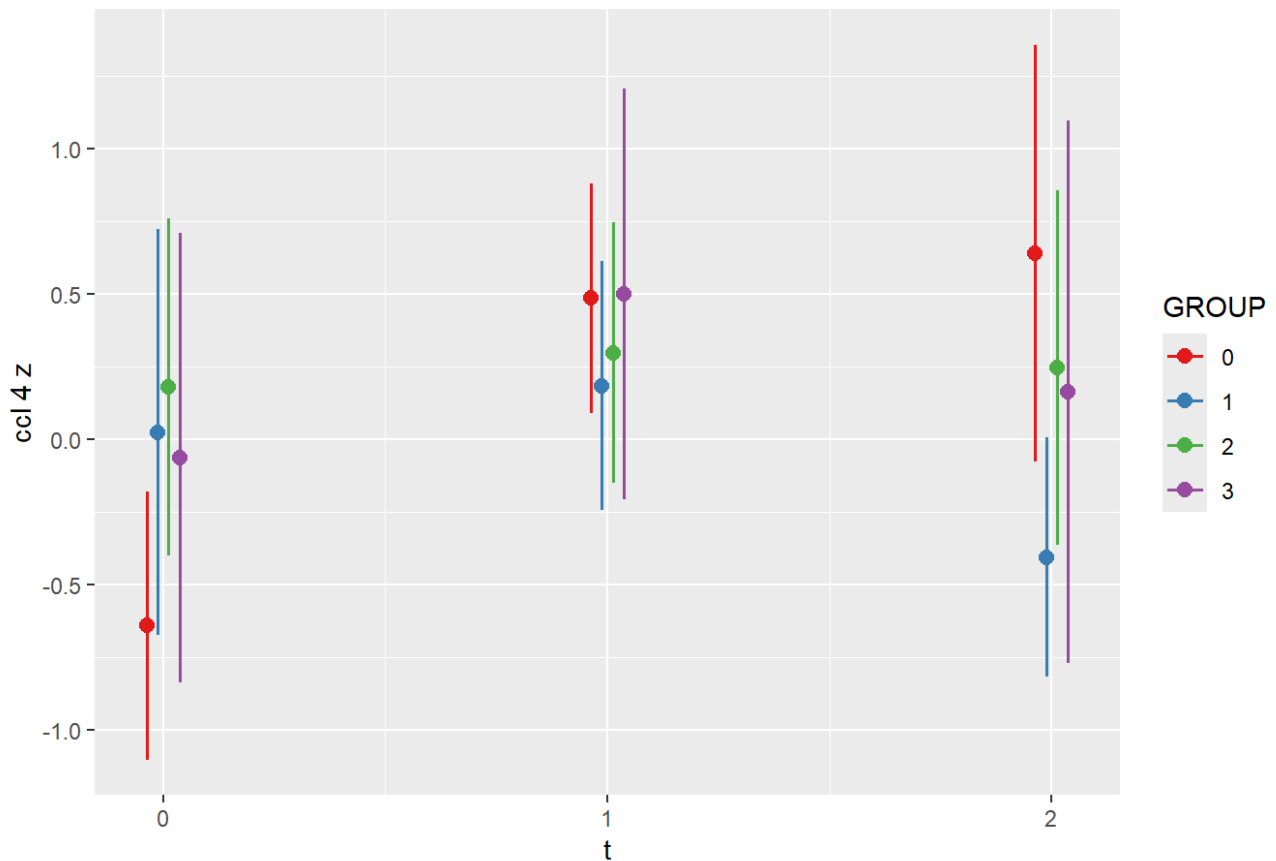
Predicted values of ccl 4 z



```
plot_model(cc14Model_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of ccl 4 z

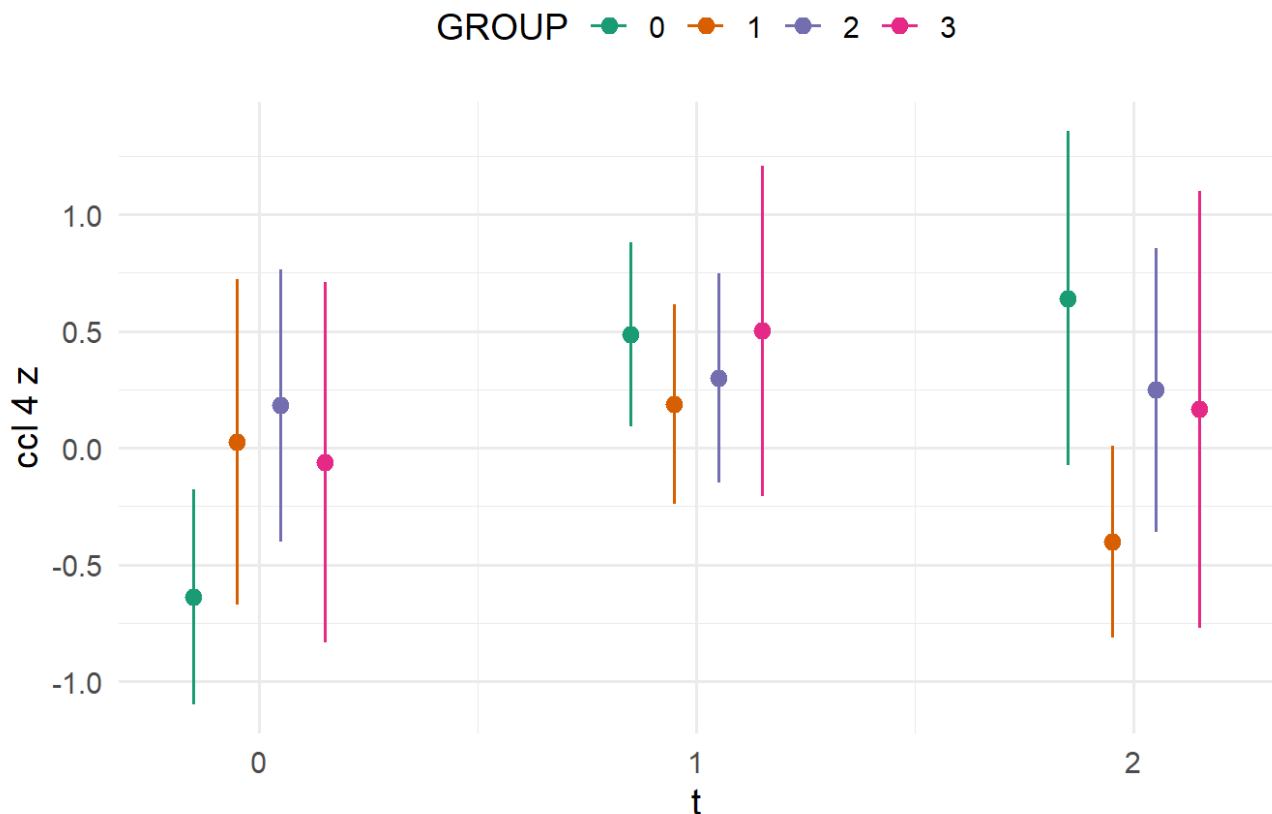


```
plot_model(ccl4Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

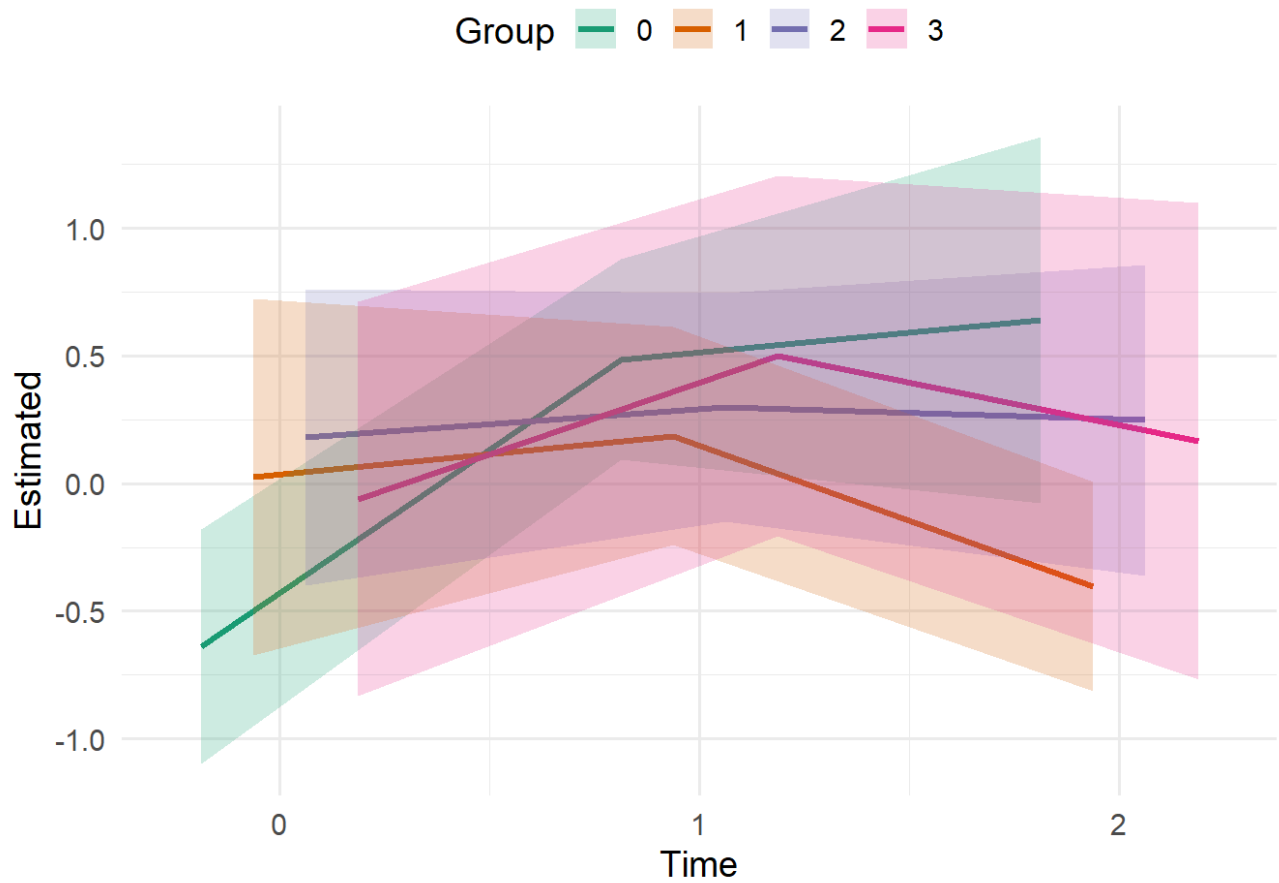
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of ccl 4 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(ccl4Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



5.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(ccl4Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.7479 0.217 Inf   -1.174   -0.322
##  1      -0.0835 0.360 Inf   -0.788    0.621
##  2       0.0722 0.294 Inf   -0.503    0.648
##  3      -0.1703 0.374 Inf   -0.903    0.562
##
## t = 1:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.3770 0.173 Inf    0.038    0.716
##  1       0.0771 0.213 Inf   -0.341    0.495
##  2       0.1897 0.222 Inf   -0.246    0.625
##  3       0.3921 0.337 Inf   -0.268    1.052
##
## t = 2:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.5321 0.353 Inf   -0.160    1.224
##  1      -0.5126 0.203 Inf   -0.910   -0.115
##  2       0.1395 0.301 Inf   -0.450    0.729
##  3       0.0557 0.457 Inf   -0.839    0.951
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(ccl4Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  contrast          estimate    SE  df z.ratio p.value
##  GROUP0 - GROUP1    0.2268 0.280 Inf   0.810  0.8496
##  GROUP0 - GROUP2   -0.0801 0.255 Inf  -0.314  0.9893
##  GROUP0 - GROUP3   -0.0387 0.327 Inf  -0.118  0.9994
##  GROUP1 - GROUP2   -0.3069 0.275 Inf  -1.115  0.6800
##  GROUP1 - GROUP3   -0.2655 0.389 Inf  -0.682  0.9041
##  GROUP2 - GROUP3    0.0413 0.362 Inf   0.114  0.9995
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(ccl4Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.6644 0.371 Inf  -1.791  0.2778
## GROUP0 - GROUP2 -0.8201 0.328 Inf  -2.504  0.0593
## GROUP0 - GROUP3 -0.5776 0.417 Inf  -1.384  0.5091
## GROUP1 - GROUP2 -0.1557 0.398 Inf  -0.391  0.9797
## GROUP1 - GROUP3  0.0868 0.497 Inf   0.175  0.9981
## GROUP2 - GROUP3  0.2425 0.458 Inf   0.529  0.9521
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.3000 0.295 Inf   1.016  0.7398
## GROUP0 - GROUP2  0.1873 0.269 Inf   0.696  0.8988
## GROUP0 - GROUP3 -0.0150 0.361 Inf  -0.042  1.0000
## GROUP1 - GROUP2 -0.1127 0.304 Inf  -0.370  0.9827
## GROUP1 - GROUP3 -0.3150 0.425 Inf  -0.740  0.8808
## GROUP2 - GROUP3 -0.2023 0.398 Inf  -0.508  0.9572
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  1.0447 0.409 Inf   2.556  0.0518
## GROUP0 - GROUP2  0.3926 0.372 Inf   1.055  0.7171
## GROUP0 - GROUP3  0.4764 0.427 Inf   1.115  0.6804
## GROUP1 - GROUP2 -0.6522 0.357 Inf  -1.825  0.2616
## GROUP1 - GROUP3 -0.5683 0.503 Inf  -1.130  0.6710
## GROUP2 - GROUP3  0.0839 0.468 Inf   0.179  0.9980
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(ccl4Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    -1.1249 0.237 Inf  -4.753 <0.0001
## t0 - t2    -1.2800 0.433 Inf  -2.955  0.0088
## t1 - t2    -0.1551 0.346 Inf  -0.448  0.8951
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    -0.1606 0.308 Inf  -0.520  0.8613
## t0 - t2     0.4291 0.383 Inf   1.122  0.5005
## t1 - t2     0.5897 0.228 Inf   2.583  0.0265
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    -0.1175 0.281 Inf  -0.418  0.9083
## t0 - t2    -0.0673 0.410 Inf  -0.164  0.9852
## t1 - t2     0.0502 0.302 Inf   0.166  0.9849
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    -0.5623 0.367 Inf  -1.532  0.2758
## t0 - t2    -0.2260 0.508 Inf  -0.445  0.8967
## t1 - t2     0.3364 0.413 Inf   0.815  0.6940
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

6 Model cx3cl1

```
bptest(lm(cx3cl1 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(cx3cl1 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 9.4494, df = 14, p-value = 0.8012
```



```
vars <- c("cx3cl1", "t", "GROUP", "Age", "Sex", "CIRS_severity_0", "id")
db_cc <- db[complete.cases(db[, vars]), vars]

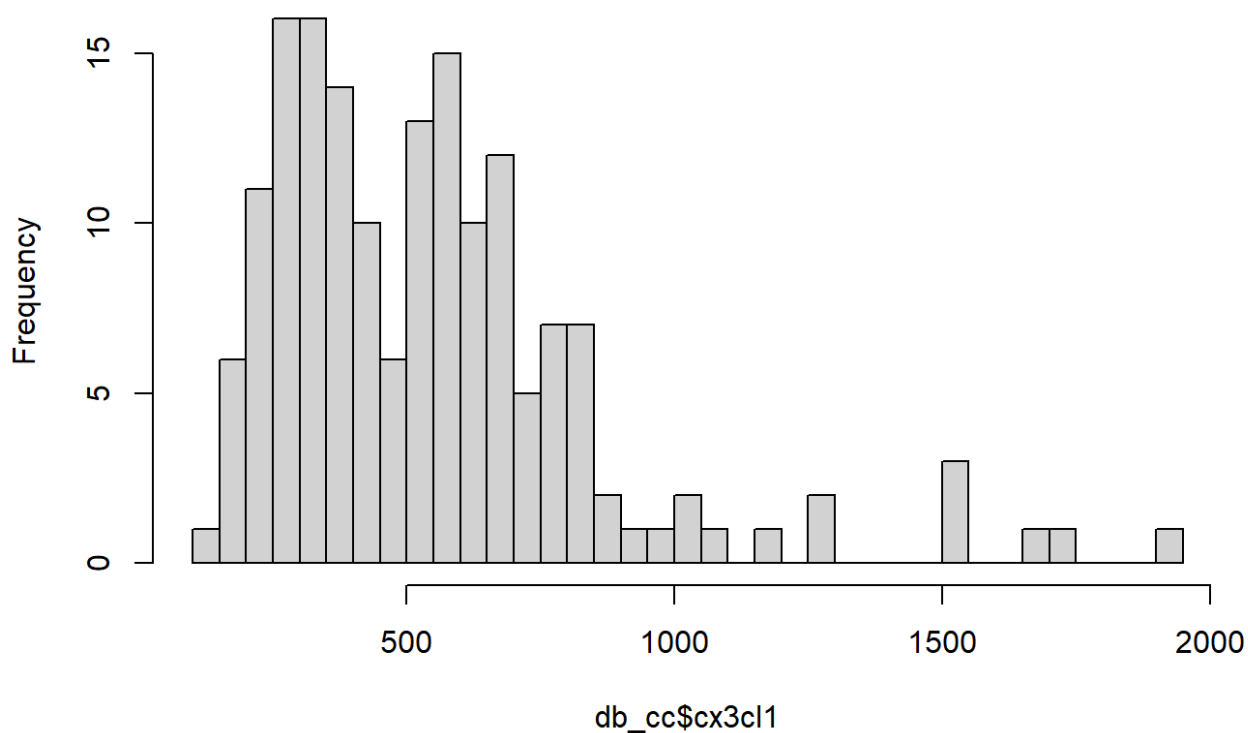
nrow(db); nrow(db_cc)
```

```
## [1] 243
```

```
## [1] 165
```

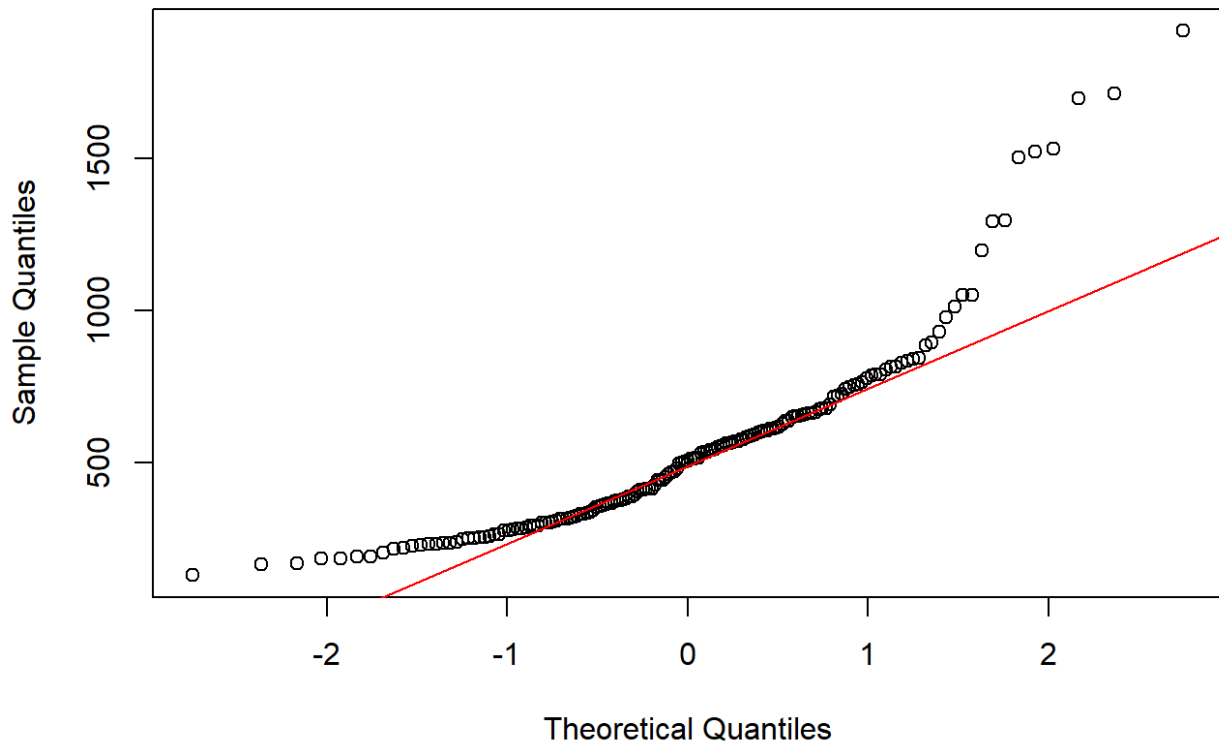
```
hist(db_cc$cx3cl1, breaks=30)
```

Histogram of db_cc\$cx3cl1



```
qqnorm(db_cc$cx3cl1); qqline(db_cc$cx3cl1, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$cx3cl1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##   130.3   316.9   507.0   547.6   661.5   1918.3
```

```
min(db_cc$cx3cl1, na.rm=TRUE)
```

```
## [1] 130.2519
```

```
db$cx3cl1_z <- scale(log(db$cx3cl1))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$cx3cl1_z <- scale(log(db_cc$cx3cl1))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

cx3cl1Model_hetero <- glmmTMB(
  cx3cl1_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(cx3cl1Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## cx3cl1_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1
| id)
## Dispersion:                ~t
## Data: db_cc
##
##          AIC          BIC      logLik -2*log(L)  df.resid
##          400          459       -181      362      146
##
## Random effects:
##
## Conditional model:
##   Groups   Name          Variance Std.Dev.
##   id       (Intercept) 0.4157   0.6448
##   Residual                NA        NA
## Number of obs: 165, groups: id, 78
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -2.572195   2.601314  -0.989 0.322758
## t1             0.786798   0.221802   3.547 0.000389 ***
## t2             0.361628   0.424505   0.852 0.394280
## GROUP1         0.466182   0.375265   1.242 0.214136
## GROUP2         0.157801   0.337525   0.468 0.640125
## GROUP3        -0.489471   0.424269  -1.154 0.248631
## Age            0.031994   0.033727   0.949 0.342816
## SexM           -0.267923   0.170007  -1.576 0.115038
## CIRS_severity_0 -0.207713   0.162726  -1.276 0.201793
## t1:GROUP1      -0.052071   0.334056  -0.156 0.876131
## t2:GROUP1      -0.585946   0.460350  -1.273 0.203079
## t1:GROUP2       0.003731   0.285354   0.013 0.989567
## t2:GROUP2      -0.069895   0.432293  -0.162 0.871554
## t1:GROUP3       0.071667   0.342776   0.209 0.834387
## t2:GROUP3       0.266529   0.463303   0.575 0.565101
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.31098    0.12901  -2.410  0.0159 *
## t1           -1.27703    0.75182  -1.699  0.0894 .
## t2           -0.07411    0.16093  -0.460  0.6451
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

cx3cl1Model_homo <- glmmTMB(
  cx3cl1_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(cx3cl1Model_homo, cx3cl1Model_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## cx3cl1Model_homo: cx3cl1_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_se
verity_0 + , zi=~0, disp=~1
## cx3cl1Model_homo:      (1 | id), zi=~0, disp=~t
## cx3cl1Model_hetero: cx3cl1_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_
severity_0 + , zi=~0, disp=~1
## cx3cl1Model_hetero:      (1 | id), zi=~0, disp=~t
##
##          Df      AIC      BIC  logLik deviance Chisq Chi Df Pr(>Ch
isq)
## cx3cl1Model_homo    17 408.41 461.21 -187.20   374.41
## cx3cl1Model_hetero  19 399.98 458.99 -180.99   361.98 12.43      2  0.00
1999 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

sim_res <- simulateResiduals(cx3cl1Model_hetero, n = 1000)
plot(sim_res)

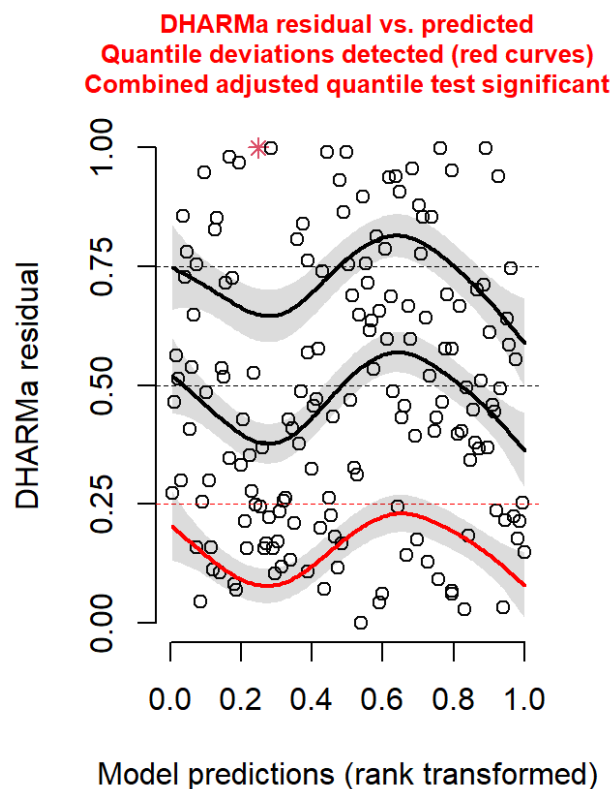
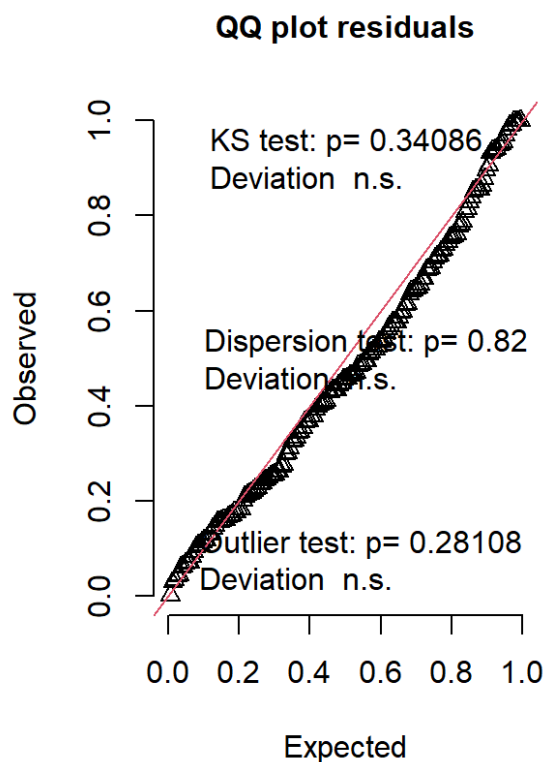
```

```

## Warning in newton(lsp = lsp, X = G$X, y = G$y, Eb = G$Eb, UrS = G$UrS, L
= G$L,
## : Fitting terminated with step failure - check results carefully

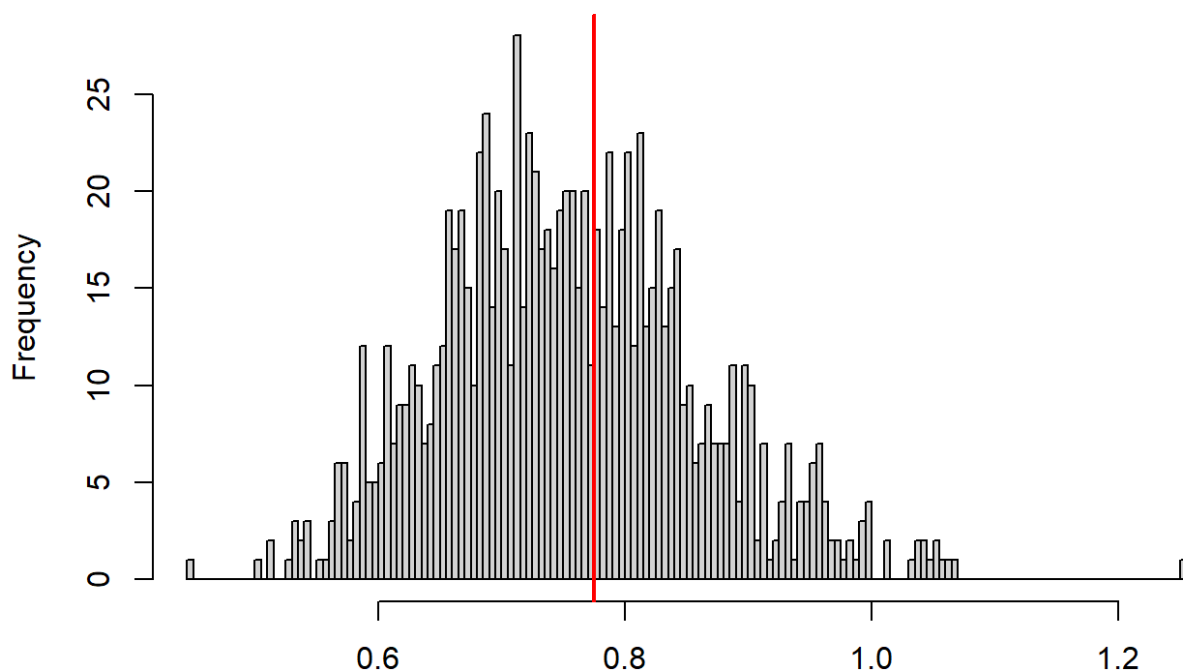
```

DHARMA residual



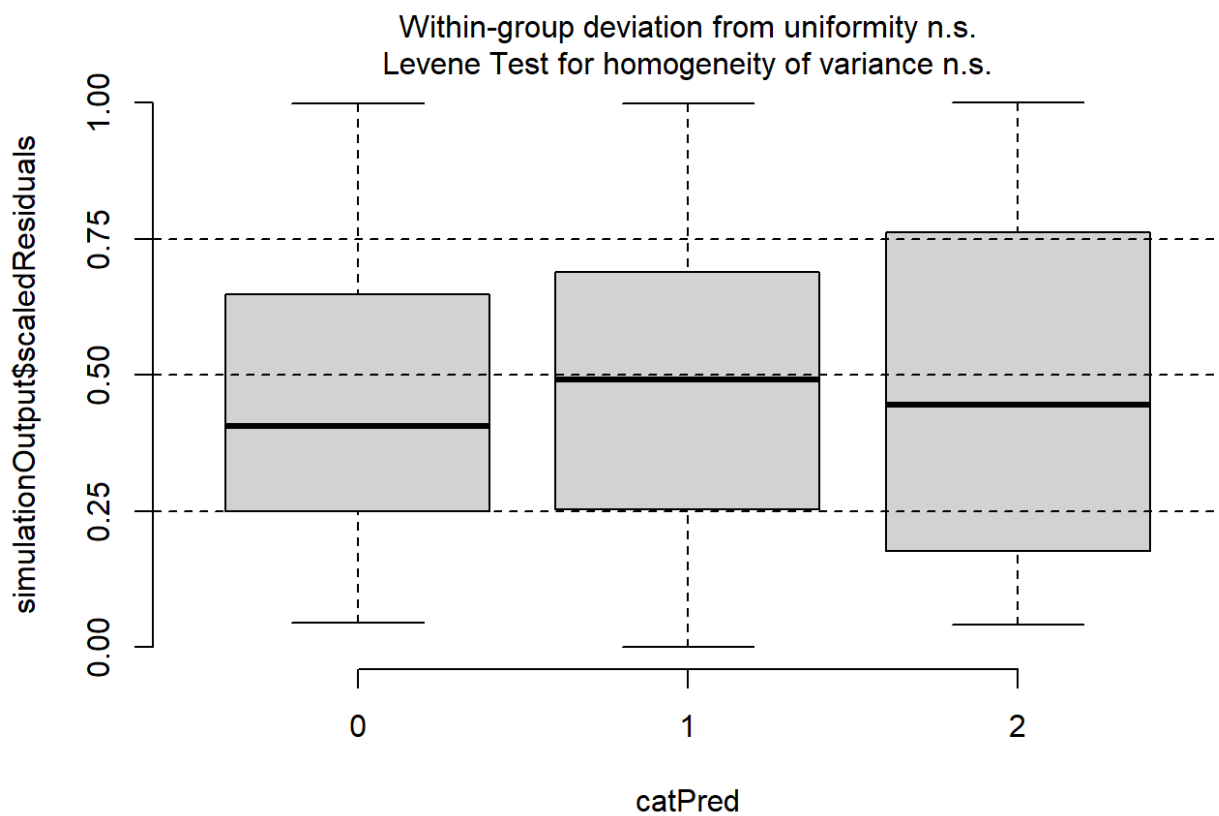
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 1.0264, p-value = 0.82
## alternative hypothesis: two.sided
```

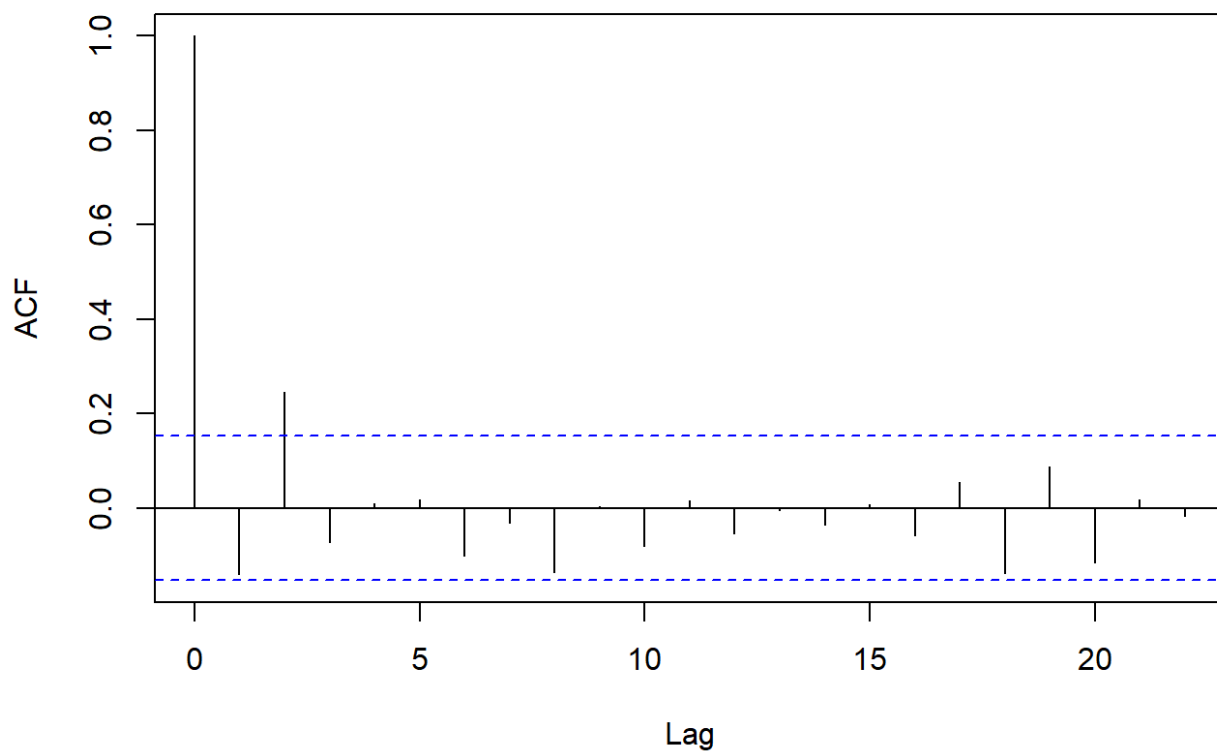
```
testQuantiles(sim_res, db_cc$t)
```



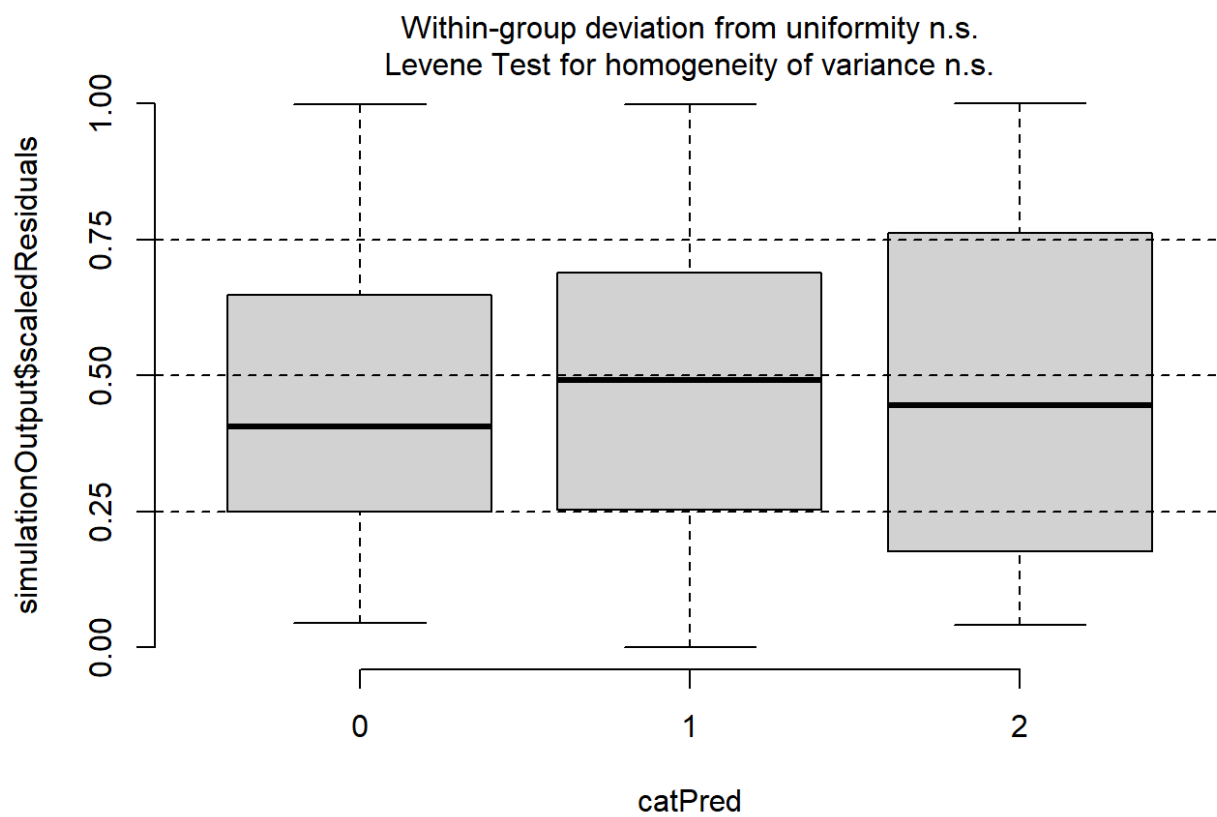
```
## NULL
```

```
acf(residuals(cx3cl1Model_hetero))
```

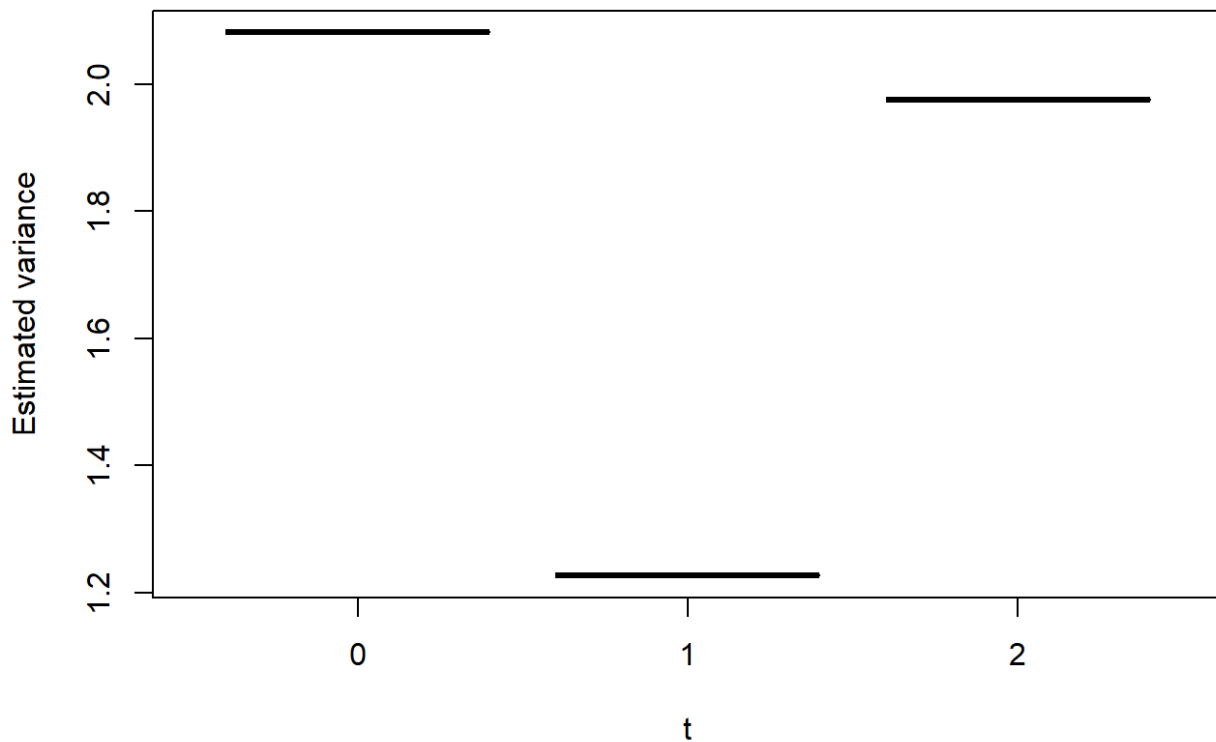
Series residuals(cx3cl1Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

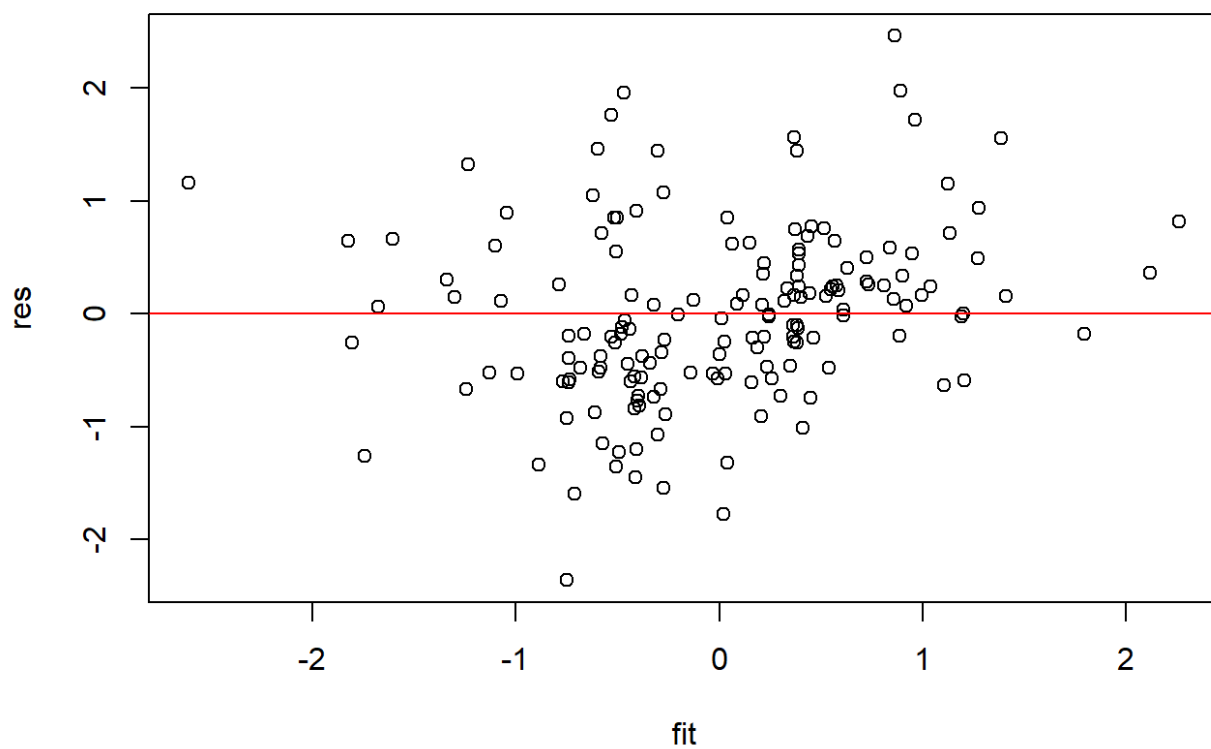



```
pred_disp <- predict(cx3cl1Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



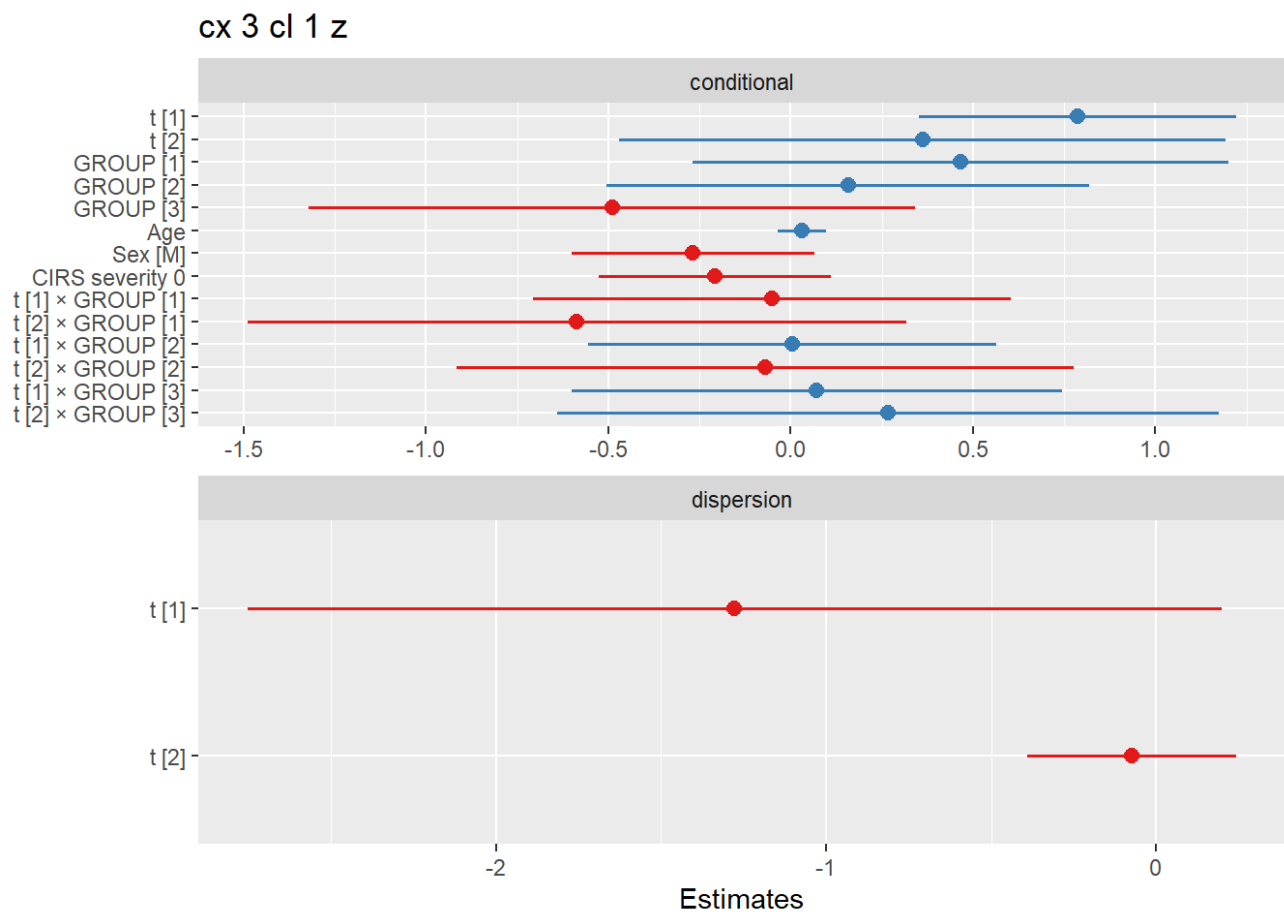
```
res <- residuals(cx3cl1Model_hetero, type="pearson")
fit <- fitted(cx3cl1Model_hetero)

plot(fit, res)
abline(h=0, col="red")
```



6.1 Visualizzazione del modello

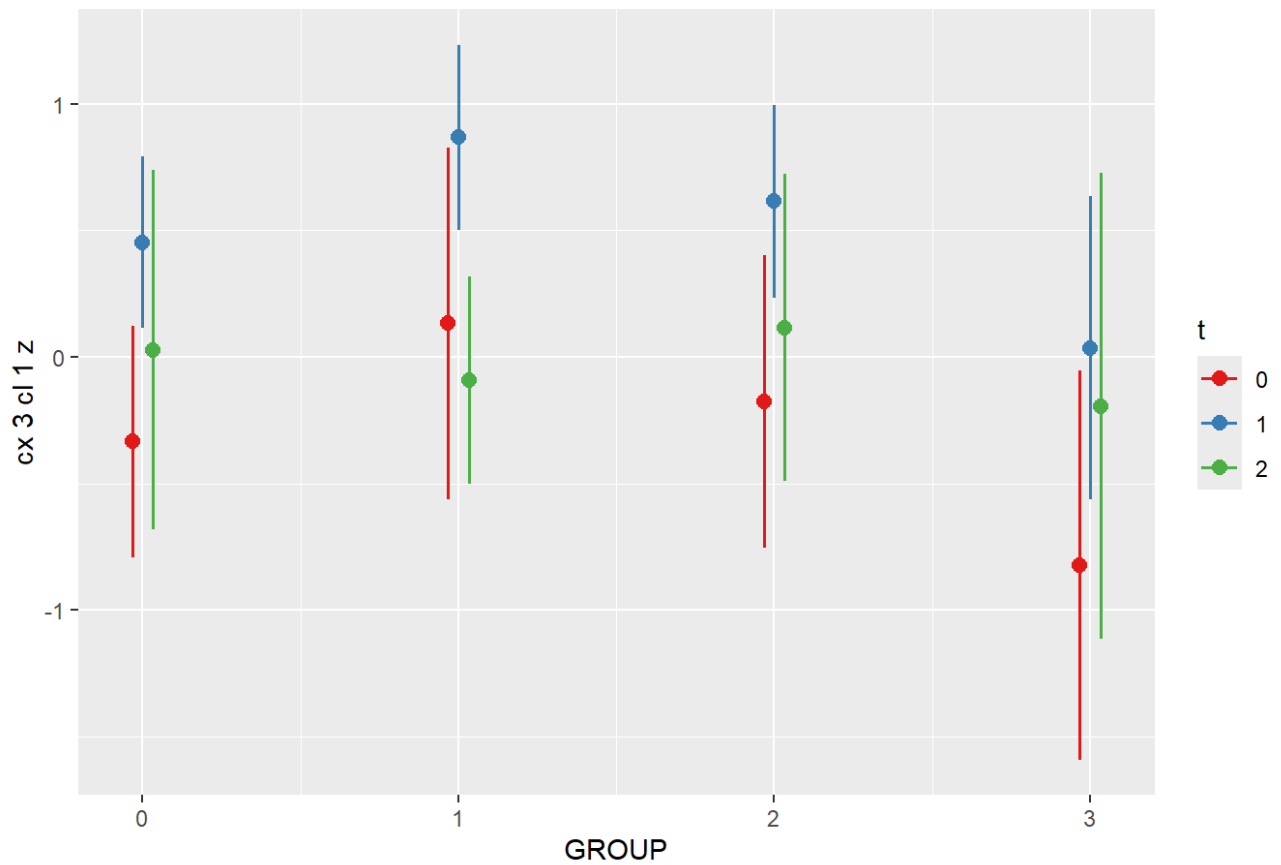
```
plot_model(cx3cl1Model_hetero)
```



```
plot_model(cx3cl1Model_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

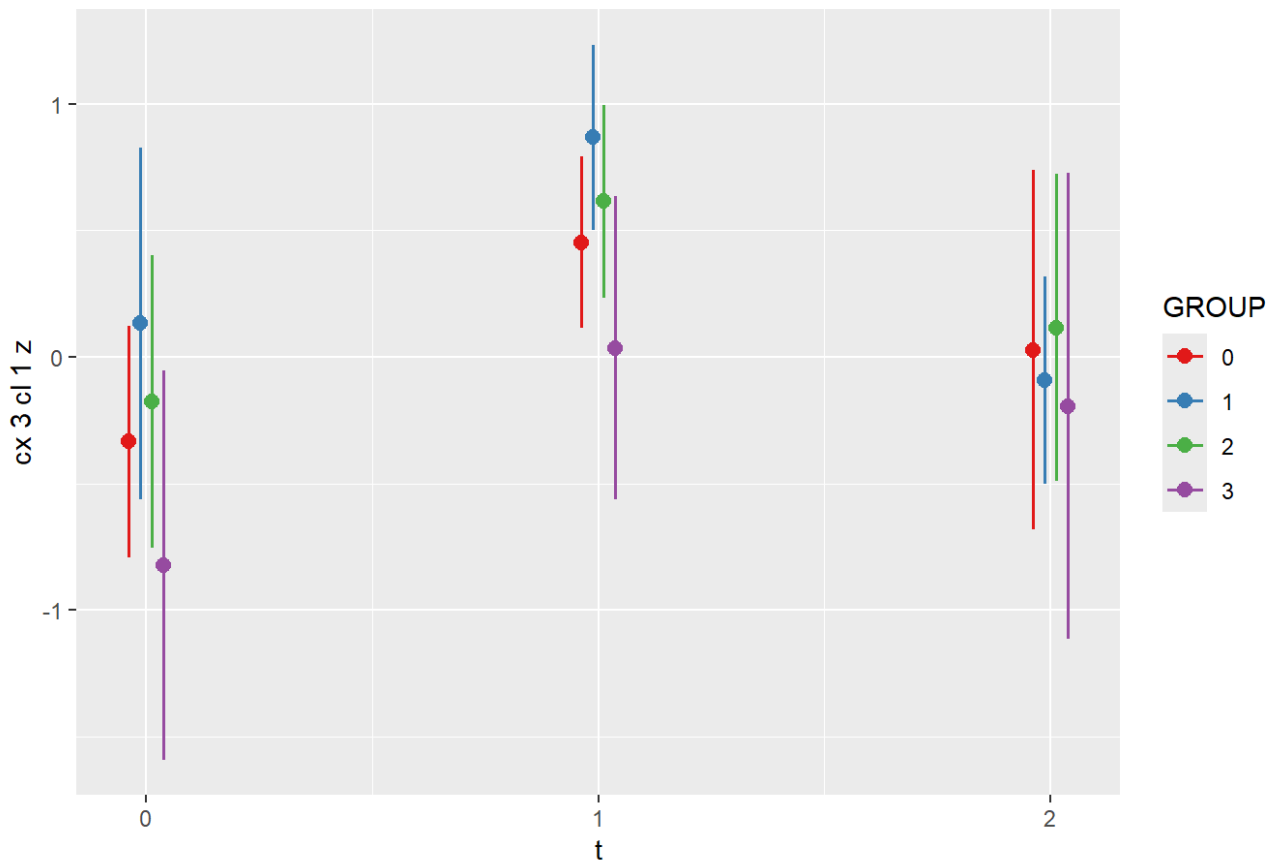
Predicted values of cx 3 cl 1 z



```
plot_model(cx3cl1Model_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of cx 3 cl 1 z

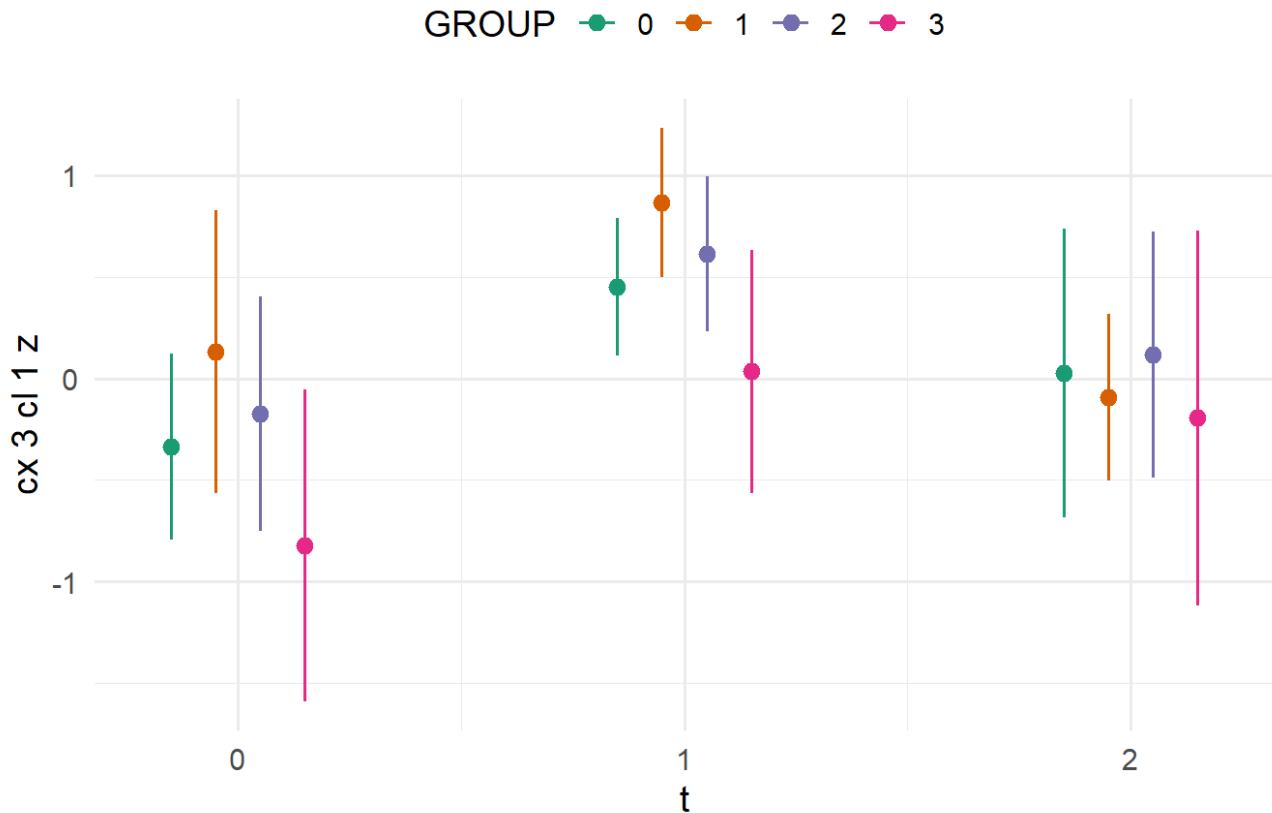


```
plot_model(cx3cl1Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

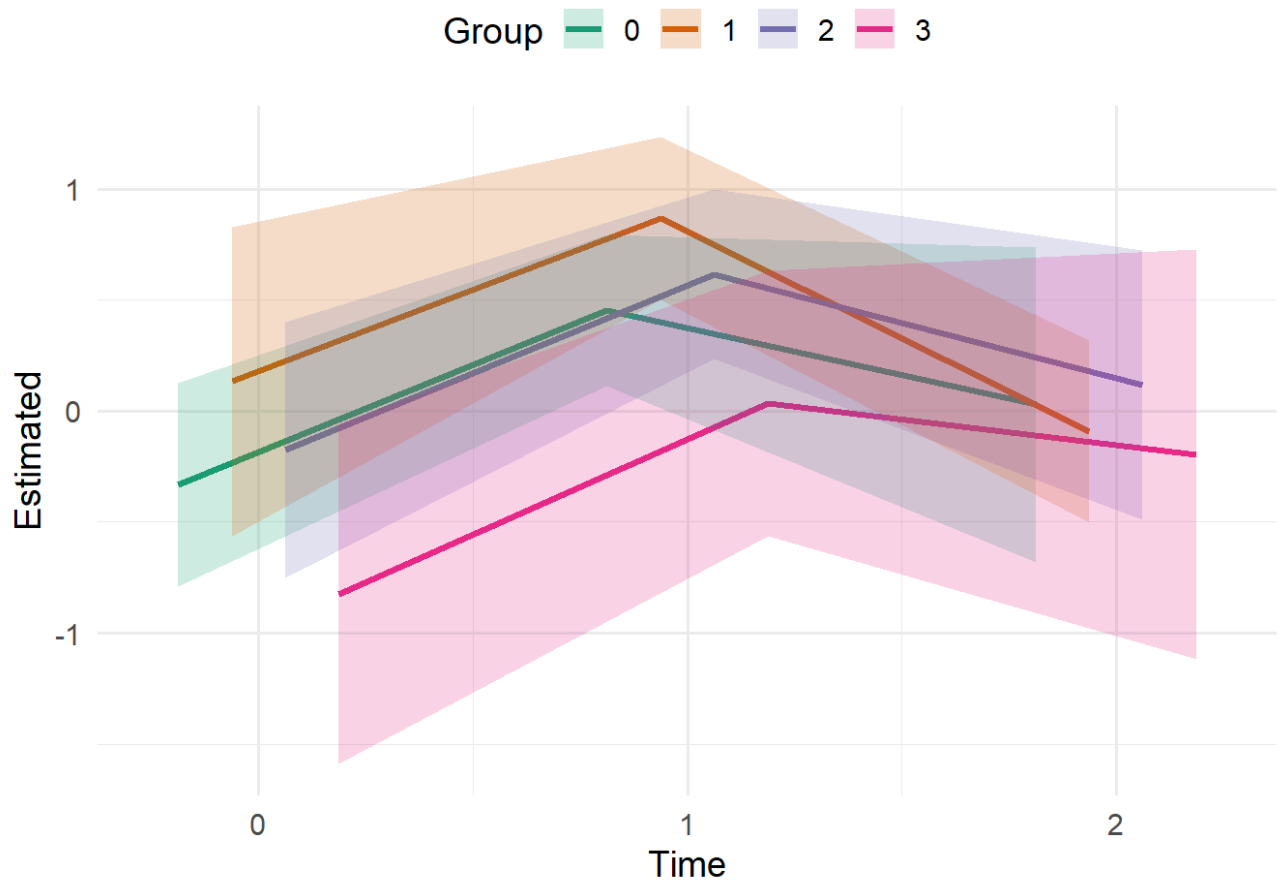
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of cx 3 cl 1 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per g
ruppo)
pred <- ggpredict(cx3cl1Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = grou
p)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



6.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(cx3cl1Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP    emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.46777 0.223 Inf   -0.9049  -0.0307
##  1      -0.00158 0.360 Inf   -0.7075   0.7043
##  2      -0.30996 0.294 Inf   -0.8856   0.2656
##  3      -0.95724 0.376 Inf   -1.6937  -0.2208
##
## t = 1:
##  GROUP    emmean    SE  df asymp.LCL asymp.UCL
##  0       0.31903 0.147 Inf    0.0316   0.6065
##  1       0.73314 0.186 Inf    0.3690   1.0972
##  2       0.48056 0.187 Inf    0.1136   0.8476
##  3      -0.09877 0.281 Inf   -0.6490   0.4514
##
## t = 2:
##  GROUP    emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.10614 0.349 Inf   -0.7899   0.5777
##  1      -0.22590 0.203 Inf   -0.6240   0.1722
##  2      -0.01823 0.299 Inf   -0.6039   0.5675
##  3      -0.32908 0.451 Inf   -1.2123   0.5542
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(cx3cl1Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  contrast          estimate    SE  df z.ratio p.value
##  GROUP0 - GROUP1    -0.254 0.273 Inf   -0.930  0.7889
##  GROUP0 - GROUP2    -0.136 0.251 Inf   -0.542  0.9488
##  GROUP0 - GROUP3     0.377 0.320 Inf    1.177  0.6415
##  GROUP1 - GROUP2     0.118 0.273 Inf    0.431  0.9732
##  GROUP1 - GROUP3     0.630 0.378 Inf    1.666  0.3419
##  GROUP2 - GROUP3     0.512 0.351 Inf    1.462  0.4606
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(cx3cl1Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```



```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.4662 0.375 Inf  -1.242  0.5999
## GROUP0 - GROUP2 -0.1578 0.338 Inf  -0.468  0.9662
## GROUP0 - GROUP3  0.4895 0.424 Inf   1.154  0.6561
## GROUP1 - GROUP2  0.3084 0.407 Inf   0.758  0.8732
## GROUP1 - GROUP3  0.9557 0.500 Inf   1.912  0.2230
## GROUP2 - GROUP3  0.6473 0.464 Inf   1.395  0.5023
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.4141 0.256 Inf  -1.619  0.3677
## GROUP0 - GROUP2 -0.1615 0.225 Inf  -0.718  0.8899
## GROUP0 - GROUP3  0.4178 0.300 Inf   1.392  0.5041
## GROUP1 - GROUP2  0.2526 0.266 Inf   0.948  0.7786
## GROUP1 - GROUP3  0.8319 0.365 Inf   2.281  0.1023
## GROUP2 - GROUP3  0.5793 0.331 Inf   1.752  0.2970
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.1198 0.405 Inf   0.296  0.9910
## GROUP0 - GROUP2 -0.0879 0.391 Inf  -0.225  0.9960
## GROUP0 - GROUP3  0.2229 0.435 Inf   0.513  0.9561
## GROUP1 - GROUP2 -0.2077 0.354 Inf  -0.586  0.9364
## GROUP1 - GROUP3  0.1032 0.497 Inf   0.208  0.9968
## GROUP2 - GROUP3  0.3108 0.476 Inf   0.653  0.9146
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(cx3cl1Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.787 0.222 Inf  -3.547  0.0011
## t0 - t2      -0.362 0.425 Inf  -0.852  0.6707
## t1 - t2       0.425 0.331 Inf   1.285  0.4038
##
## GROUP = 1:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.735 0.296 Inf  -2.480  0.0350
## t0 - t2       0.224 0.385 Inf   0.582  0.8296
## t1 - t2       0.959 0.206 Inf   4.650 <0.0001
##
## GROUP = 2:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.791 0.265 Inf  -2.978  0.0082
## t0 - t2      -0.292 0.402 Inf  -0.726  0.7481
## t1 - t2       0.499 0.269 Inf   1.857  0.1513
##
## GROUP = 3:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.858 0.326 Inf  -2.629  0.0233
## t0 - t2      -0.628 0.505 Inf  -1.243  0.4279
## t1 - t2       0.230 0.370 Inf   0.623  0.8077
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

7 Model cxcl10

```
bptest(lm(cxcl10 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(cxcl10 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 21.935, df = 14, p-value = 0.07995
```

```
vars <- c("cxcl10", "t", "GROUP", "Age", "Sex", "CIRS_severity_0", "id")
db_cc <- db[complete.cases(db[, vars]), vars]

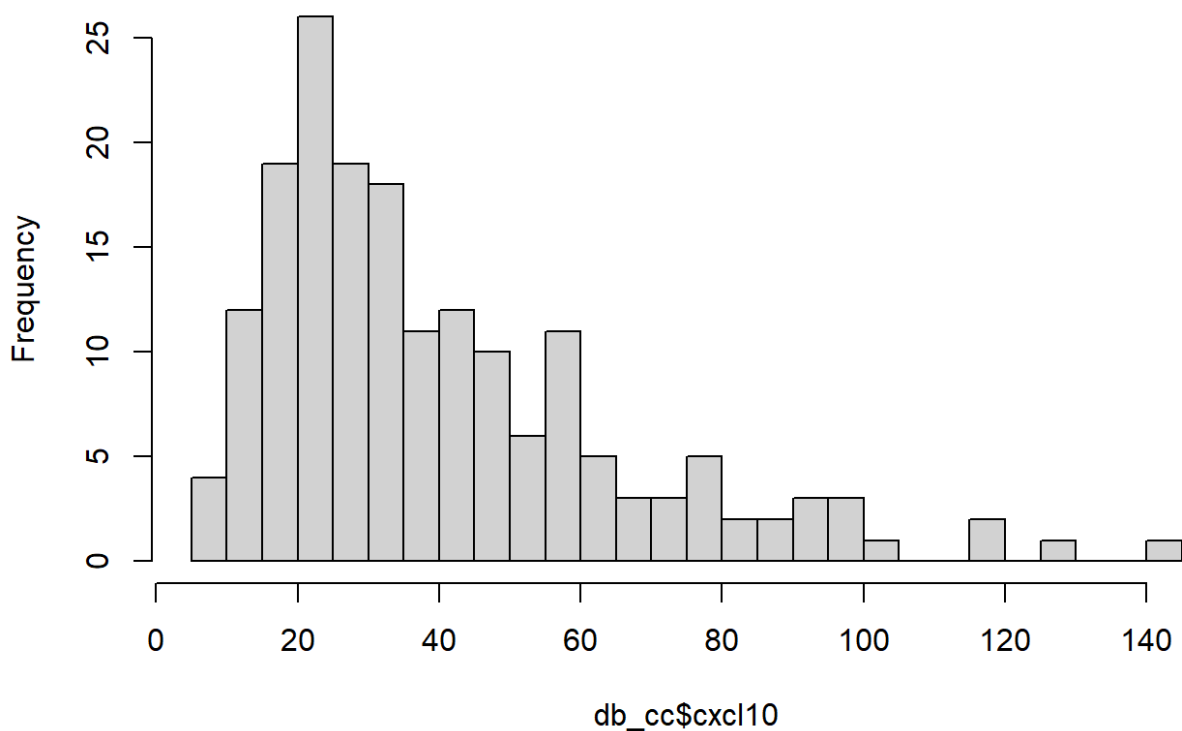
nrow(db); nrow(db_cc)
```

```
## [1] 243
```

```
## [1] 179
```

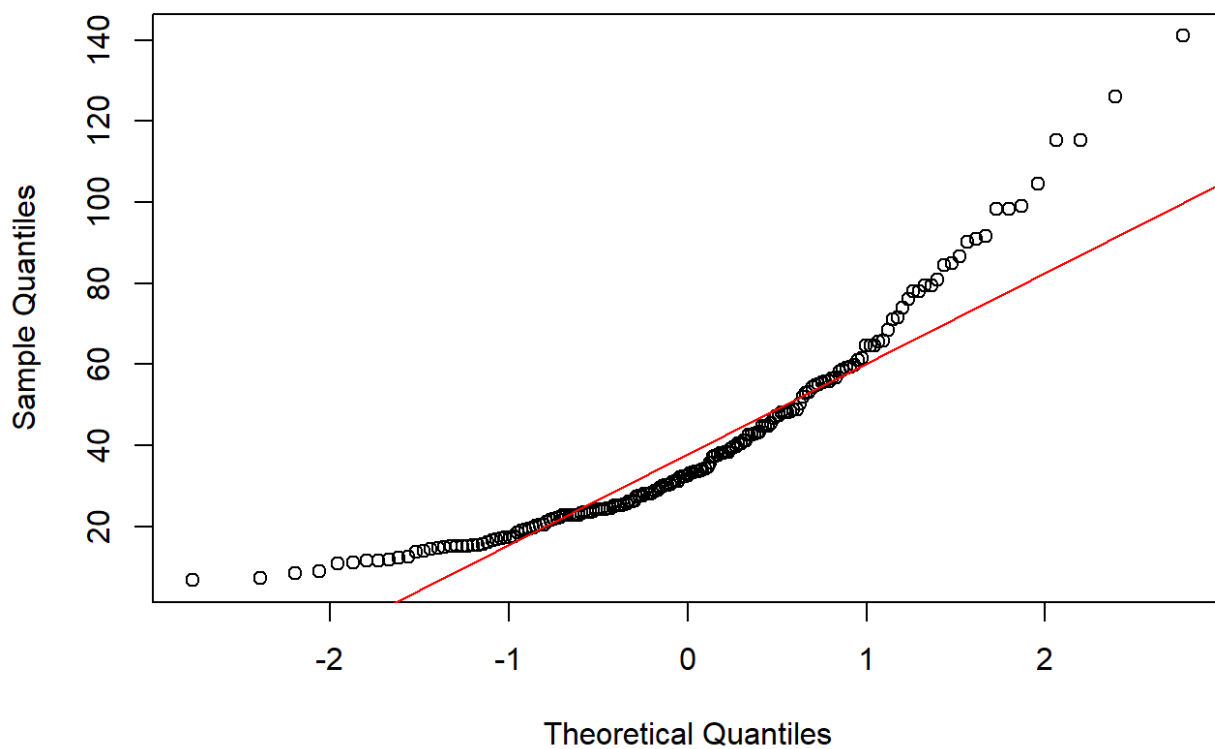
```
hist(db_cc$cxcl10, breaks=30)
```

Histogram of db_cc\$cxcl10



```
qqnorm(db_cc$cxcl10); qqline(db_cc$cxcl10, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$cxc110)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   6.791  22.841  32.572  40.224  53.032 141.017
```

```
min(db_cc$cxc110, na.rm=TRUE)
```

```
## [1] 6.791397
```

```
db$cxcl10_z <- scale(log(db$cxcl10))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$cxcl10_z <- scale(log(db_cc$cxcl10))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

cxcl10Model_hetero <- glmmTMB(
  cxcl10_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(cxcl10Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## cxcl10_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1
| id)
## Dispersion:                ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    483.6    544.1    -222.8    445.6      160
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id       (Intercept) 0.5354   0.7317
##   Residual              NA       NA
## Number of obs: 179, groups: id, 79
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -6.47252    3.08050  -2.101   0.0356 *
## t1            -0.21665    0.23112  -0.937   0.3486
## t2            -0.26631    0.43712  -0.609   0.5424
## GROUP1        -0.34107    0.33864  -1.007   0.3138
## GROUP2        -0.61176    0.30655  -1.996   0.0460 *
## GROUP3        -0.75962    0.39395  -1.928   0.0538 .
## Age           0.08536    0.03982   2.144   0.0321 *
## SexM          -0.40658    0.20014  -2.031   0.0422 *
## CIRS_severity_0 0.21224    0.17805   1.192   0.2333
## t1:GROUP1      0.10372    0.32117   0.323   0.7467
## t2:GROUP1      0.38589    0.42345   0.911   0.3621
## t1:GROUP2      0.40411    0.28528   1.417   0.1566
## t2:GROUP2      0.45676    0.38117   1.198   0.2308
## t1:GROUP3      0.55840    0.36611   1.525   0.1272
## t2:GROUP3      0.61638    0.40651   1.516   0.1295
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.57289    0.16725  -3.425 0.000614 ***
## t1            0.03974    0.25677   0.155 0.877004
## t2            0.22465    0.22029   1.020 0.307827
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

cxcl10Model_homo <- glmmTMB(
  cxcl10_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(cxcl10Model_homo, cxcl10Model_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## cxcl10Model_homo: cxcl10_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## cxcl10Model_homo:      (1 | id), zi=~0, disp=~t
## cxcl10Model_hetero: cxcl10_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## cxcl10Model_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>C
hisq)
## cxcl10Model_homo    17 480.95 535.14 -223.47   446.95
## cxcl10Model_hetero  19 483.56 544.12 -222.78   445.56 1.3885      2
0.4995

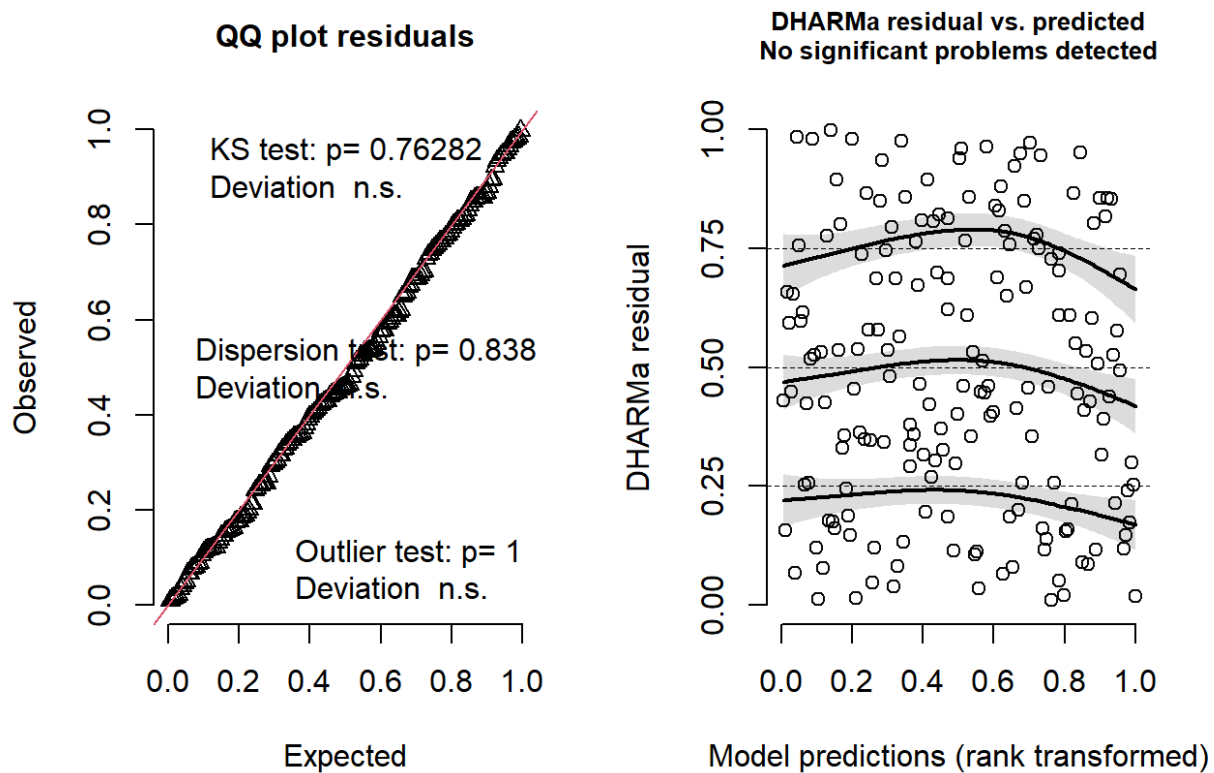
```

```

sim_res <- simulateResiduals(cxcl10Model_hetero, n = 1000)
plot(sim_res)

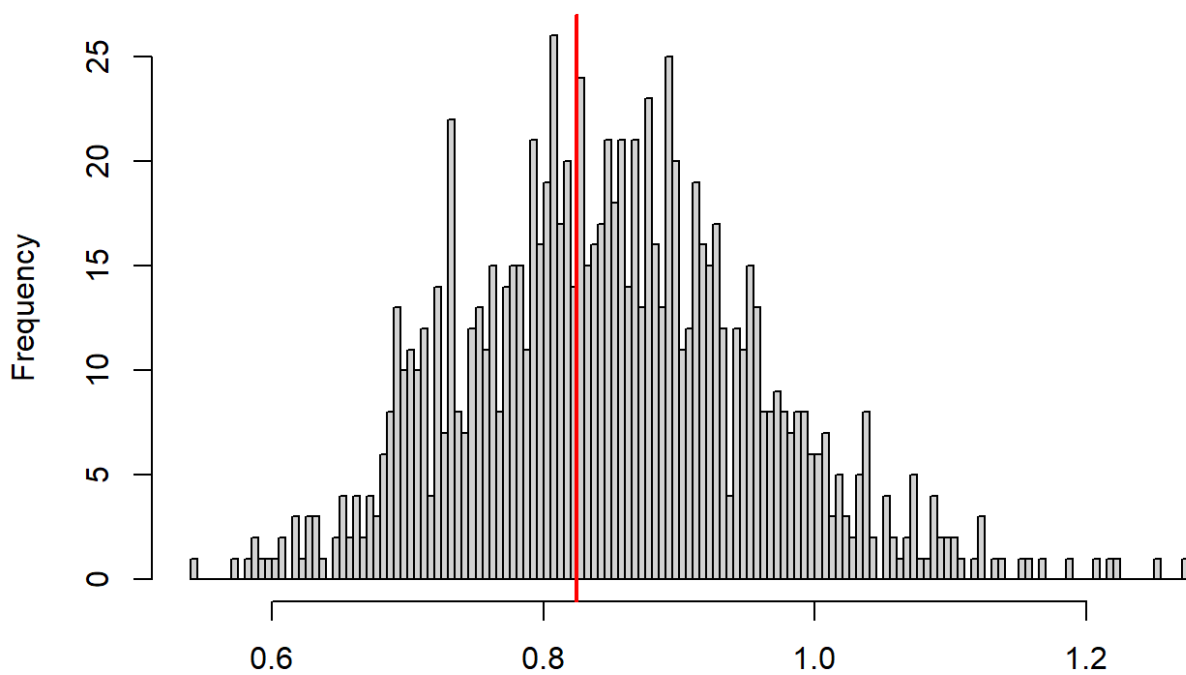
```

DHARMA residual



```
testDispersion(sim_res)
```

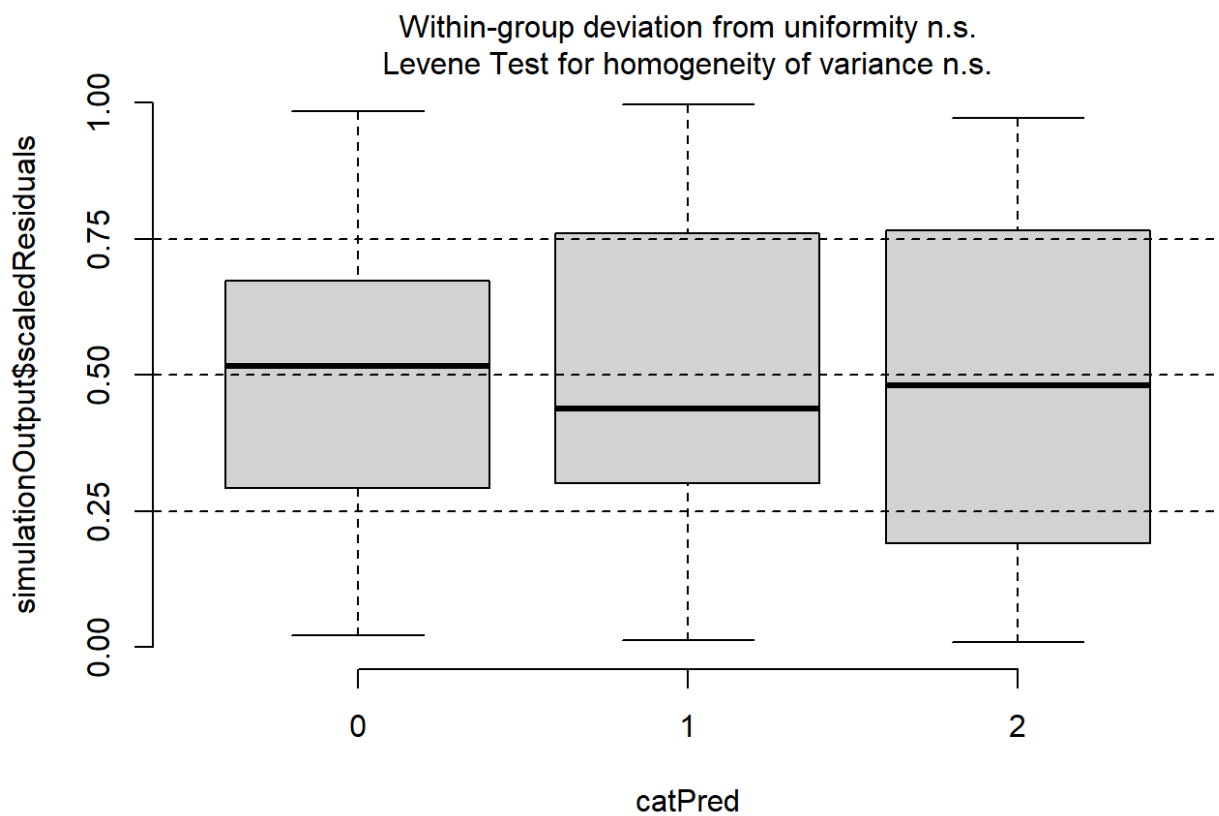
DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



Simulated values, red line = fitted model. p -value (two.sided) = 0.838


```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 0.96976, p-value = 0.838
## alternative hypothesis: two.sided
```

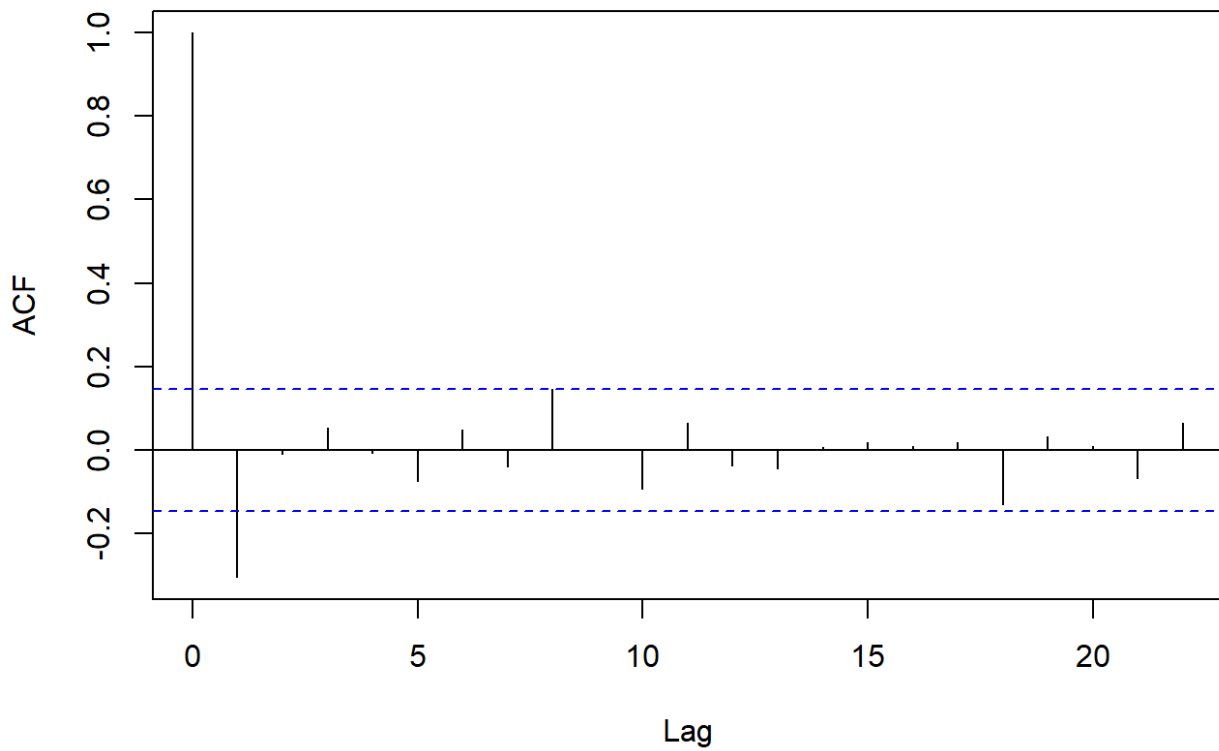
```
testQuantiles(sim_res, db_cc$t)
```



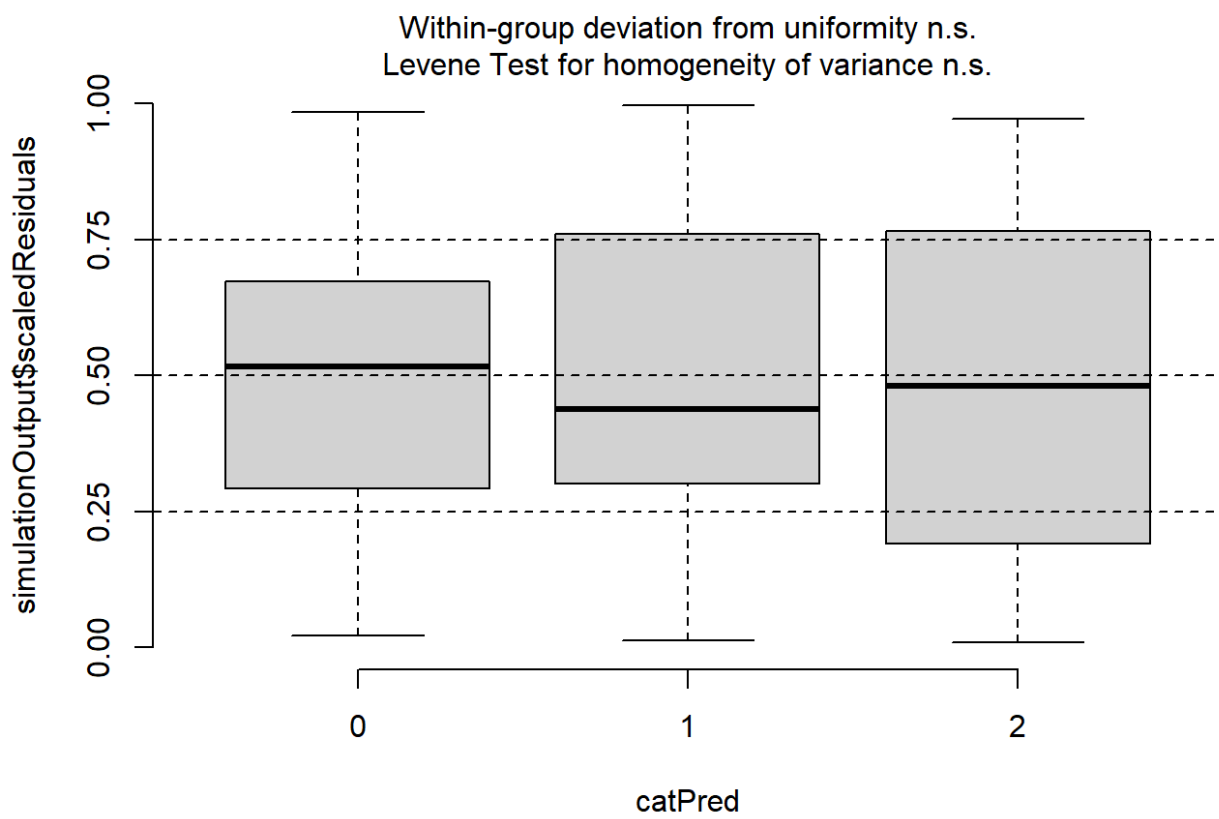
```
## NULL
```

```
acf(residuals(cxcl10Model_hetero))
```

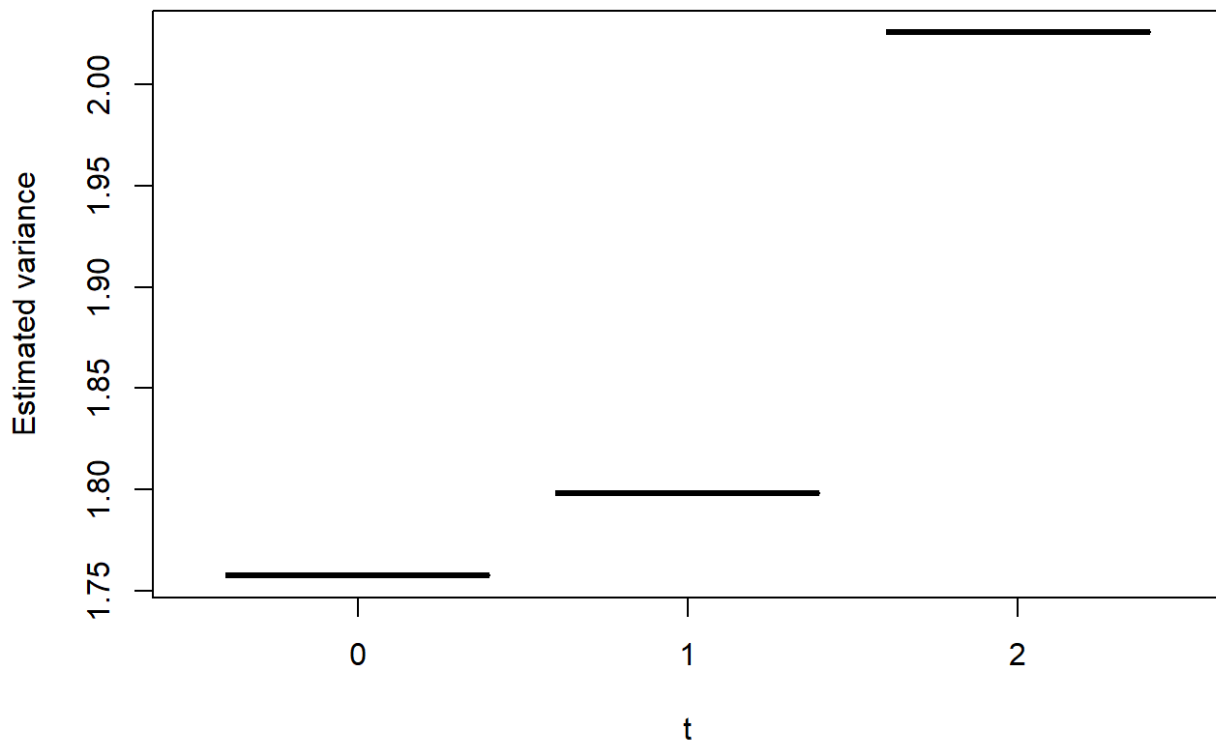
Series residuals(cxcl10Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

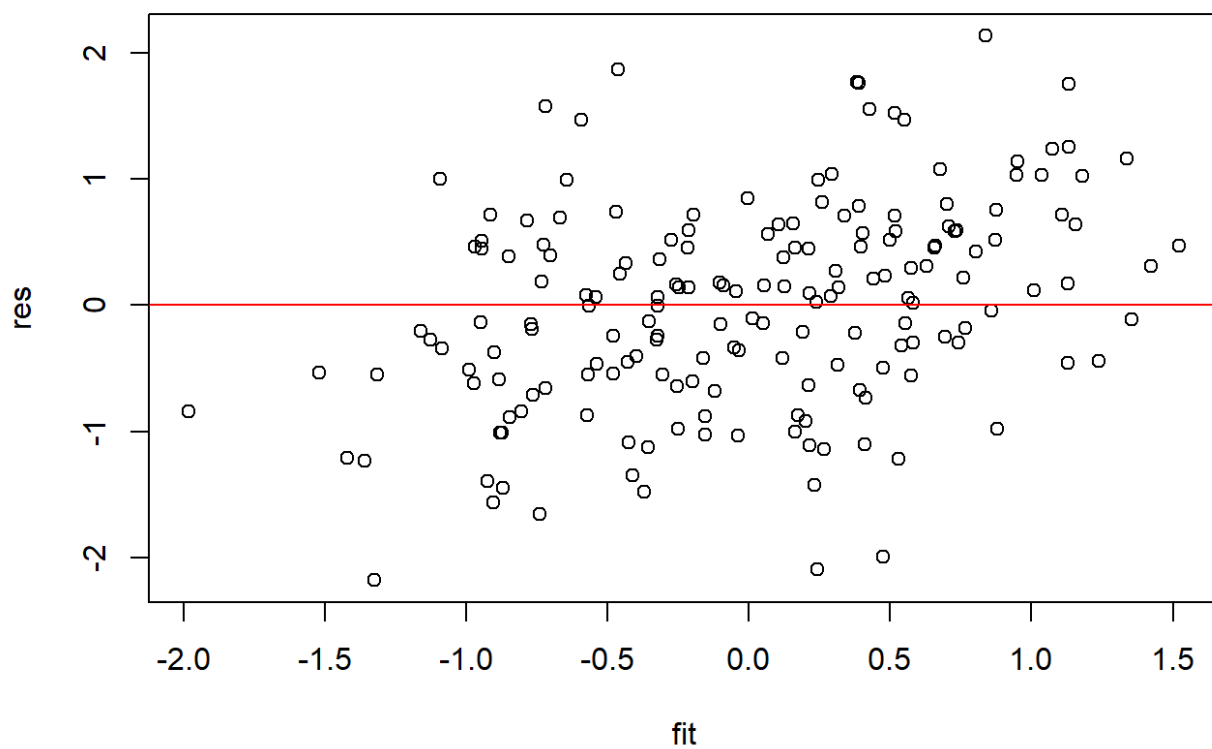


```
pred_disp <- predict(cxcl10Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(cxcl10Model_hetero, type="pearson")
fit <- fitted(cxcl10Model_hetero)

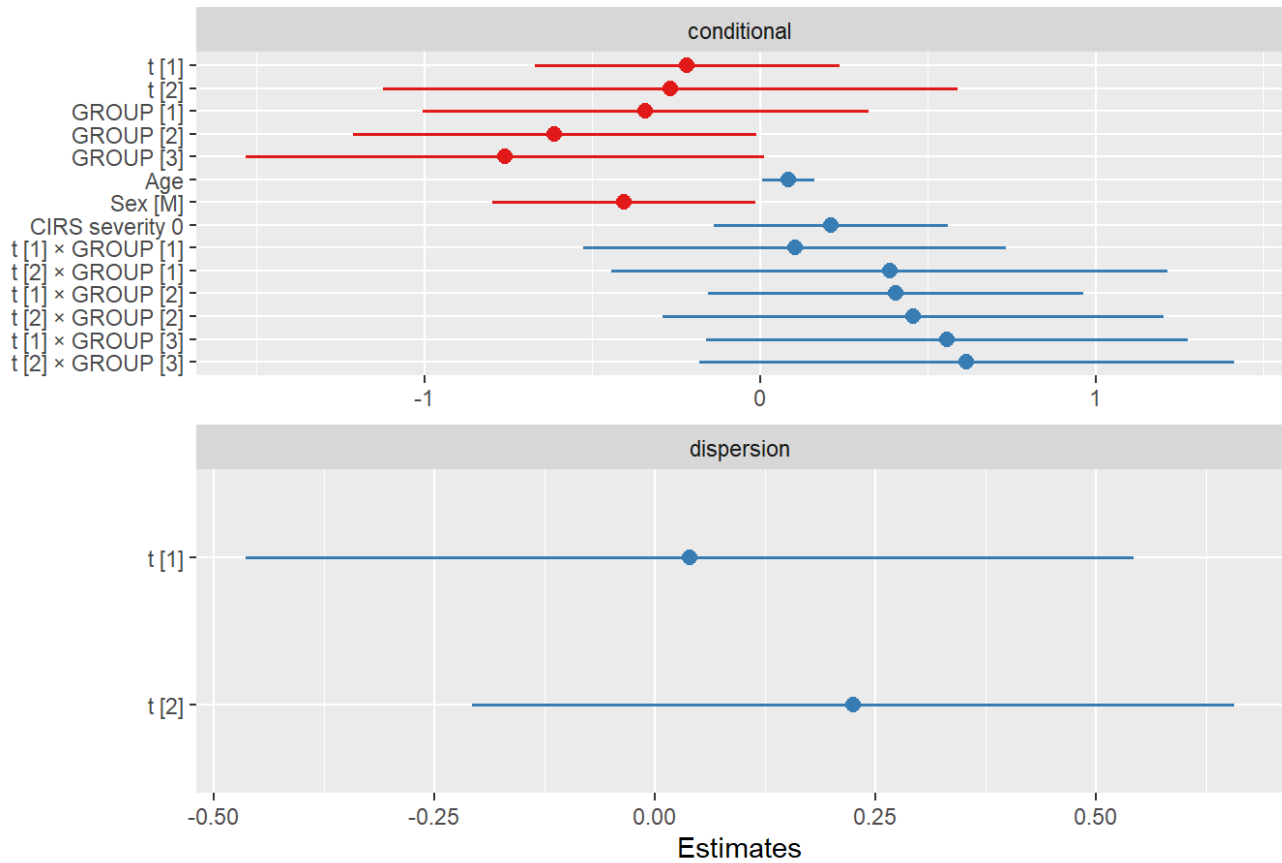
plot(fit, res)
abline(h=0, col="red")
```



7.1 Visualizzazione del modello

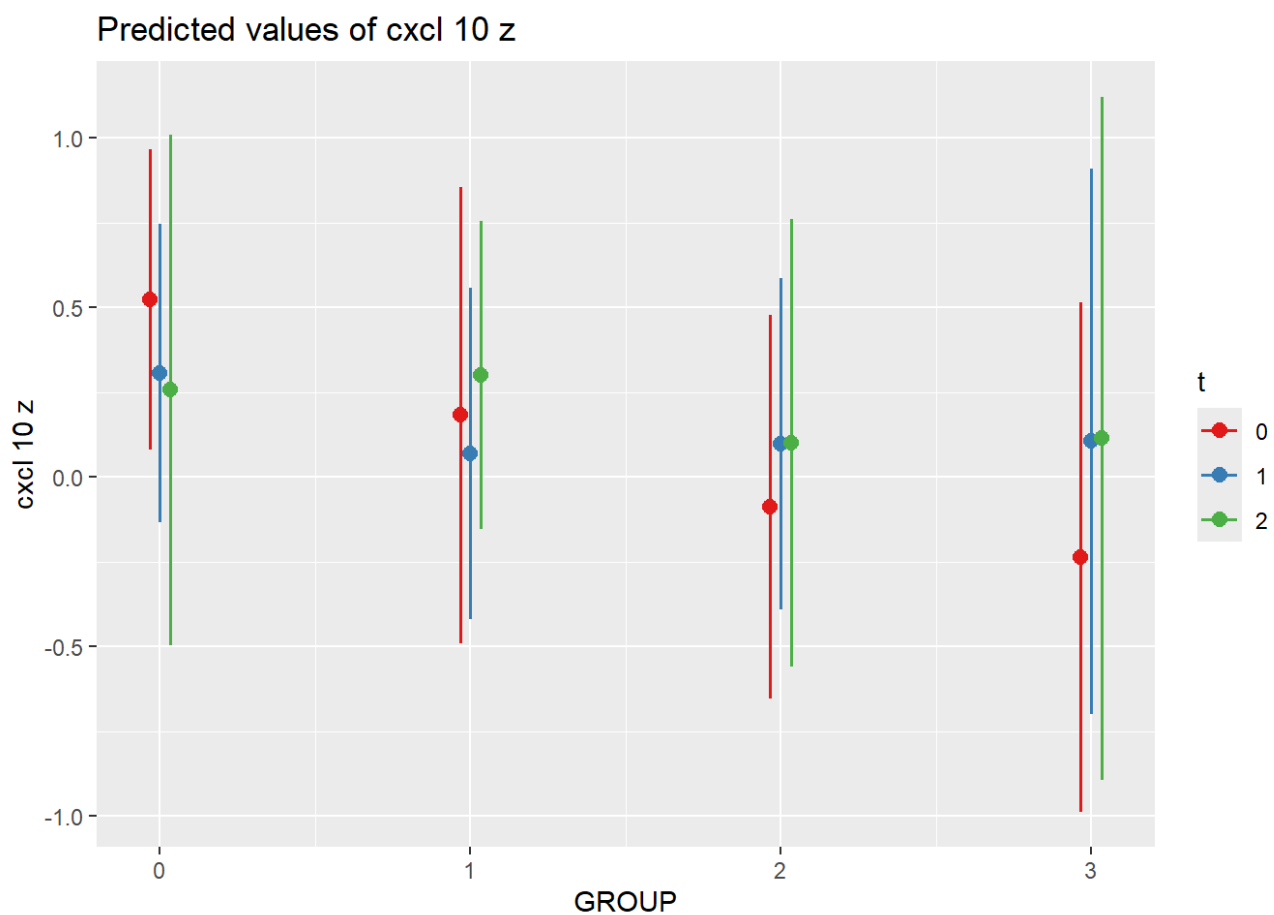
```
plot_model(cxcl10Model_hetero)
```

cxcl 10 z



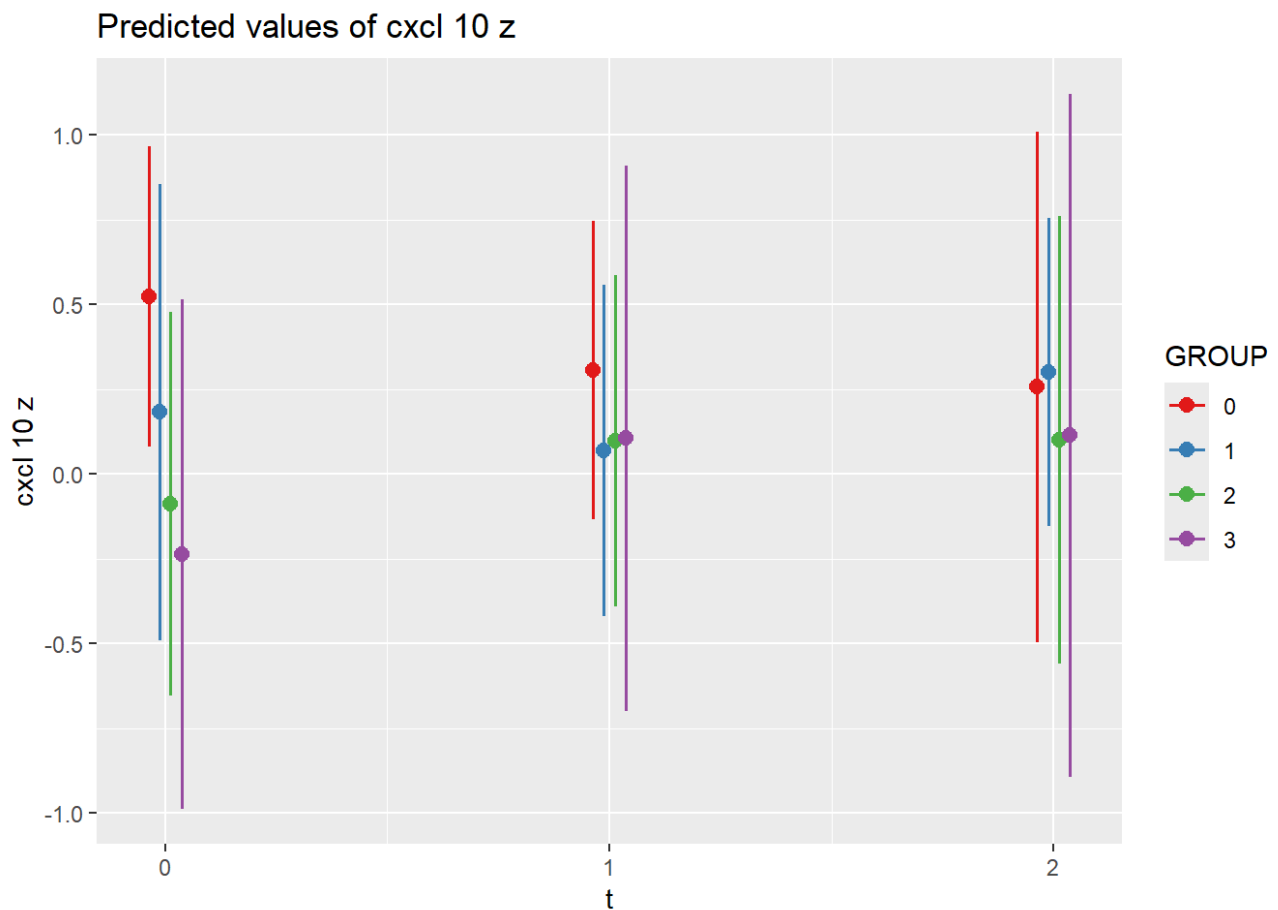
```
plot_model(cxcl10Model_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```



```
plot_model(cxcl10Model_hetero, type = "pred",  
           terms = c("t", "GROUP"),  
           pred.type = "fe")
```

```
## Ignoring unknown labels:  
## • linetype : "GROUP"  
## • shape : "GROUP"
```

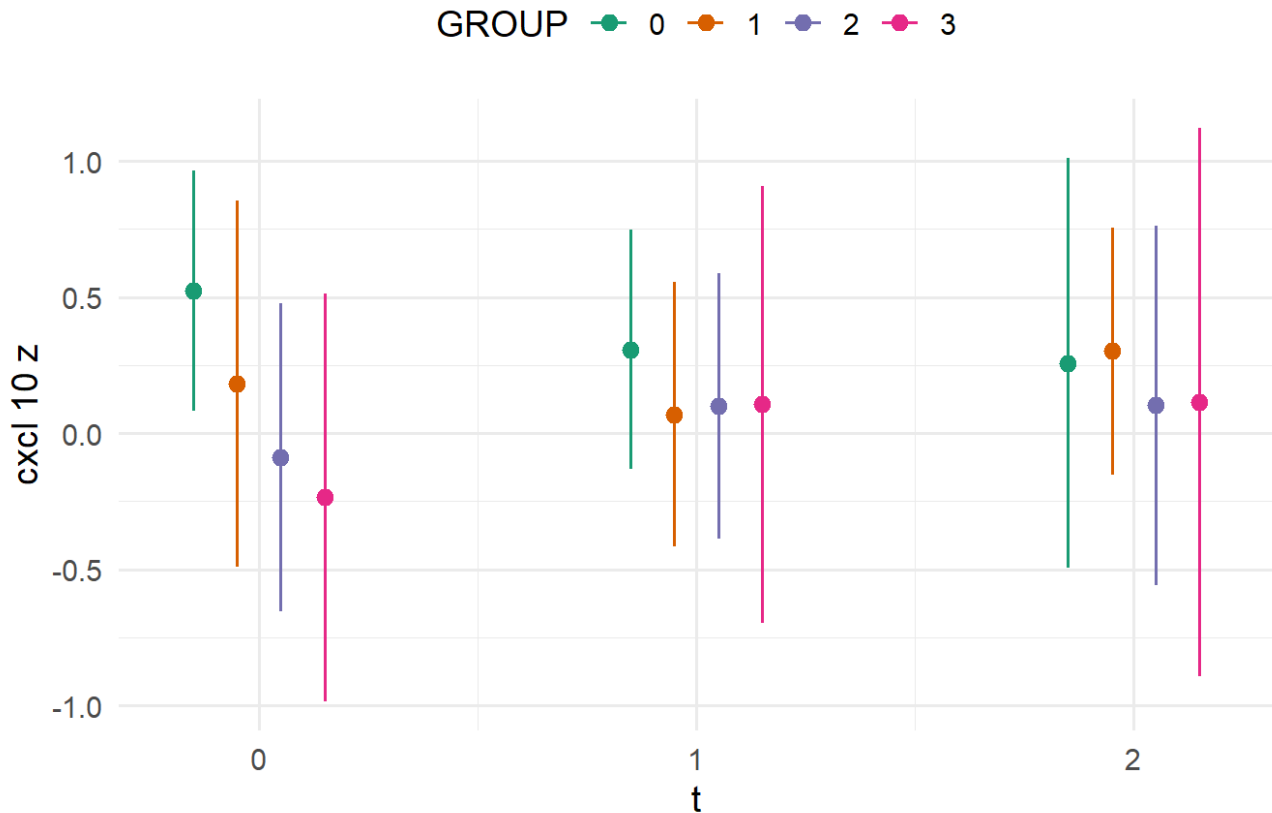


```
plot_model(cxcl10Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

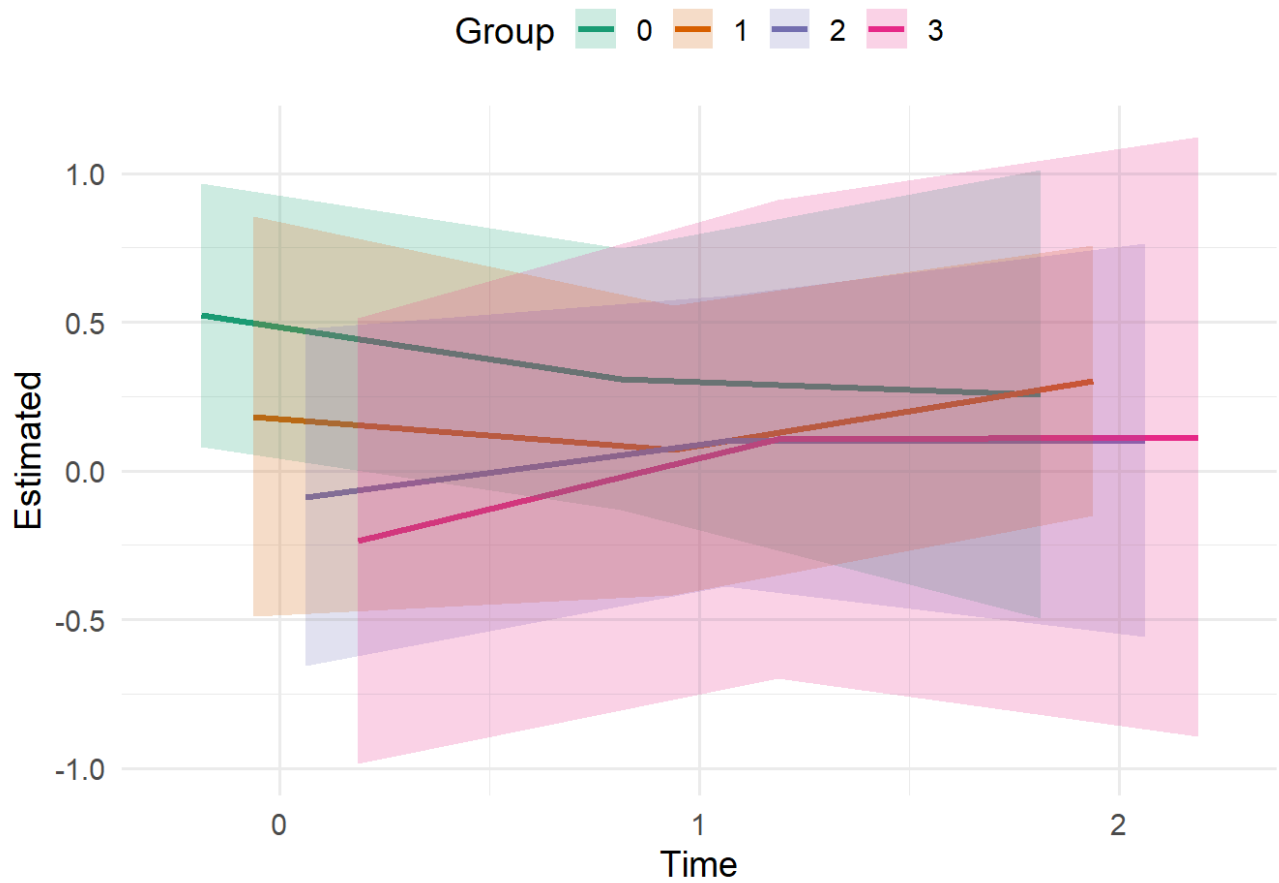
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of cxcl 10 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(cxcl10Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```

7.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(cxcl10Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##   GROUP   emmean    SE  df asymp.LCL asymp.UCL
##   0       0.3205 0.204 Inf   -0.0792    0.720
##   1      -0.0206 0.345 Inf   -0.6971    0.656
##   2      -0.2912 0.290 Inf   -0.8594    0.277
##   3      -0.4391 0.359 Inf   -1.1433    0.265
##
## t = 1:
##   GROUP   emmean    SE  df asymp.LCL asymp.UCL
##   0       0.1039 0.198 Inf   -0.2842    0.492
##   1      -0.1335 0.244 Inf   -0.6120    0.345
##   2      -0.1038 0.245 Inf   -0.5842    0.377
##   3      -0.0974 0.386 Inf   -0.8544    0.660
##
## t = 2:
##   GROUP   emmean    SE  df asymp.LCL asymp.UCL
##   0       0.0542 0.365 Inf   -0.6608    0.769
##   1       0.0990 0.223 Inf   -0.3377    0.536
##   2      -0.1008 0.327 Inf   -0.7426    0.541
##   3      -0.0890 0.492 Inf   -1.0526    0.875
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(cxcl10Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##   contrast          estimate    SE  df z.ratio p.value
##   GROUP0 - GROUP1    0.1779 0.298 Inf    0.597  0.9330
##   GROUP0 - GROUP2    0.3248 0.272 Inf    1.194  0.6304
##   GROUP0 - GROUP3    0.3680 0.353 Inf    1.044  0.7237
##   GROUP1 - GROUP2    0.1469 0.292 Inf    0.504  0.9581
##   GROUP1 - GROUP3    0.1902 0.420 Inf    0.453  0.9690
##   GROUP2 - GROUP3    0.0432 0.392 Inf    0.110  0.9995
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(cxcl10Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.34107 0.339 Inf   1.007  0.7452
## GROUP0 - GROUP2  0.61176 0.307 Inf   1.996  0.1896
## GROUP0 - GROUP3  0.75962 0.394 Inf   1.928  0.2162
## GROUP1 - GROUP2  0.27069 0.366 Inf   0.741  0.8807
## GROUP1 - GROUP3  0.41854 0.470 Inf   0.891  0.8094
## GROUP2 - GROUP3  0.14785 0.441 Inf   0.335  0.9870
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.23735 0.335 Inf   0.708  0.8941
## GROUP0 - GROUP2  0.20765 0.305 Inf   0.682  0.9041
## GROUP0 - GROUP3  0.20122 0.417 Inf   0.483  0.9629
## GROUP1 - GROUP2 -0.02970 0.343 Inf  -0.087  0.9998
## GROUP1 - GROUP3 -0.03614 0.484 Inf  -0.075  0.9999
## GROUP2 - GROUP3 -0.00643 0.454 Inf  -0.014  1.0000
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.04482 0.427 Inf  -0.105  0.9996
## GROUP0 - GROUP2  0.15500 0.397 Inf   0.390  0.9798
## GROUP0 - GROUP3  0.14323 0.452 Inf   0.317  0.9890
## GROUP1 - GROUP2  0.19982 0.387 Inf   0.516  0.9553
## GROUP1 - GROUP3  0.18805 0.540 Inf   0.348  0.9855
## GROUP2 - GROUP3 -0.01177 0.510 Inf  -0.023  1.0000
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(cxcl10Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    0.21665 0.231 Inf   0.937  0.6166
## t0 - t2    0.26631 0.437 Inf   0.609  0.8151
## t1 - t2    0.04966 0.355 Inf   0.140  0.9893
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1    0.11293 0.281 Inf   0.401  0.9150
## t0 - t2   -0.11959 0.374 Inf  -0.320  0.9452
## t1 - t2   -0.23252 0.260 Inf  -0.895  0.6433
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1   -0.18746 0.280 Inf  -0.670  0.7810
## t0 - t2   -0.19046 0.426 Inf  -0.447  0.8958
## t1 - t2   -0.00300 0.331 Inf  -0.009  1.0000
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1   -0.34175 0.364 Inf  -0.940  0.6151
## t0 - t2   -0.35008 0.507 Inf  -0.691  0.7688
## t1 - t2   -0.00833 0.452 Inf  -0.018  0.9998
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

8 Model cxcl16

```
bptest(lm(cxcl16 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(cxcl16 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 18.69, df = 14, p-value = 0.1771
```

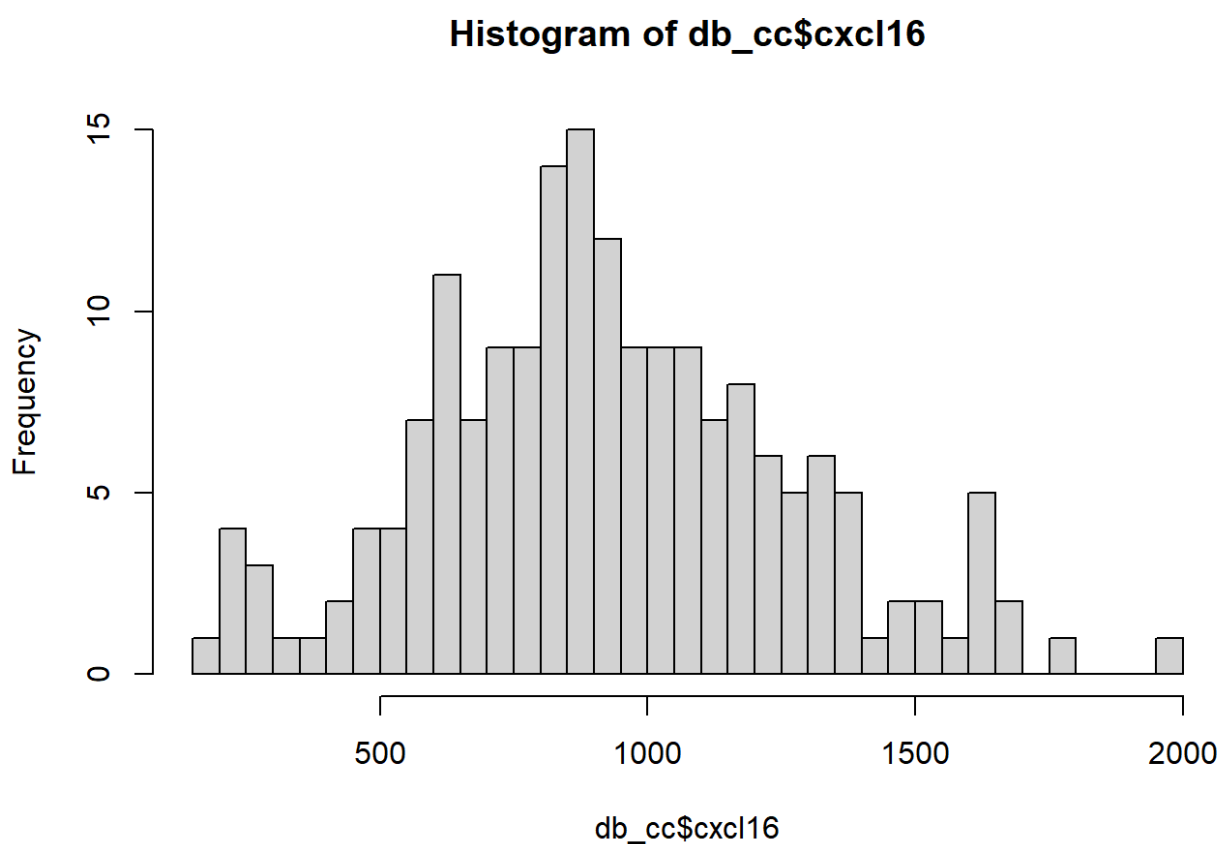
```
vars <- c("cxcl16", "t", "GROUP", "Age", "Sex", "CIRS_severity_0", "id")
db_cc <- db[complete.cases(db[, vars]), vars]

nrow(db); nrow(db_cc)
```

```
## [1] 243
```

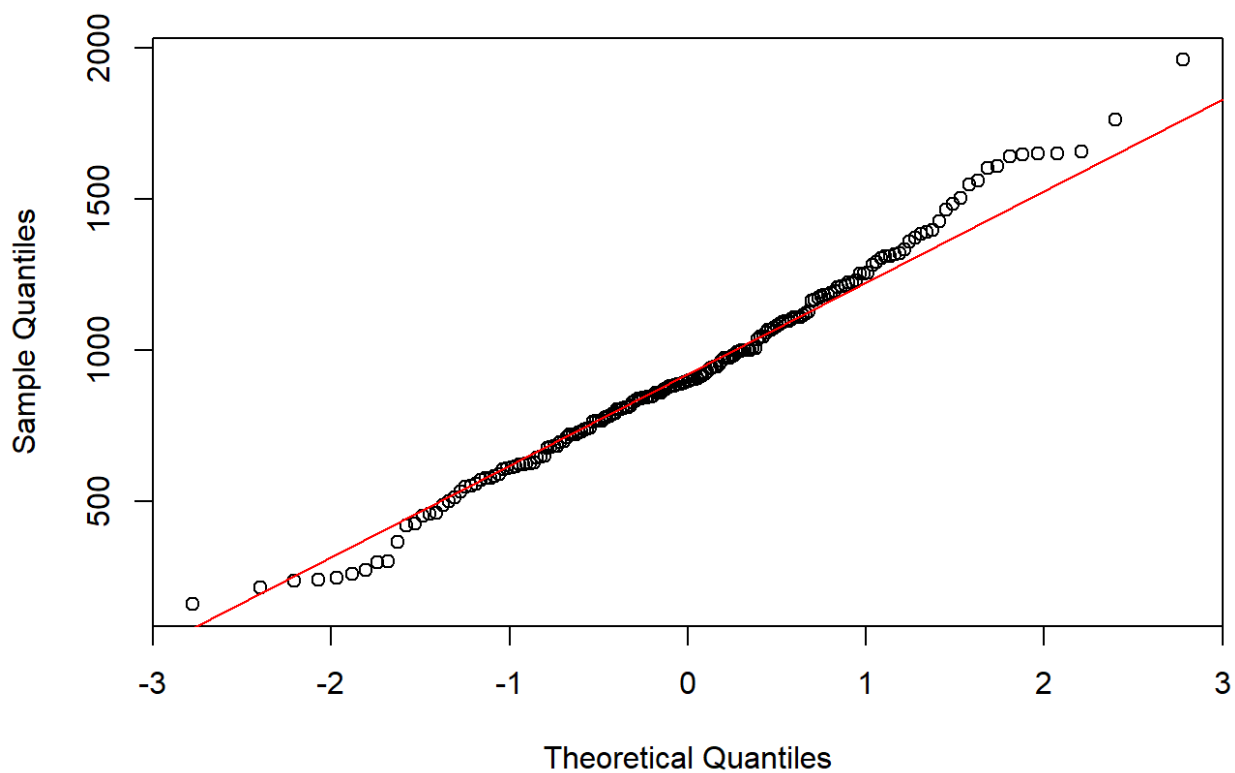
```
## [1] 183
```

```
hist(db_cc$cxcl16, breaks=30)
```



```
qqnorm(db_cc$cxcl16); qqline(db_cc$cxcl16, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$cxc116)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##   161.0   717.8   898.4   929.1  1125.9  1960.8
```

```
min(db_cc$cxc116, na.rm=TRUE)
```

```
## [1] 161.0224
```

```
db$cxcl16_z <- scale(log(db$cxcl16))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$cxcl16_z <- scale(log(db_cc$cxcl16))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

cxcl16Model_hetero <- glmmTMB(
  cxcl16_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(cxcl16Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## cxcl16_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1
| id)
## Dispersion:                ~t
## Data: db_cc
##
##           AIC           BIC      logLik -2*log(L)  df.resid
##        490.7        551.7      -226.4     452.7      164
##
## Random effects:
##
## Conditional model:
##   Groups   Name                Variance Std.Dev.
##   id       (Intercept) 0.4355    0.66
##   Residual                    NA        NA
## Number of obs: 183, groups: id, 80
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.29529    2.92093  -0.443 0.657440
## t1              0.15762    0.24103   0.654 0.513141
## t2              0.33931    0.42441   0.799 0.424007
## GROUP1          0.79269    0.35953   2.205 0.027468 *
## GROUP2          1.06452    0.31842   3.343 0.000828 ***
## GROUP3          0.93329    0.41532   2.247 0.024630 *
## Age             0.01135    0.03760   0.302 0.762697
## SexM            0.08454    0.18804   0.450 0.652987
## CIRS_severity_0 0.11626    0.17526   0.663 0.507124
## t1:GROUP1       -0.96663    0.35260  -2.741 0.006117 **
## t2:GROUP1       -1.66162    0.44154  -3.763 0.000168 ***
## t1:GROUP2       -0.91237    0.30571  -2.984 0.002841 **
## t2:GROUP2       -1.06681    0.38590  -2.764 0.005701 **
## t1:GROUP3       -0.67562    0.39849  -1.695 0.089990 .
## t2:GROUP3       -1.07285    0.44213  -2.427 0.015244 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.34483    0.16254  -2.122  0.0339 *
## t1            -0.26313    0.32875  -0.800  0.4235
## t2            -0.02688    0.27021  -0.100  0.9207
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```



```

cxcl16Model_homo <- glmmTMB(
  cxcl16_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(cxcl16Model_homo, cxcl16Model_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## cxcl16Model_homo: cxcl16_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## cxcl16Model_homo:      (1 | id), zi=~0, disp=~t
## cxcl16Model_hetero: cxcl16_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## cxcl16Model_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Ch
isq)
## cxcl16Model_homo    17 488.64 543.2 -227.32   454.64
## cxcl16Model_hetero  19 490.72 551.7 -226.36   452.72 1.9185      2    0.
3832

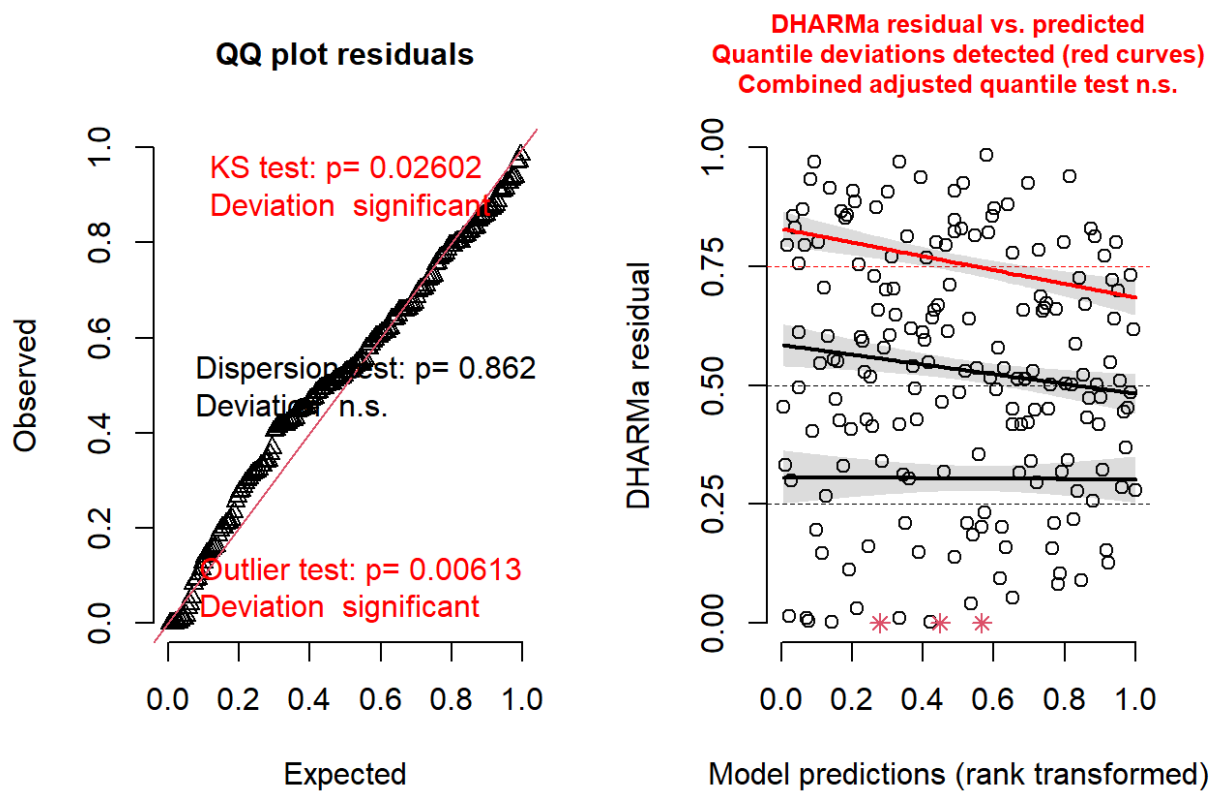
```

```

sim_res <- simulateResiduals(cxcl16Model_hetero, n = 1000)
plot(sim_res)

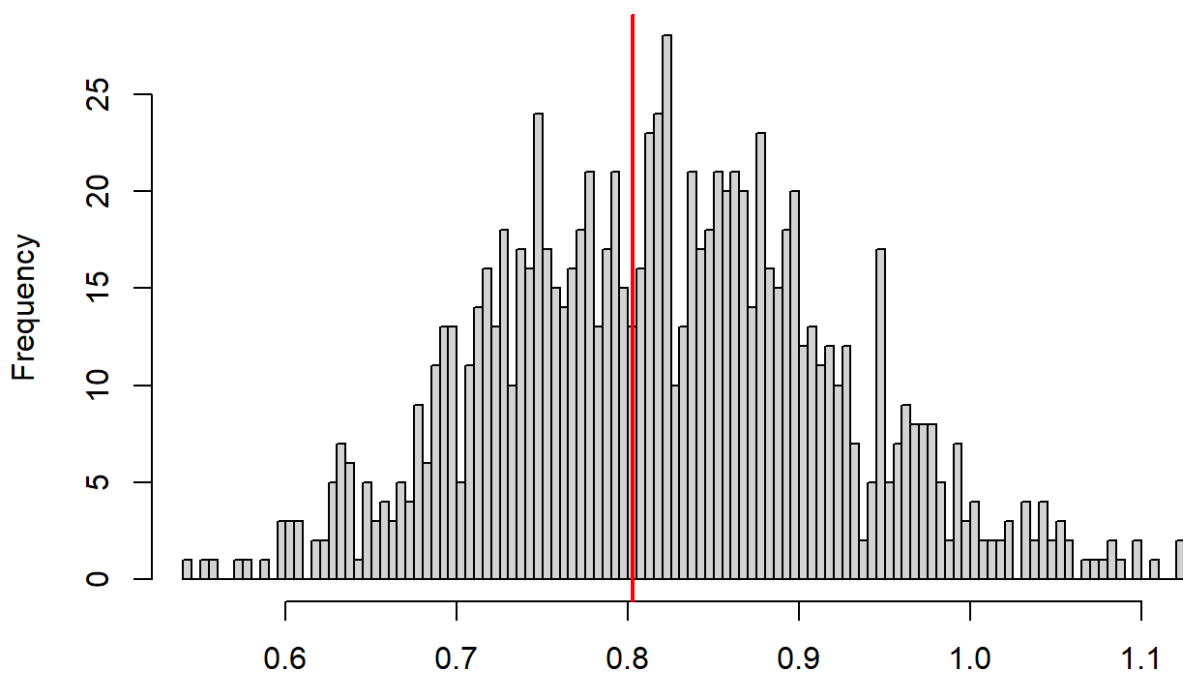
```

DHARMA residual



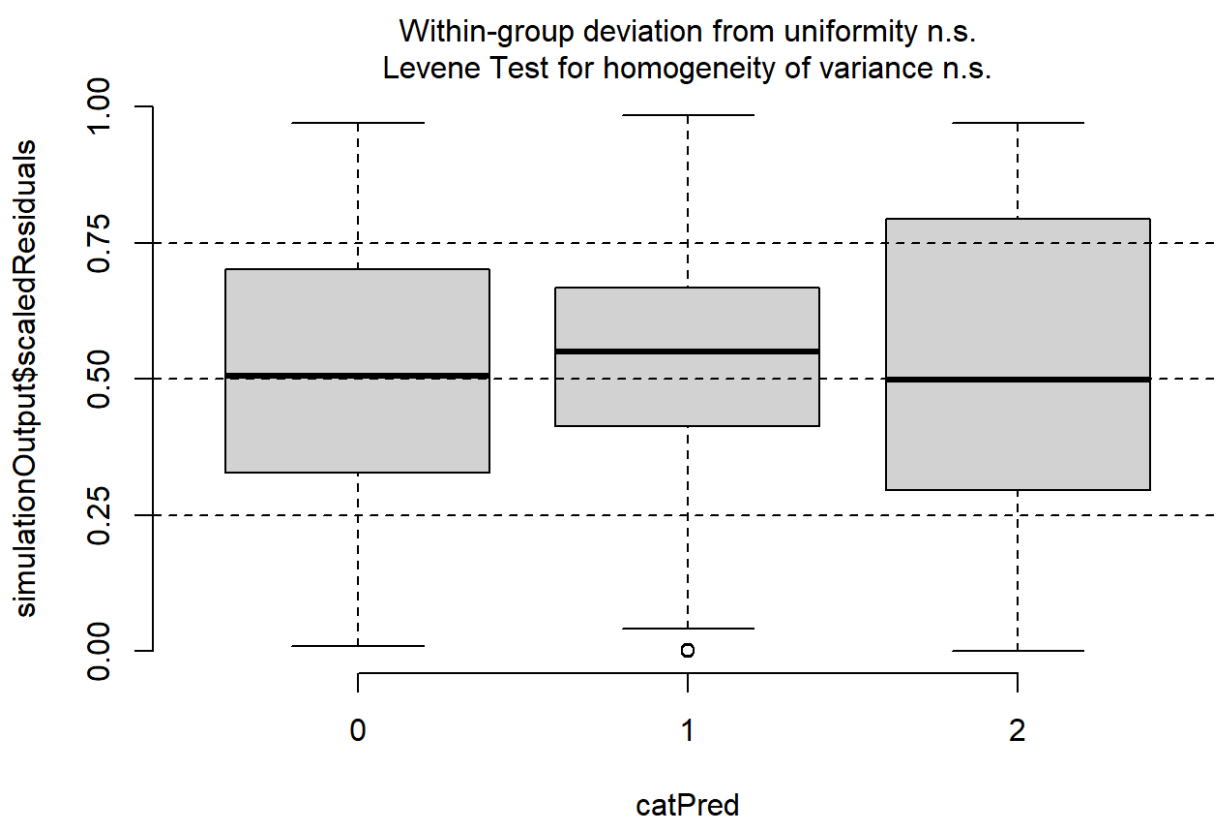
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 0.97736, p-value = 0.862
## alternative hypothesis: two.sided
```

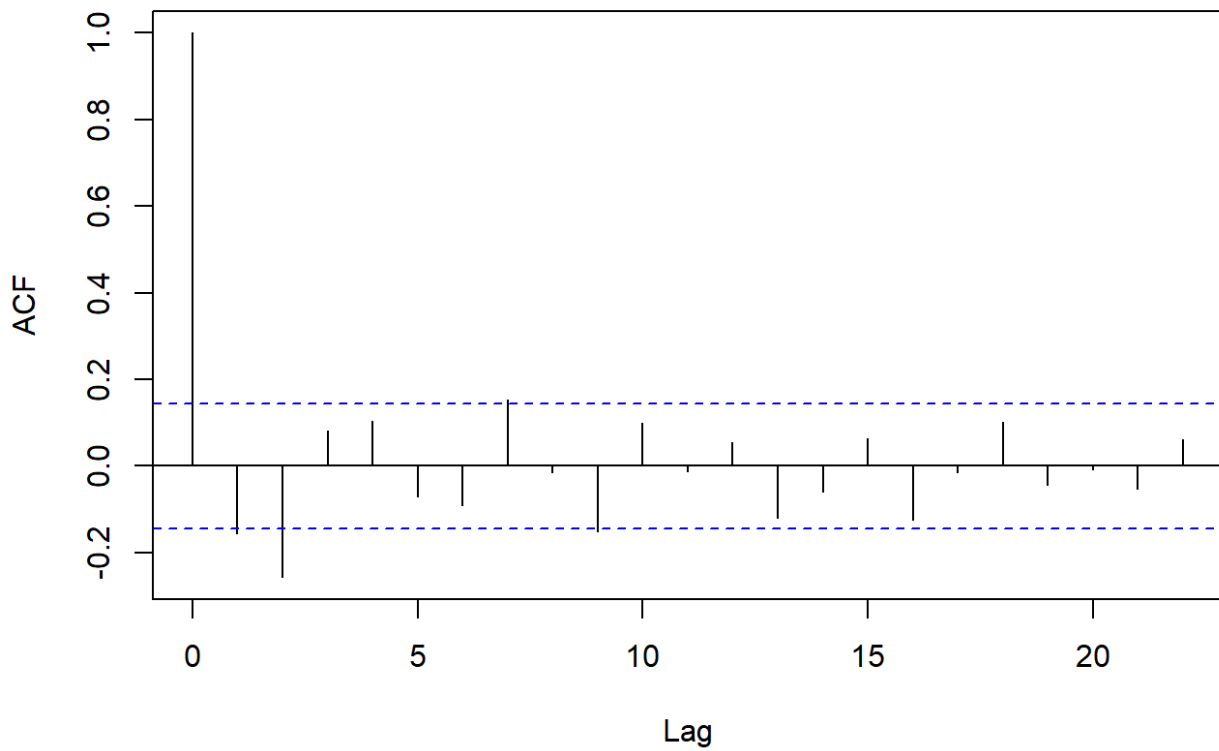
```
testQuantiles(sim_res, db_cc$t)
```



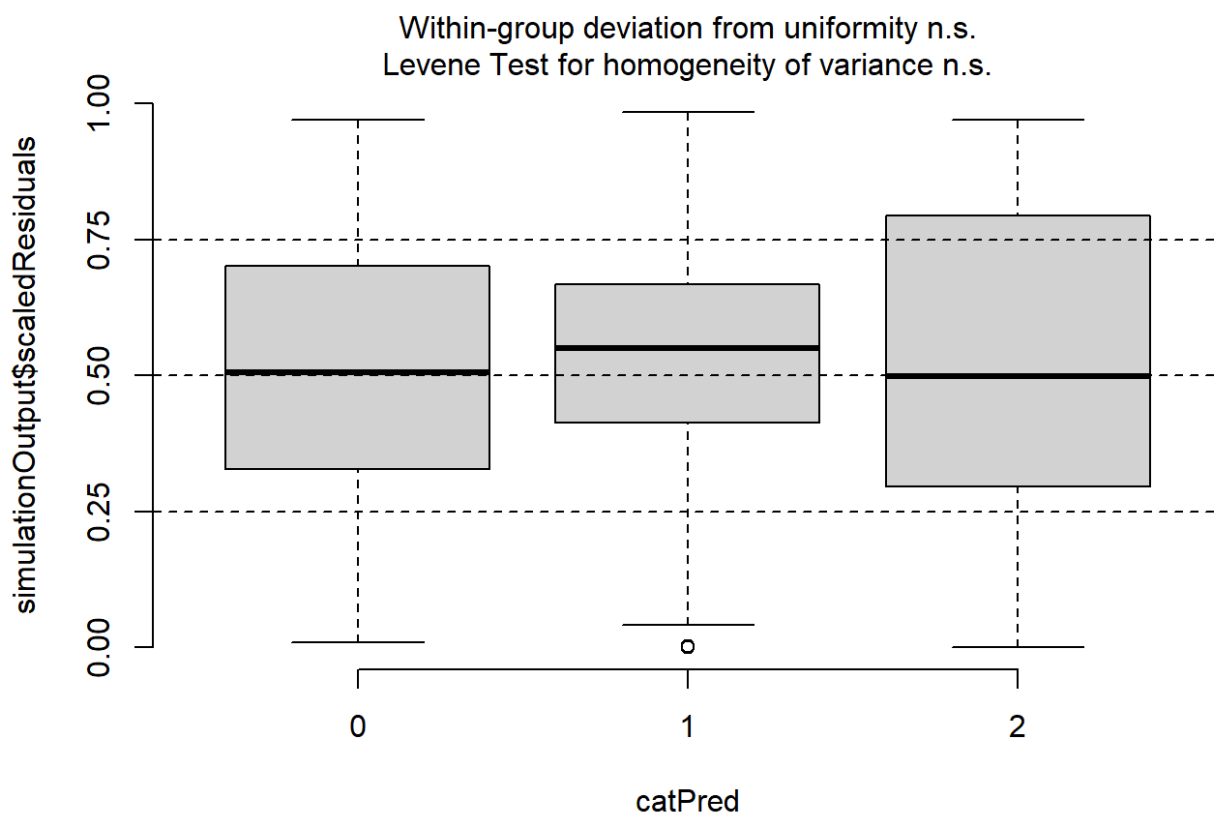
```
## NULL
```

```
acf(residuals(cxcl16Model_hetero))
```

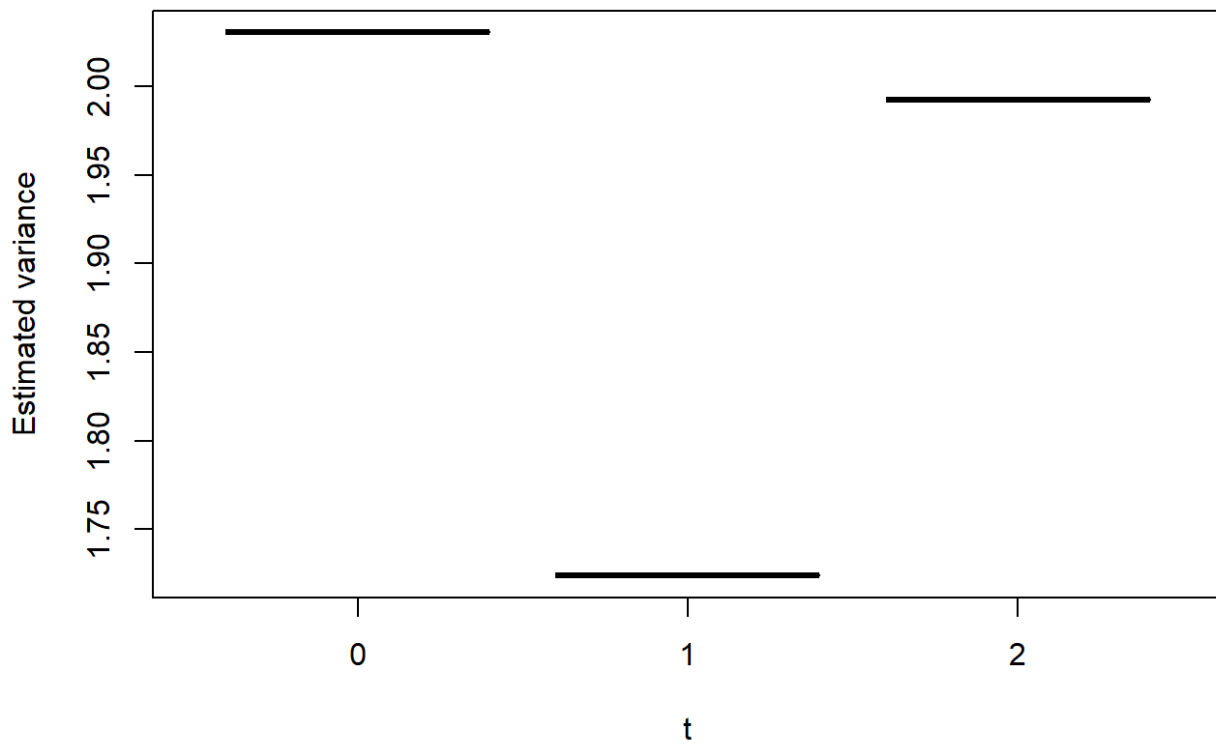
Series residuals(cxcl16Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

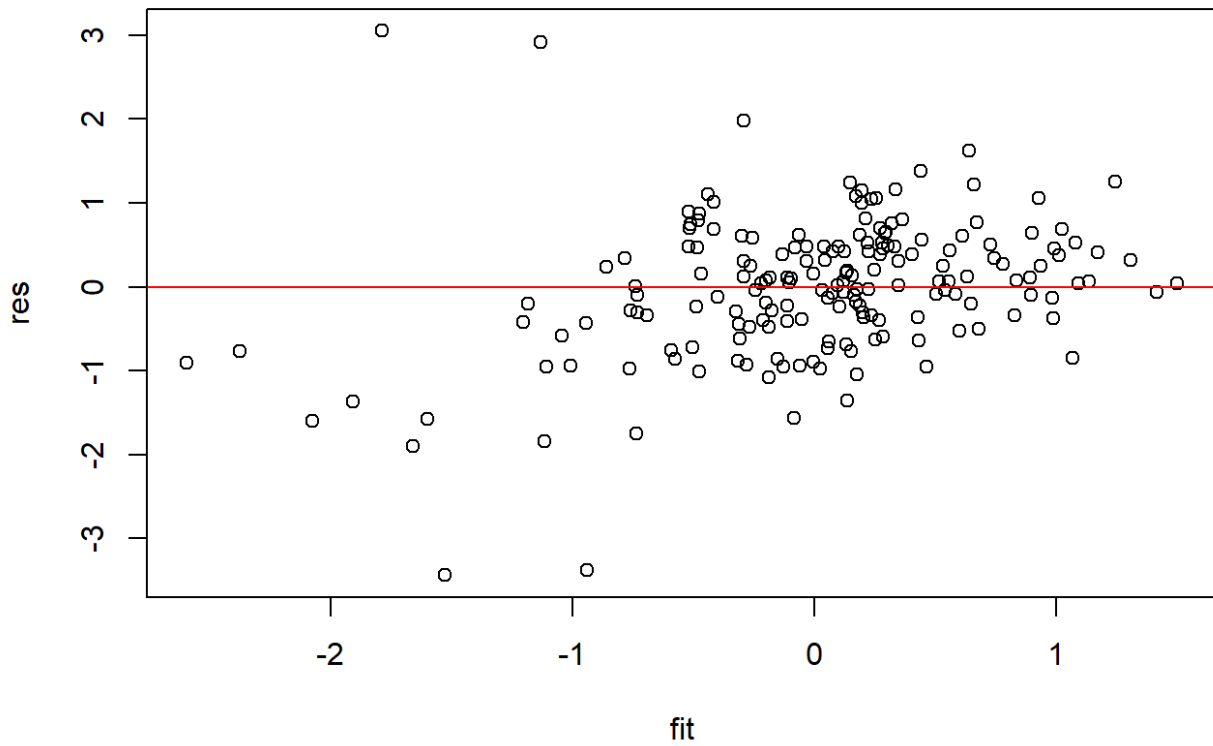


```
pred_disp <- predict(cxcl16Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(cxcl16Model_hetero, type="pearson")
fit <- fitted(cxcl16Model_hetero)

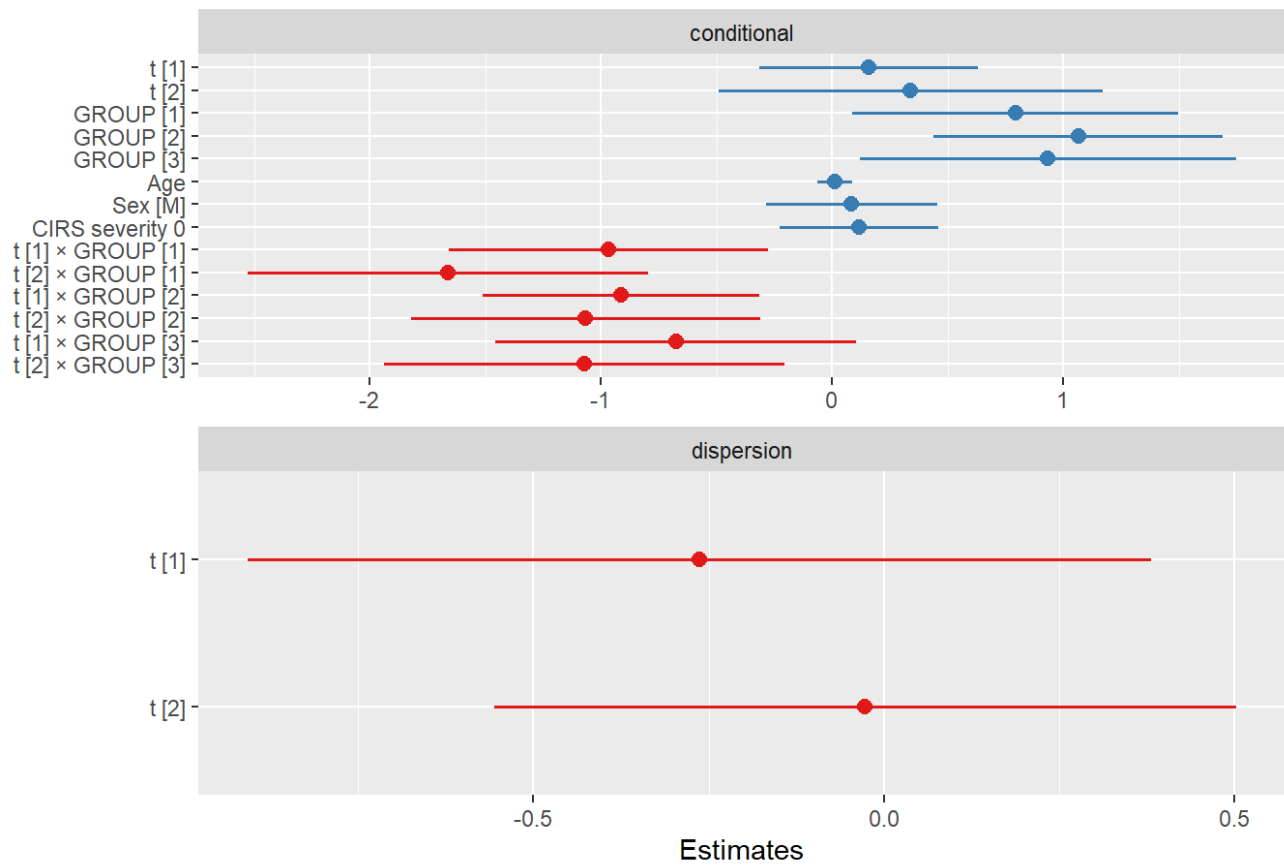
plot(fit, res)
abline(h=0, col="red")
```



8.1 Visualizzazione del modello

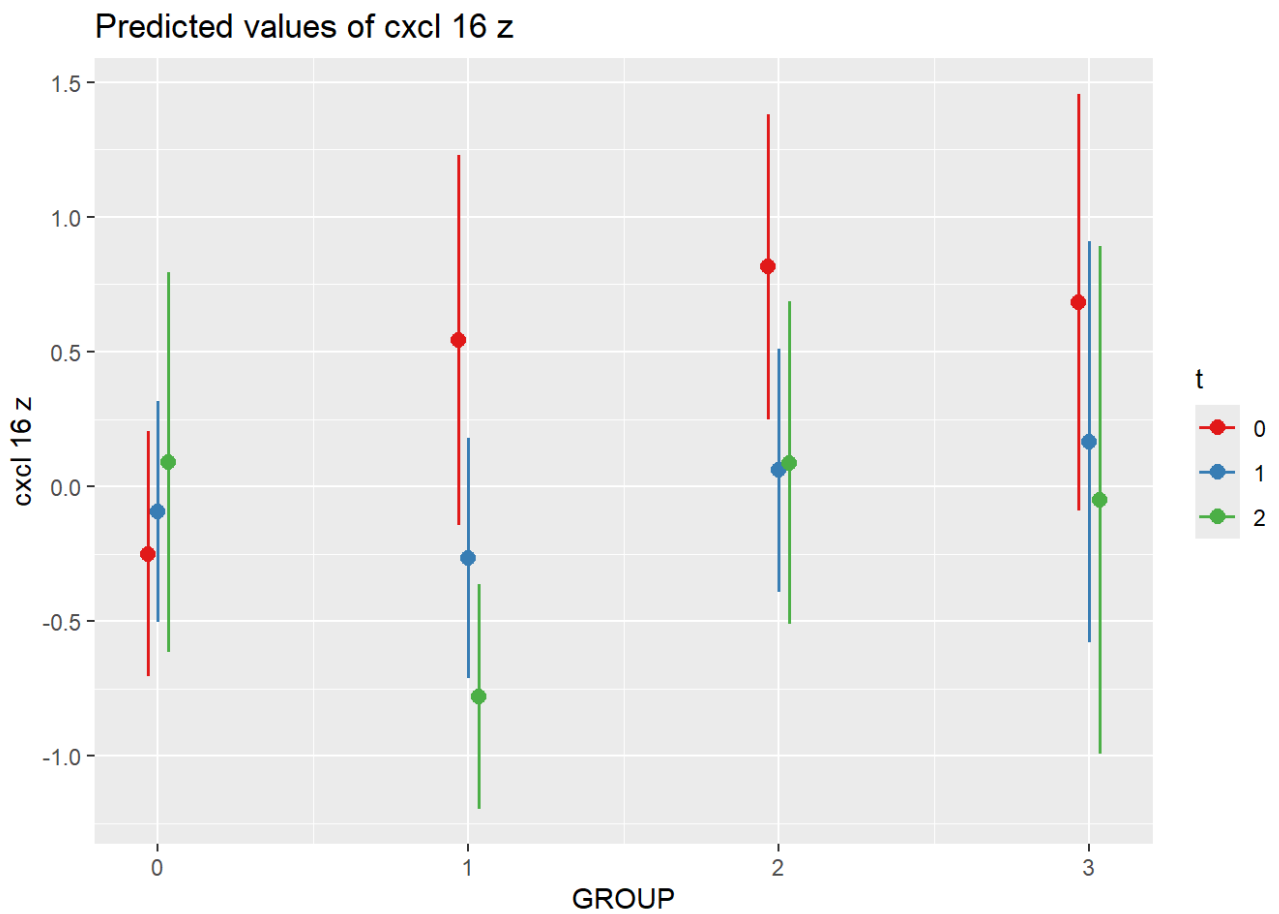
```
plot_model(cxcl16Model_hetero)
```

cxcl 16 z



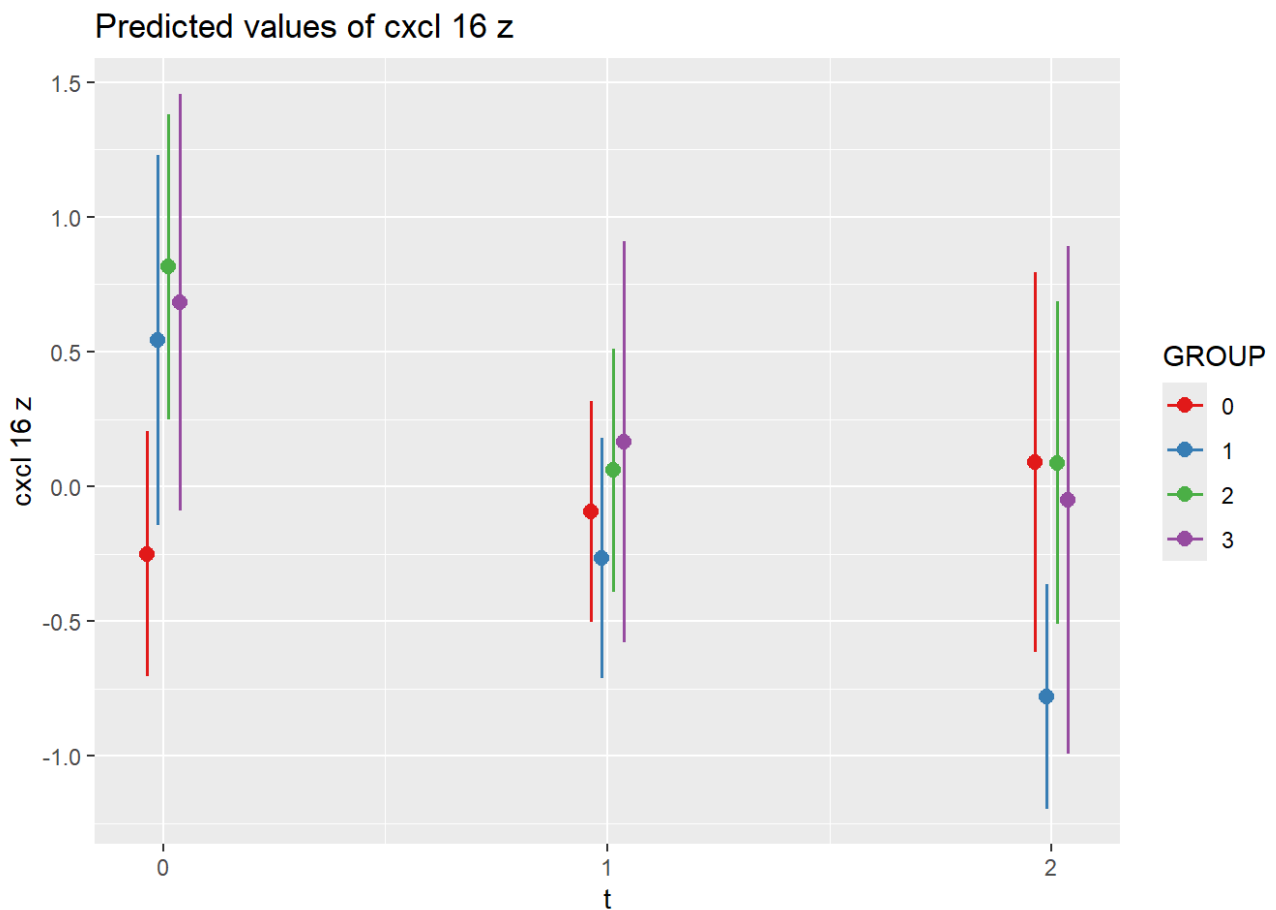
```
plot_model(cxcl16Model_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```



```
plot_model(cxcl16Model_hetero, type = "pred",  
           terms = c("t", "GROUP"),  
           pred.type = "fe")
```

```
## Ignoring unknown labels:  
## • linetype : "GROUP"  
## • shape : "GROUP"
```

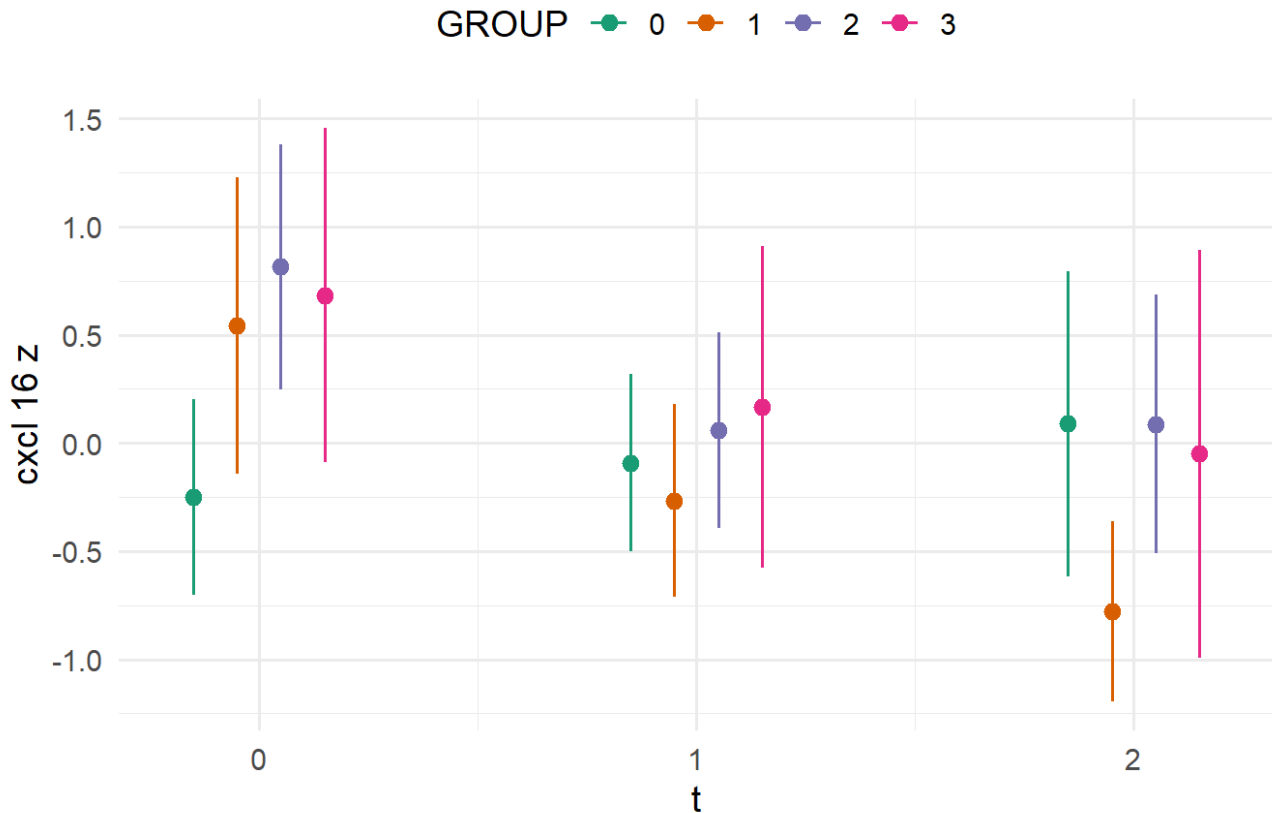



```
plot_model(cxcl16Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

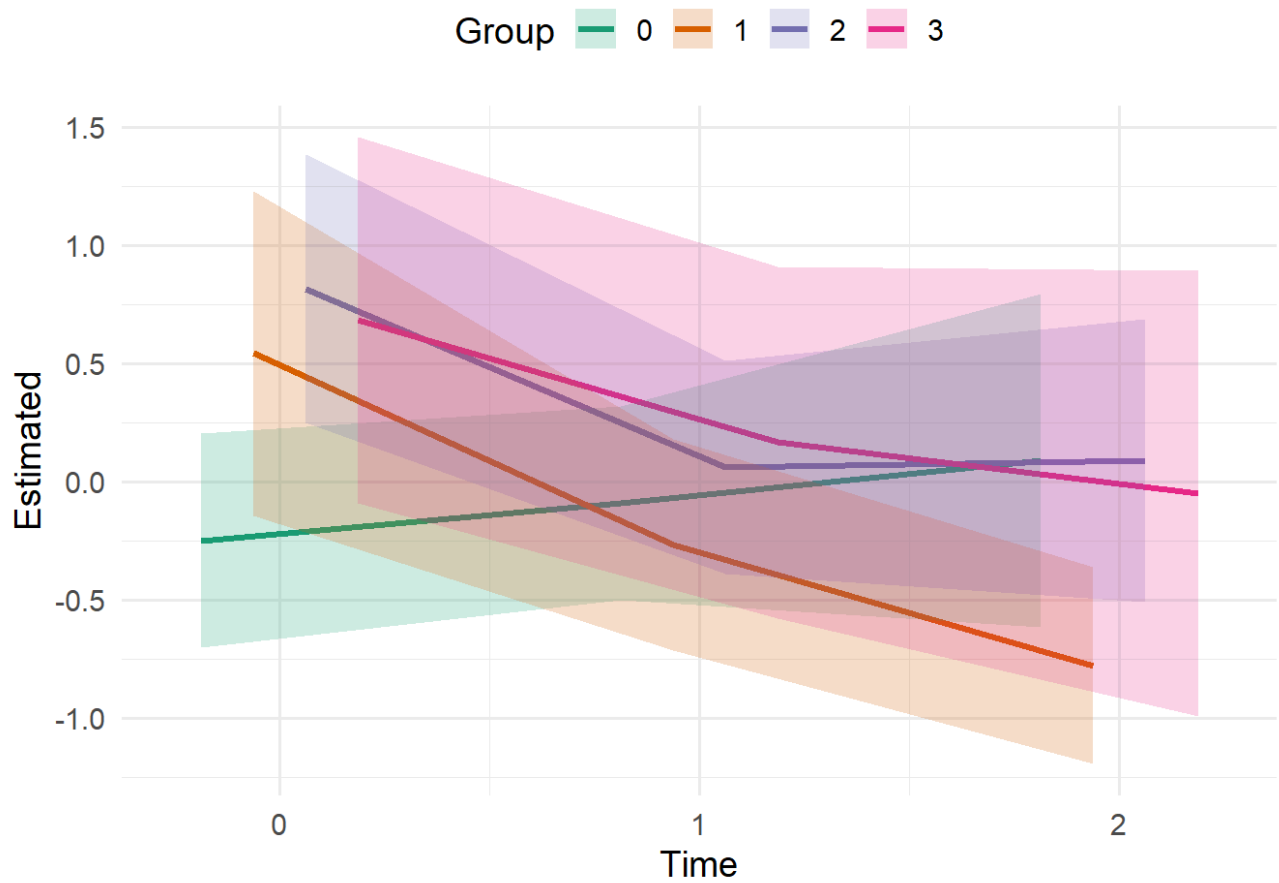
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of cxcl 16 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(cxcl16Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



8.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(cxcl16Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##   GROUP    emmean      SE  df asymp.LCL asymp.UCL
##   0      -0.20738 0.214 Inf   -0.62682    0.212
##   1       0.58531 0.353 Inf   -0.10632    1.277
##   2       0.85714 0.288 Inf    0.29170    1.423
##   3       0.72591 0.375 Inf   -0.00846    1.460
##
## t = 1:
##   GROUP    emmean      SE  df asymp.LCL asymp.UCL
##   0      -0.04975 0.182 Inf   -0.40685    0.307
##   1      -0.22370 0.225 Inf   -0.66410    0.217
##   2       0.10240 0.226 Inf   -0.34134    0.546
##   3       0.20792 0.356 Inf   -0.49071    0.907
##
## t = 2:
##   GROUP    emmean      SE  df asymp.LCL asymp.UCL
##   0       0.13194 0.343 Inf   -0.54112    0.805
##   1      -0.73700 0.205 Inf   -1.13956   -0.334
##   2       0.12965 0.296 Inf   -0.44988    0.709
##   3      -0.00762 0.460 Inf   -0.90892    0.894
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(cxcl16Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##   contrast          estimate      SE  df z.ratio p.value
##   GROUP0 - GROUP1    0.0834 0.281 Inf    0.297 0.9909
##   GROUP0 - GROUP2   -0.4048 0.252 Inf   -1.606 0.3751
##   GROUP0 - GROUP3   -0.3505 0.330 Inf   -1.063 0.7119
##   GROUP1 - GROUP2   -0.4882 0.273 Inf   -1.786 0.2801
##   GROUP1 - GROUP3   -0.4339 0.394 Inf   -1.101 0.6888
##   GROUP2 - GROUP3    0.0543 0.364 Inf    0.149 0.9988
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(cxcl16Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.79268 0.360 Inf  -2.205  0.1219
## GROUP0 - GROUP2 -1.06452 0.318 Inf  -3.343  0.0046
## GROUP0 - GROUP3 -0.93329 0.415 Inf  -2.247  0.1107
## GROUP1 - GROUP2 -0.27184 0.384 Inf  -0.708  0.8942
## GROUP1 - GROUP3 -0.14061 0.491 Inf  -0.286  0.9918
## GROUP2 - GROUP3  0.13123 0.455 Inf   0.288  0.9917
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.17395 0.311 Inf   0.559  0.9441
## GROUP0 - GROUP2 -0.15215 0.279 Inf  -0.546  0.9476
## GROUP0 - GROUP3 -0.25767 0.381 Inf  -0.676  0.9061
## GROUP1 - GROUP2 -0.32610 0.316 Inf  -1.032  0.7304
## GROUP1 - GROUP3 -0.43162 0.450 Inf  -0.960  0.7723
## GROUP2 - GROUP3 -0.10552 0.417 Inf  -0.253  0.9943
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.86894 0.403 Inf   2.158  0.1350
## GROUP0 - GROUP2  0.00229 0.364 Inf   0.006  1.0000
## GROUP0 - GROUP3  0.13956 0.426 Inf   0.328  0.9879
## GROUP1 - GROUP2 -0.86665 0.354 Inf  -2.449  0.0682
## GROUP1 - GROUP3 -0.72938 0.507 Inf  -1.438  0.4756
## GROUP2 - GROUP3  0.13727 0.470 Inf   0.292  0.9913
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(cxcl16Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.1576 0.241 Inf  -0.654  0.7901
## t0 - t2      -0.3393 0.424 Inf  -0.799  0.7033
## t1 - t2      -0.1817 0.339 Inf  -0.536  0.8535
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1       0.8090 0.301 Inf   2.691  0.0195
## t0 - t2       1.3223 0.379 Inf   3.490  0.0014
## t1 - t2       0.5133 0.241 Inf   2.129  0.0841
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1       0.7547 0.285 Inf   2.651  0.0219
## t0 - t2       0.7275 0.406 Inf   1.792  0.1721
## t1 - t2      -0.0272 0.302 Inf  -0.090  0.9955
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1       0.5180 0.386 Inf   1.341  0.3723
## t0 - t2       0.7335 0.512 Inf   1.432  0.3244
## t1 - t2       0.2155 0.431 Inf   0.500  0.8711
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

9 Model ifny

```
bptest(lm(ifny ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(ifny ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 11.951, df = 14, p-value = 0.6102
```

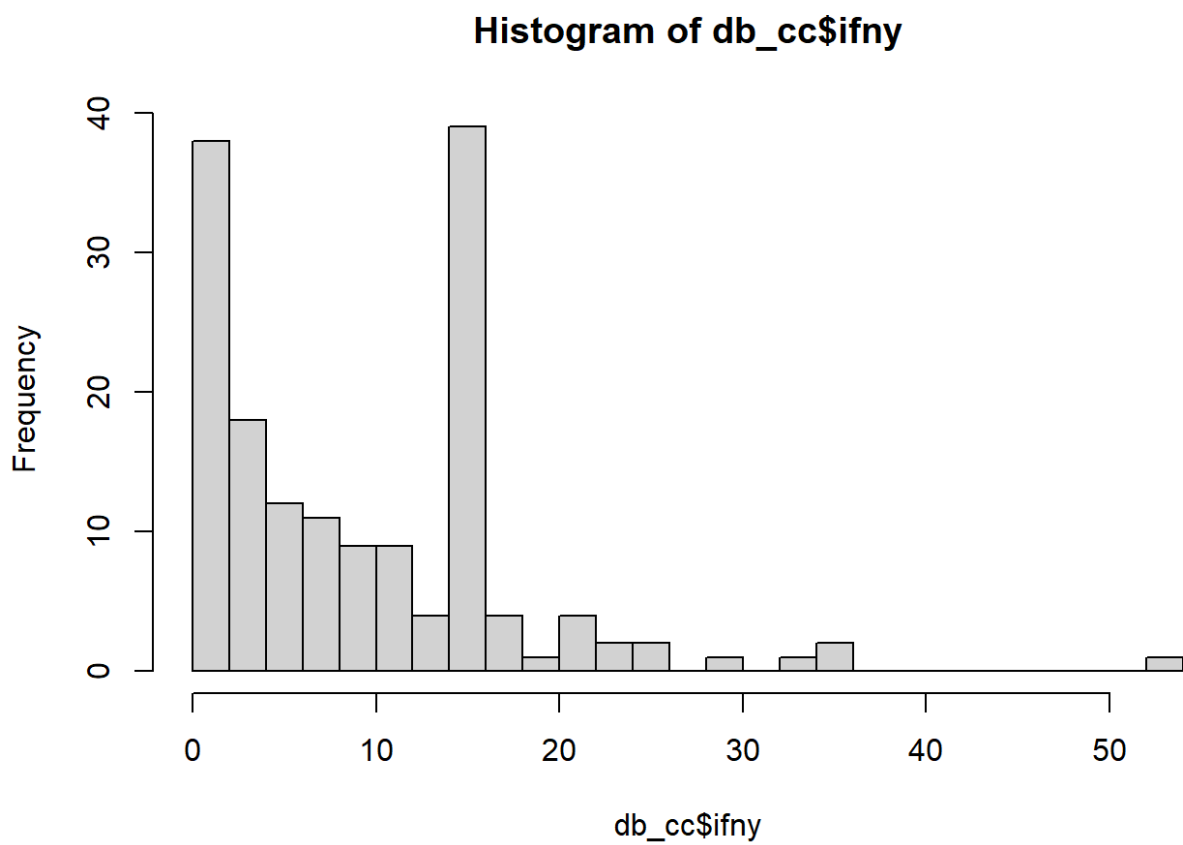
```
vars <- c("ifny","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

nrow(db); nrow(db_cc)
```

```
## [1] 243
```

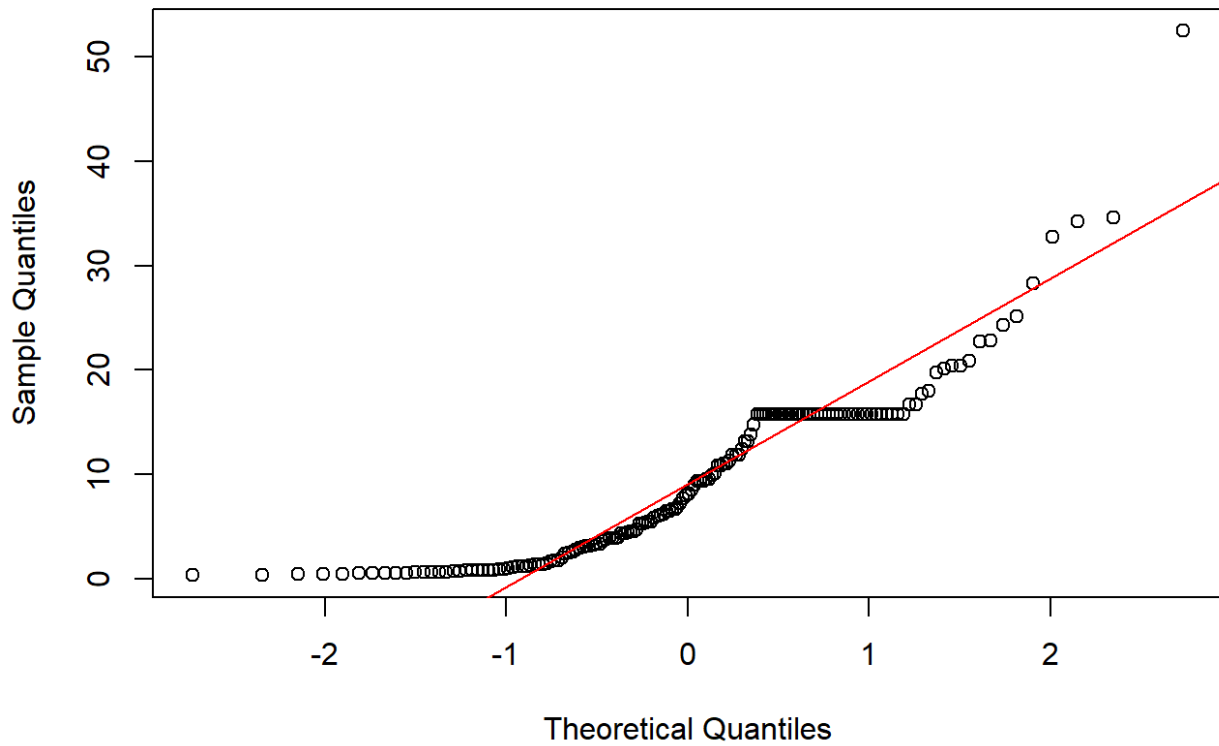
```
## [1] 158
```

```
hist(db_cc$ifny, breaks=30)
```



```
qqnorm(db_cc$ifny); qqline(db_cc$ifny, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$ifny)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.3089  2.4045   8.0572   9.6108 15.7202 52.4929
```

```
min(db_cc$ifny, na.rm=TRUE)
```

```
## [1] 0.3089046
```



```
db$ifny_z <- scale(log(db$ifny))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$ifny_z <- scale(log(db_cc$ifny))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

ifnyModel_hetero <- glmmTMB(
  ifny_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(ifnyModel_hetero)
```

```
## Family: gaussian ( identity )
## Formula:
## ifny_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion: ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    336.9    395.1   -149.5    298.9     139
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id       (Intercept) 0.1435   0.3788
##   Residual              NA       NA
## Number of obs: 158, groups: id, 77
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.41658   1.90979   0.218 0.827327
## t1             0.51372   0.23132   2.221 0.026363 *
## t2            -0.06706   0.37406  -0.179 0.857730
## GROUP1        -1.17379   0.34045  -3.448 0.000565 ***
## GROUP2        -1.12765   0.30798  -3.661 0.000251 ***
## GROUP3        -1.76790   0.38377  -4.607 4.09e-06 ***
## Age           -0.01150   0.02464  -0.467 0.640657
## SexM           0.25777   0.12099   2.131 0.033130 *
## CIRS_severity_0 0.19611   0.15120   1.297 0.194637
## t1:GROUP1      1.18241   0.34480   3.429 0.000605 ***
## t2:GROUP1      1.85043   0.43446   4.259 2.05e-05 ***
## t1:GROUP2      1.14856   0.31168   3.685 0.000229 ***
## t2:GROUP2      2.03546   0.40518   5.024 5.07e-07 ***
## t1:GROUP3      1.67859   0.38324   4.380 1.19e-05 ***
## t2:GROUP3      2.91982   0.45425   6.428 1.30e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.2737    0.1190  -2.299 0.021493 *
## t1           -0.8282    0.2334  -3.549 0.000387 ***
## t2           -0.2710    0.1724  -1.572 0.115917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```

ifnyModel_homo <- glmmTMB(
  ifny_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(ifnyModel_homo, ifnyModel_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## ifnyModel_homo: ifny_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severi
ty_0 + , zi=~0, disp=~1
## ifnyModel_homo:      (1 | id), zi=~0, disp=~t
## ifnyModel_hetero: ifny_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## ifnyModel_hetero:      (1 | id), zi=~0, disp=~t
##
##      Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Chi
sq)
## ifnyModel_homo    17 349.88 401.95 -157.94    315.88
## ifnyModel_hetero  19 336.94 395.13 -149.47    298.94 16.948      2 0.0002
089 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

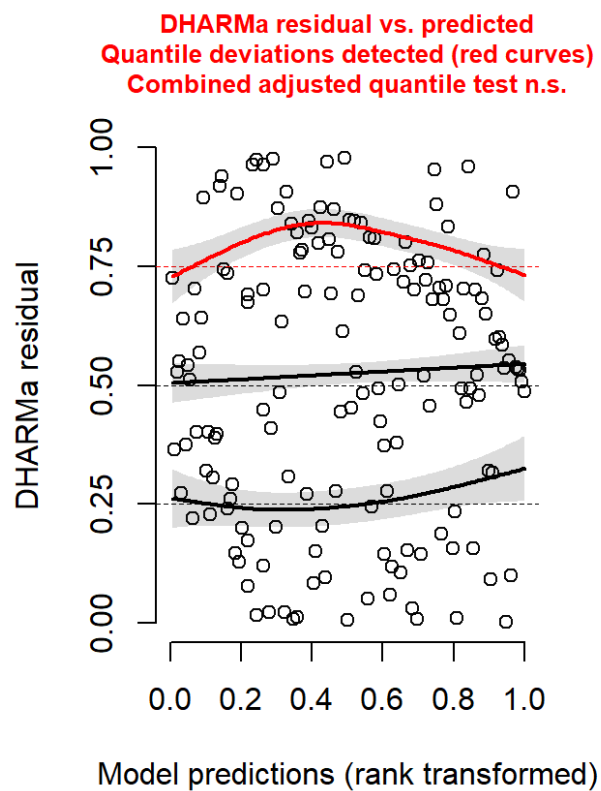
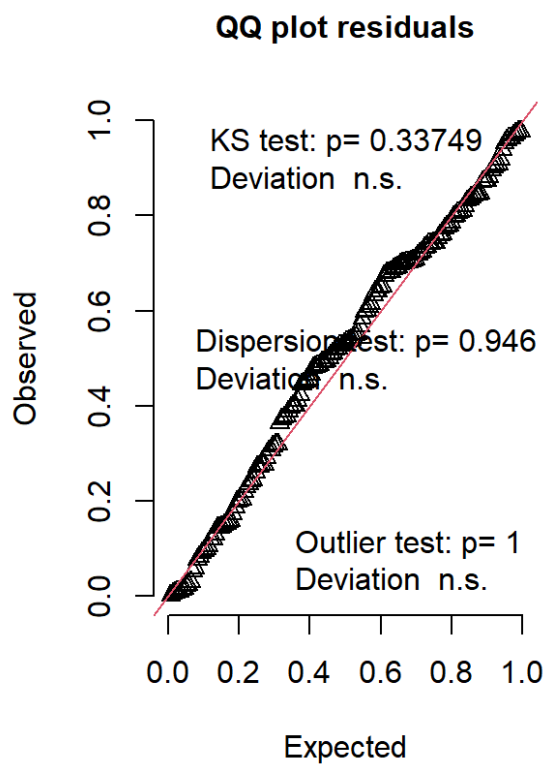
```

```

sim_res <- simulateResiduals(ifnyModel_hetero, n = 1000)
plot(sim_res)

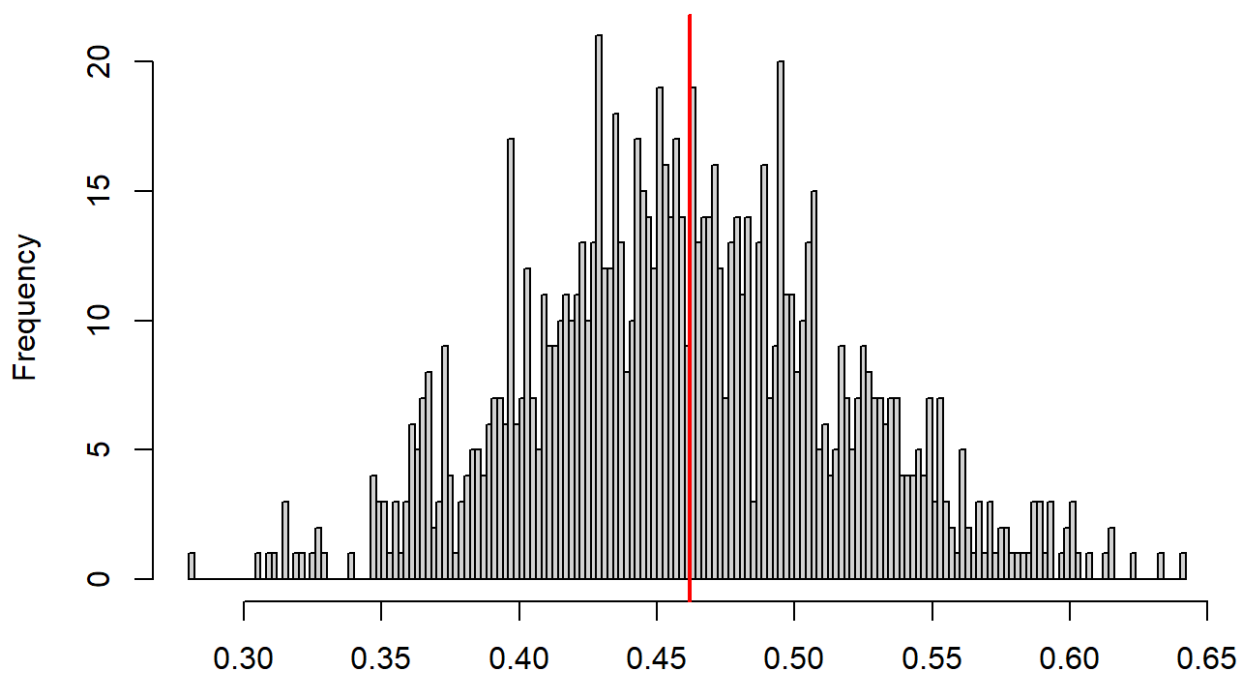
```

DHARMA residual



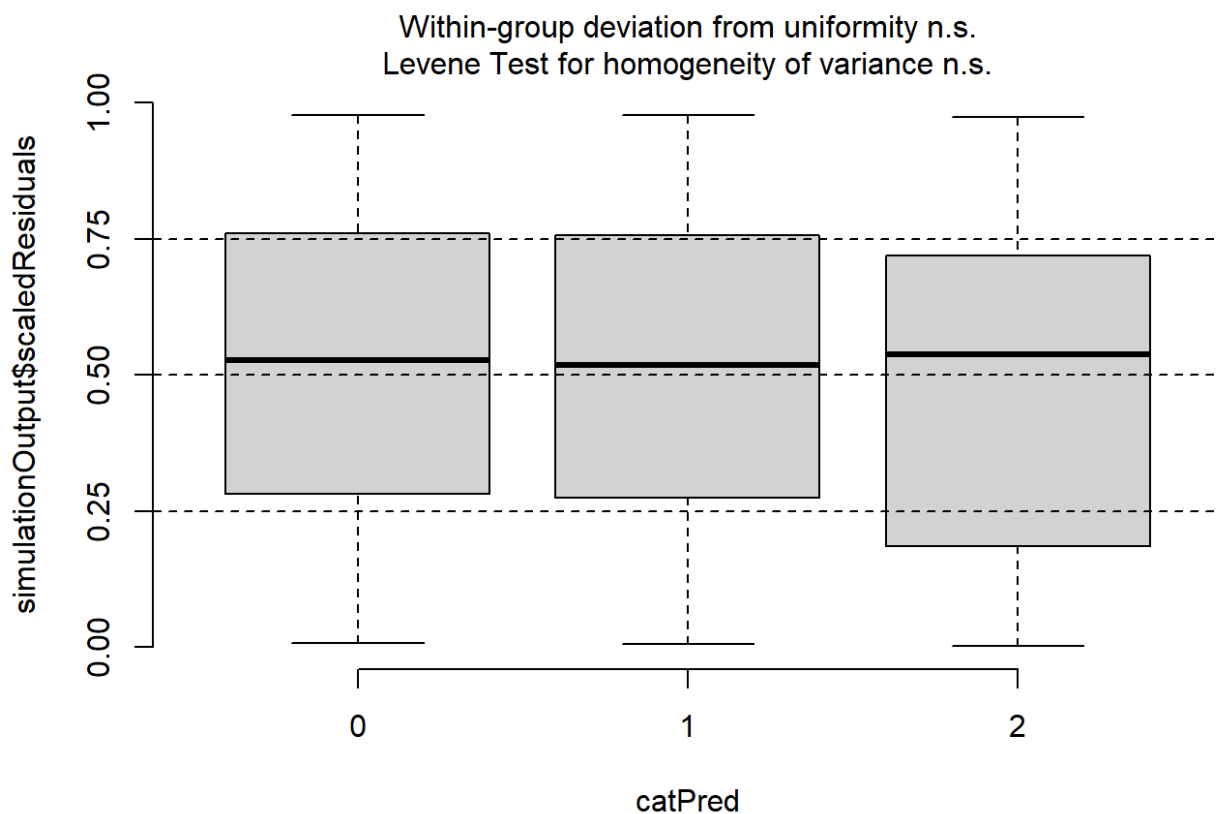
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##  
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.  
## simulated  
##  
## data: simulationOutput  
## dispersion = 1.0038, p-value = 0.946  
## alternative hypothesis: two.sided
```

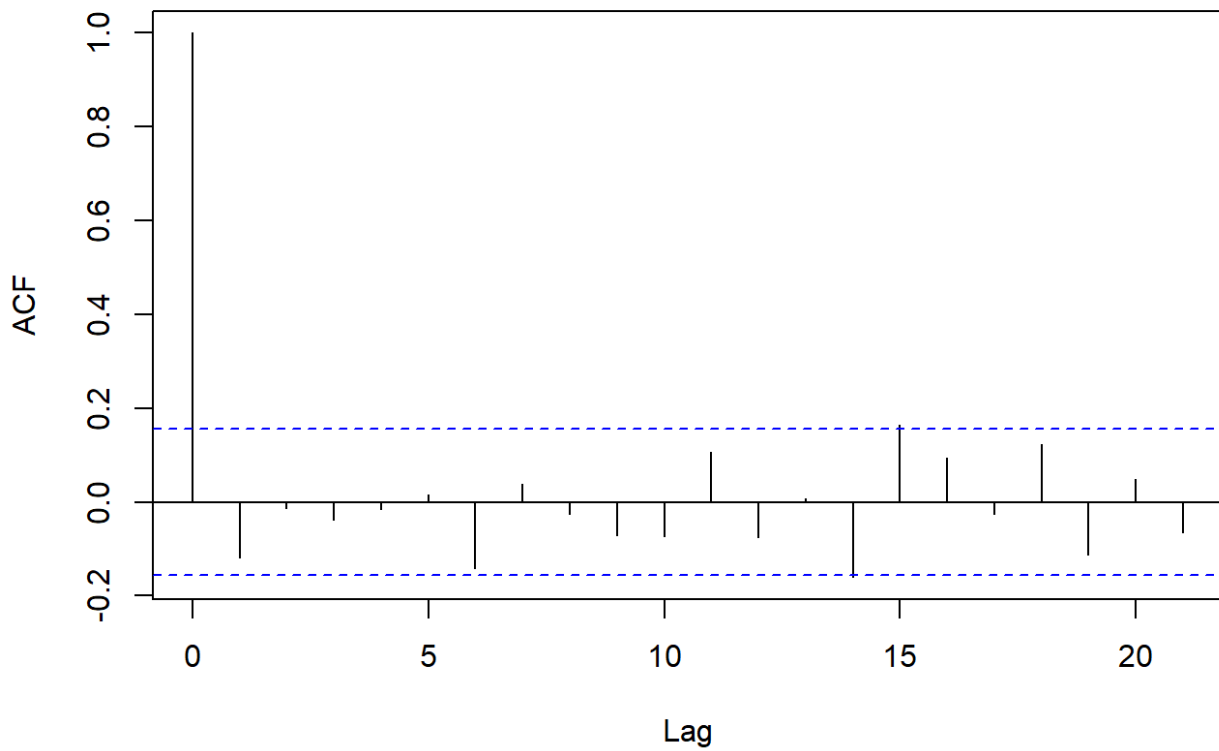
```
testQuantiles(sim_res, db_cc$t)
```



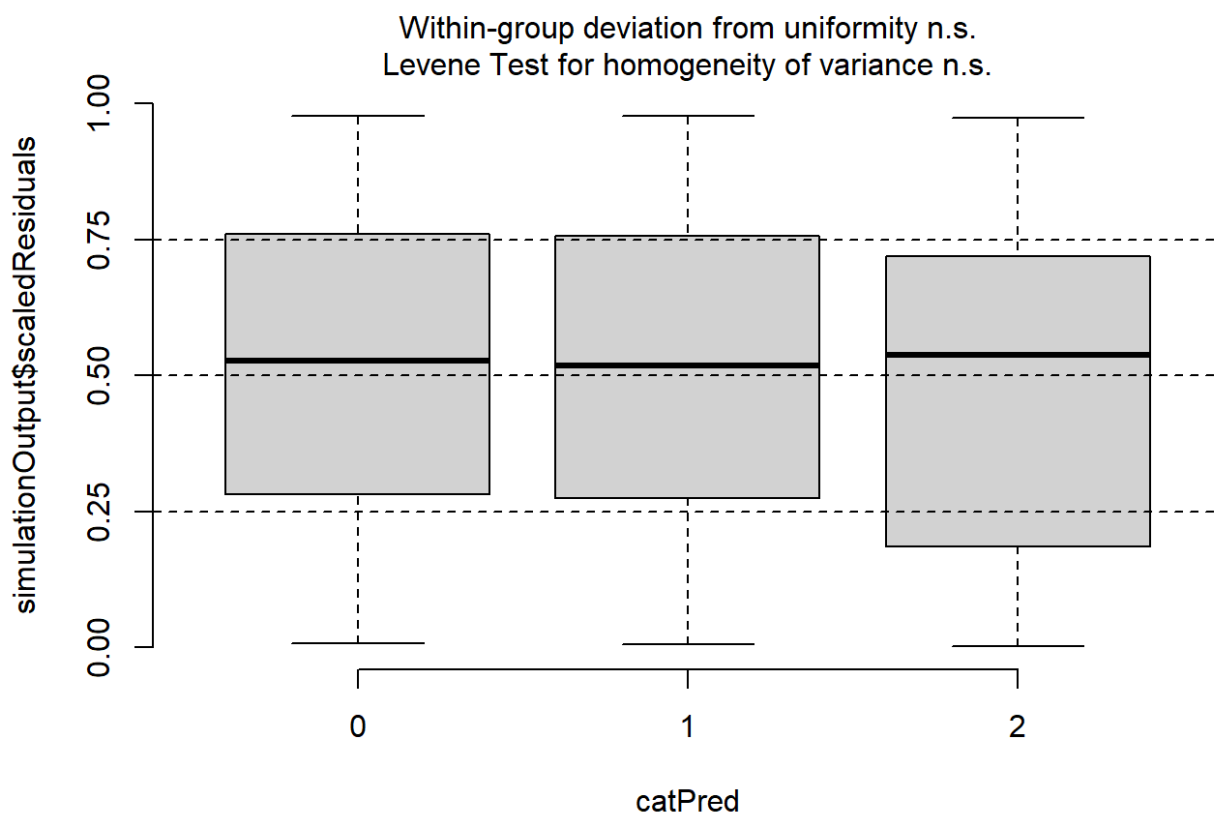
```
## NULL
```

```
acf(residuals(ifnyModel_hetero))
```

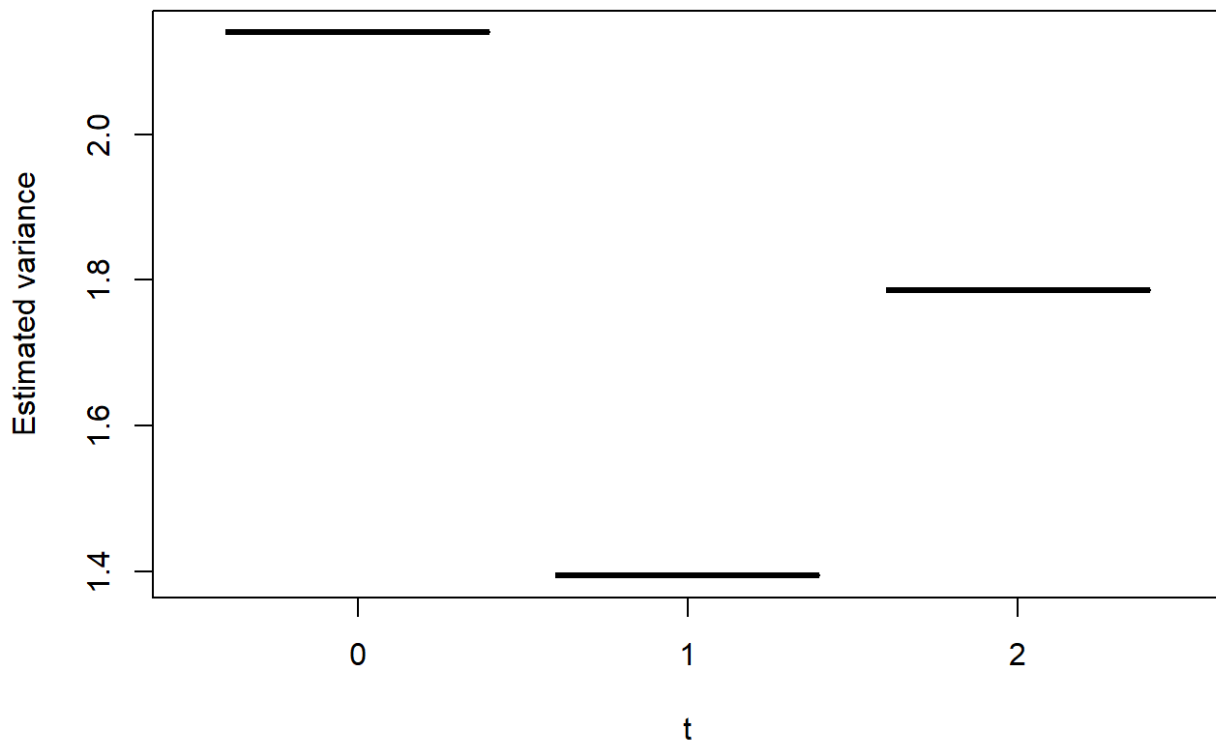
Series residuals(ifnyModel_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

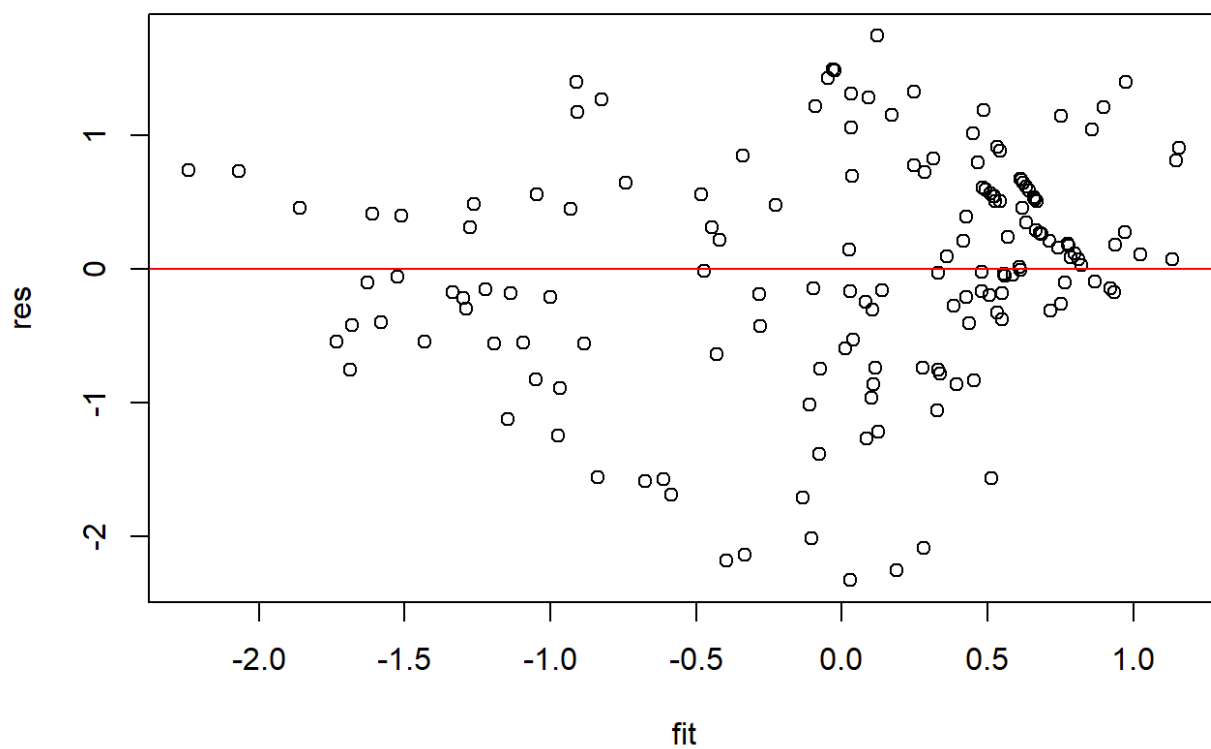


```
pred_disp <- predict(ifnyModel_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



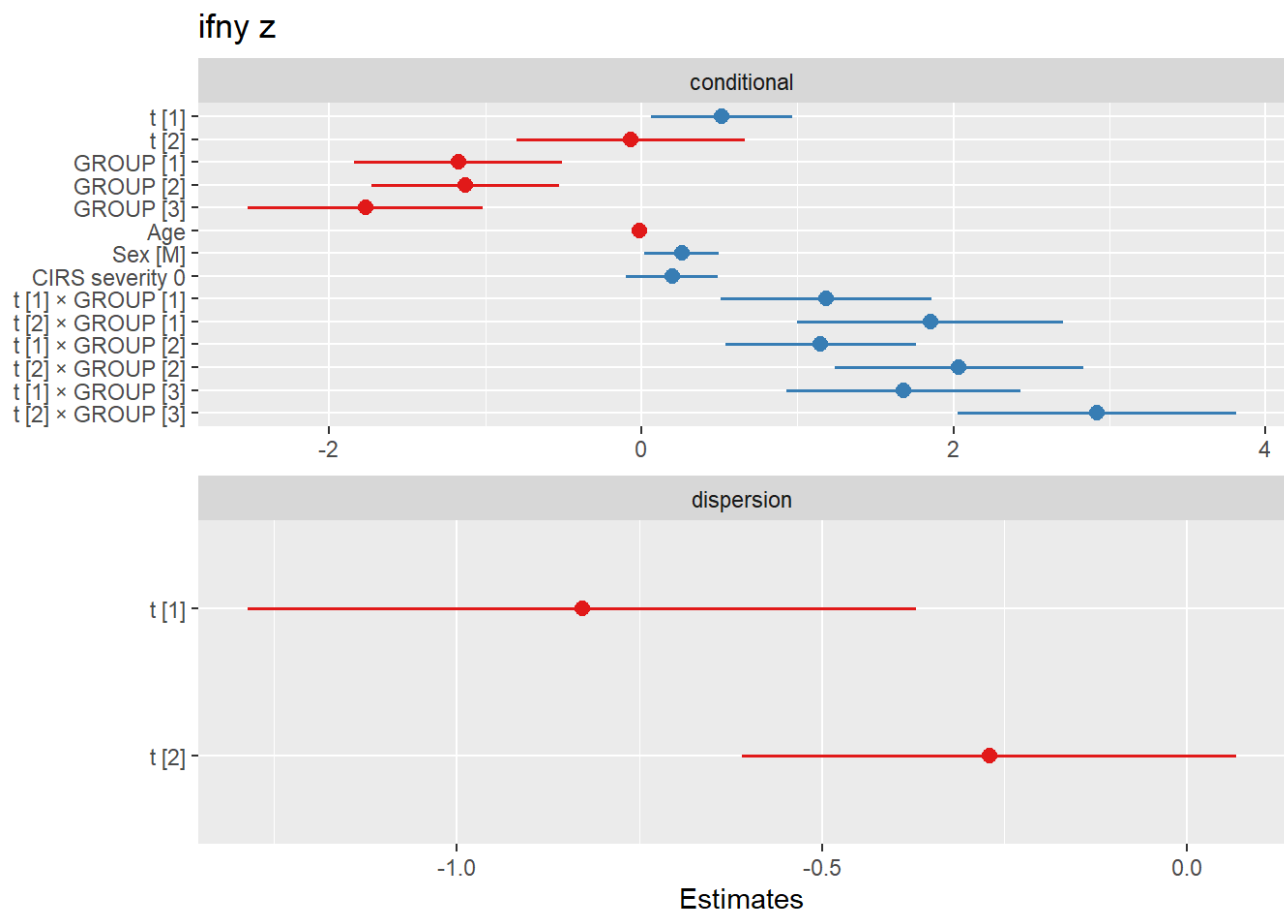
```
res <- residuals(ifnyModel_hetero, type="pearson")
fit <- fitted(ifnyModel_hetero)

plot(fit, res)
abline(h=0, col="red")
```



9.1 Visualizzazione del modello

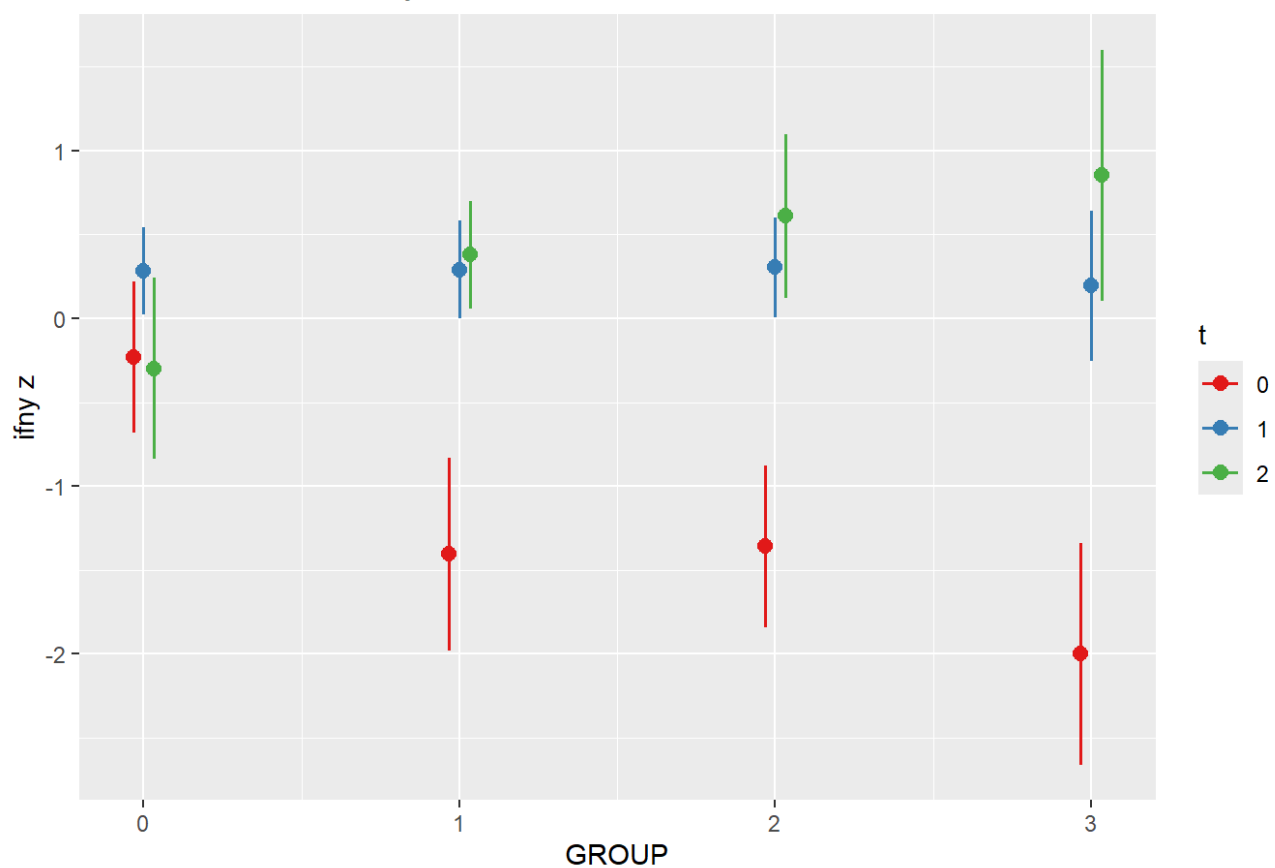
```
plot_model(ifnyModel_hetero)
```

```
plot_model(ifnyModel_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

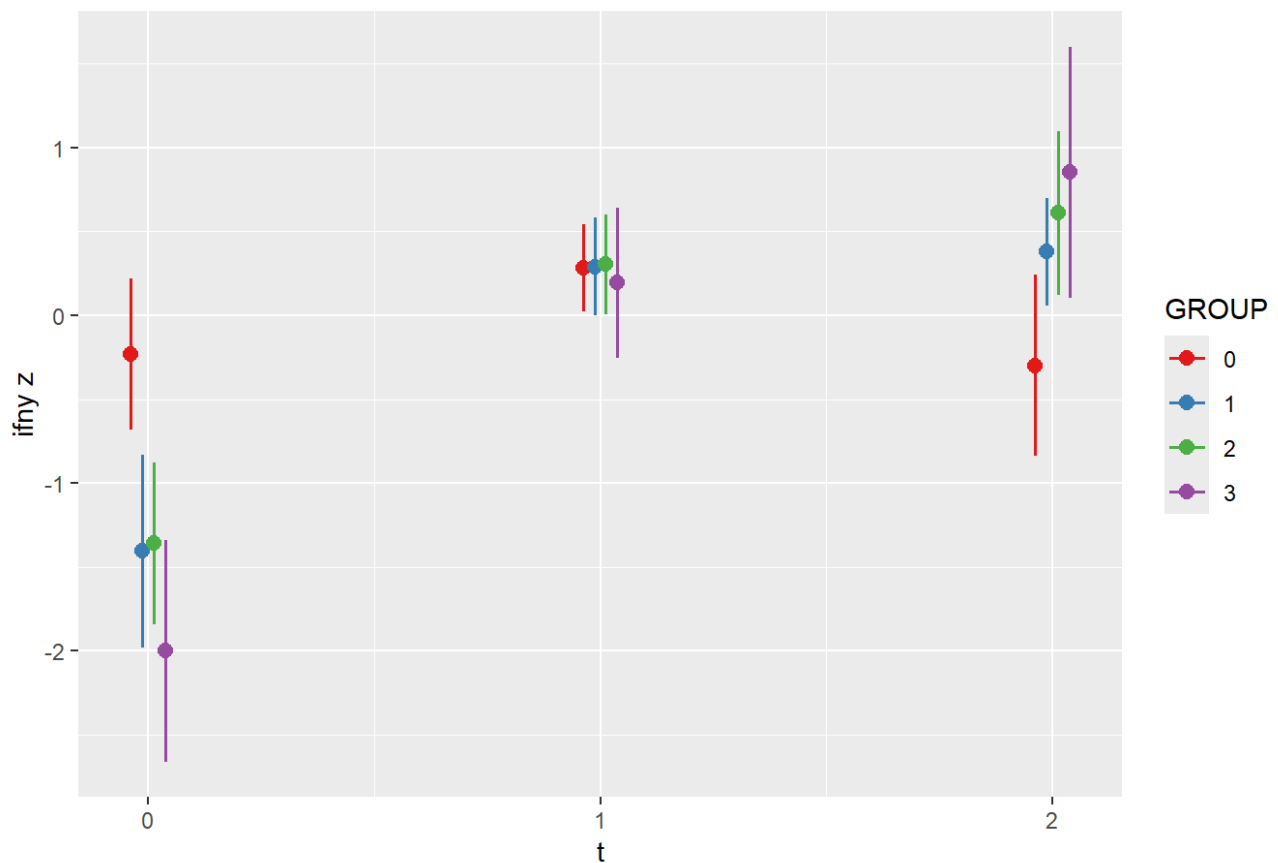
Predicted values of ifny z



```
plot_model(ifnyModel_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of ifny z

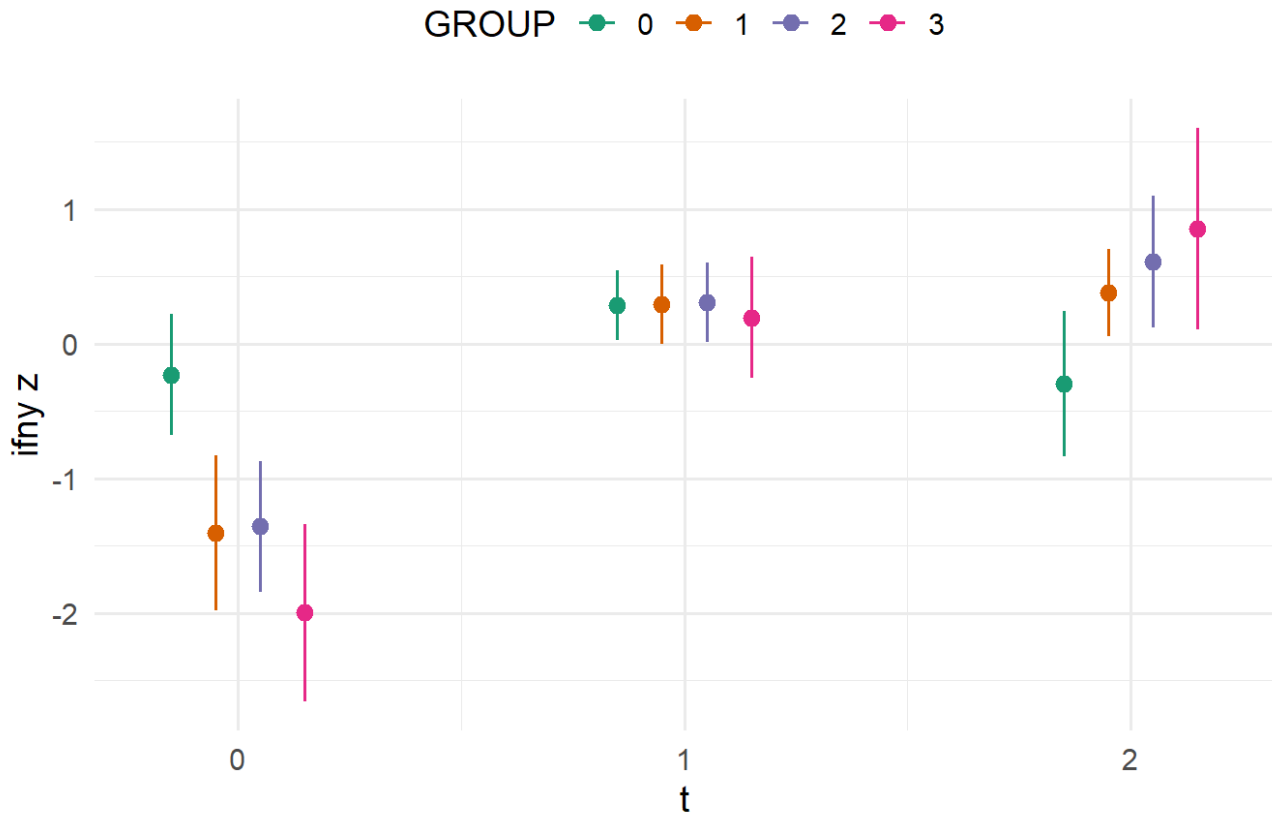


```
plot_model(ifnyModel_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

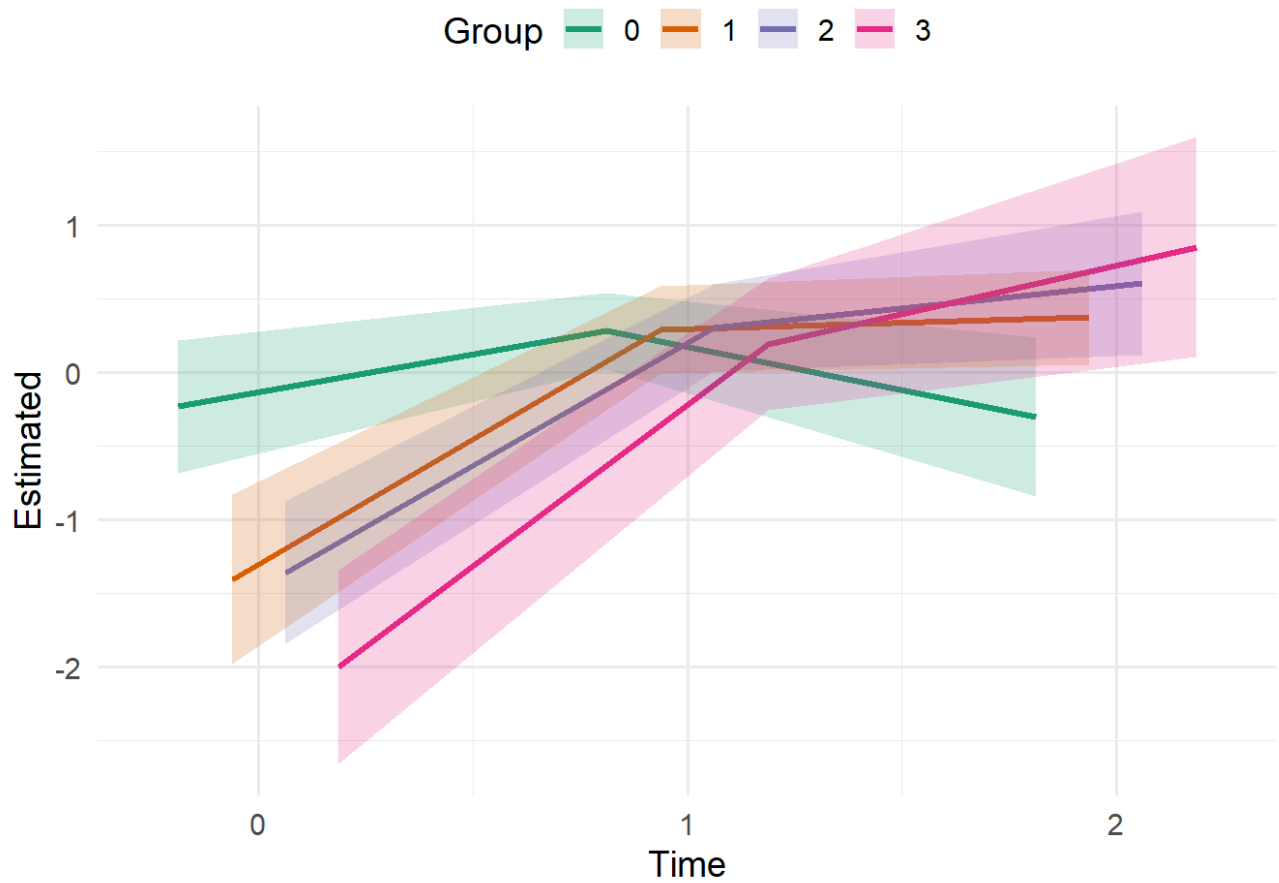
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of ifny z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(ifnyModel_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



9.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(ifnyModel_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##   GROUP emmean    SE  df asymp.LCL asymp.UCL
##   0      -0.102 0.221 Inf   -0.5340    0.331
##   1      -1.275 0.293 Inf   -1.8490   -0.702
##   2      -1.229 0.246 Inf   -1.7104   -0.748
##   3      -1.869 0.327 Inf   -2.5095   -1.229
##
## t = 1:
##   GROUP emmean    SE  df asymp.LCL asymp.UCL
##   0       0.412 0.113 Inf    0.1900    0.634
##   1       0.421 0.145 Inf    0.1373    0.704
##   2       0.433 0.147 Inf    0.1449    0.721
##   3       0.323 0.214 Inf   -0.0972    0.743
##
## t = 2:
##   GROUP emmean    SE  df asymp.LCL asymp.UCL
##   0      -0.169 0.270 Inf   -0.6971    0.360
##   1       0.508 0.159 Inf    0.1957    0.820
##   2       0.739 0.247 Inf    0.2550    1.223
##   3       0.983 0.374 Inf    0.2496    1.717
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(ifyModel_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##   contrast          estimate    SE  df z.ratio p.value
##   GROUP0 - GROUP1    0.1628 0.201 Inf    0.811  0.8493
##   GROUP0 - GROUP2    0.0663 0.185 Inf    0.358  0.9842
##   GROUP0 - GROUP3    0.2351 0.229 Inf    1.028  0.7329
##   GROUP1 - GROUP2   -0.0965 0.200 Inf   -0.483  0.9628
##   GROUP1 - GROUP3    0.0722 0.273 Inf    0.264  0.9935
##   GROUP2 - GROUP3    0.1688 0.254 Inf    0.664  0.9106
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(ifyModel_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  1.17379 0.340 Inf   3.448  0.0032
## GROUP0 - GROUP2  1.12765 0.308 Inf   3.661  0.0014
## GROUP0 - GROUP3  1.76790 0.384 Inf   4.607 <0.0001
## GROUP1 - GROUP2 -0.04614 0.348 Inf  -0.133  0.9992
## GROUP1 - GROUP3  0.59410 0.426 Inf   1.396  0.5019
## GROUP2 - GROUP3  0.64024 0.398 Inf   1.608  0.3741
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.00861 0.193 Inf  -0.045  1.0000
## GROUP0 - GROUP2 -0.02091 0.173 Inf  -0.121  0.9994
## GROUP0 - GROUP3  0.08931 0.228 Inf   0.392  0.9796
## GROUP1 - GROUP2 -0.01229 0.207 Inf  -0.059  0.9999
## GROUP1 - GROUP3  0.09792 0.275 Inf   0.356  0.9846
## GROUP2 - GROUP3  0.11021 0.252 Inf   0.438  0.9720
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.67663 0.319 Inf  -2.122  0.1460
## GROUP0 - GROUP2 -0.90780 0.311 Inf  -2.915  0.0186
## GROUP0 - GROUP3 -1.15192 0.335 Inf  -3.443  0.0032
## GROUP1 - GROUP2 -0.23117 0.292 Inf  -0.791  0.8588
## GROUP1 - GROUP3 -0.47529 0.413 Inf  -1.151  0.6581
## GROUP2 - GROUP3 -0.24412 0.397 Inf  -0.615  0.9272
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(ifnyModel_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1    -0.5137 0.231 Inf  -2.221  0.0677
## t0 - t2     0.0671 0.374 Inf   0.179  0.9824
## t1 - t2     0.5808 0.282 Inf   2.058  0.0986
##
## GROUP = 1:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1    -1.6961 0.272 Inf  -6.231 <0.0001
## t0 - t2    -1.7834 0.320 Inf  -5.565 <0.0001
## t1 - t2    -0.0872 0.184 Inf  -0.473  0.8840
##
## GROUP = 2:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1    -1.6623 0.256 Inf  -6.504 <0.0001
## t0 - t2    -1.9684 0.351 Inf  -5.607 <0.0001
## t1 - t2    -0.3061 0.254 Inf  -1.207  0.4487
##
## GROUP = 3:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1    -2.1923 0.337 Inf  -6.501 <0.0001
## t0 - t2    -2.8528 0.479 Inf  -5.953 <0.0001
## t1 - t2    -0.6605 0.375 Inf  -1.763  0.1822
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

10 Model il1a

```
bptest(lm(il1a ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(il1a ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 19.479, df = 14, p-value = 0.1474
```

```
vars <- c("il1a","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

nrow(db); nrow(db_cc)
```

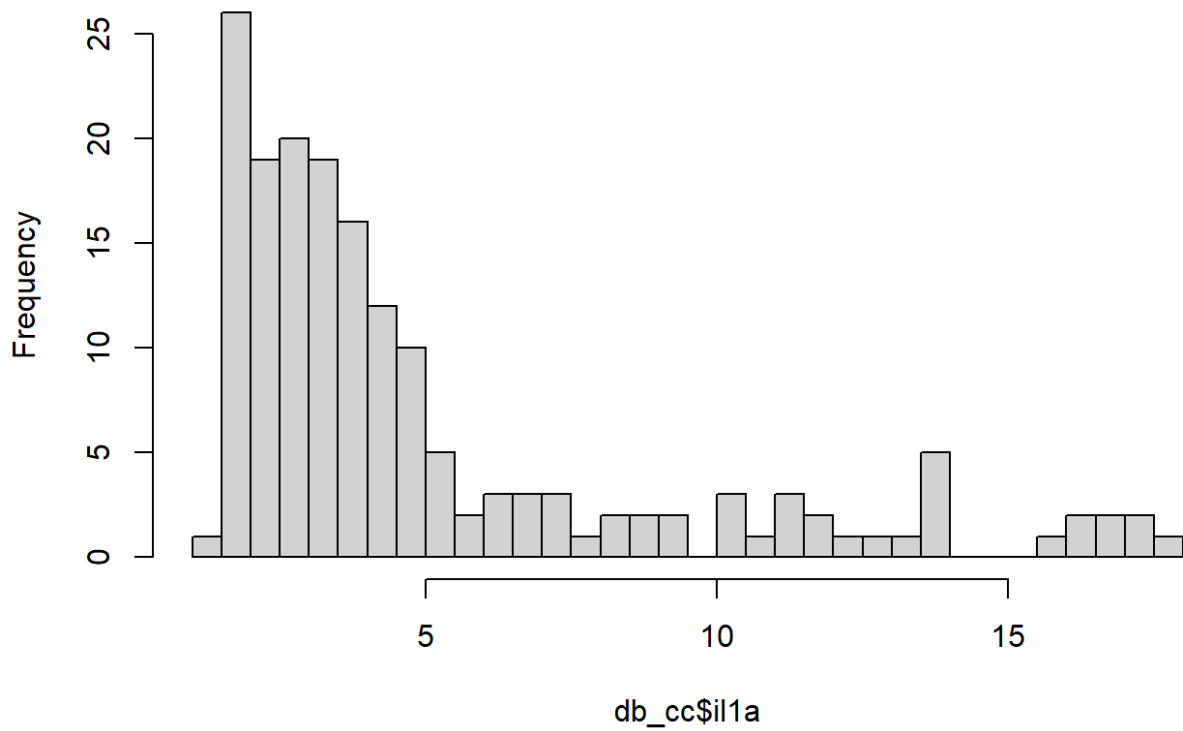


```
## [1] 243
```

```
## [1] 171
```

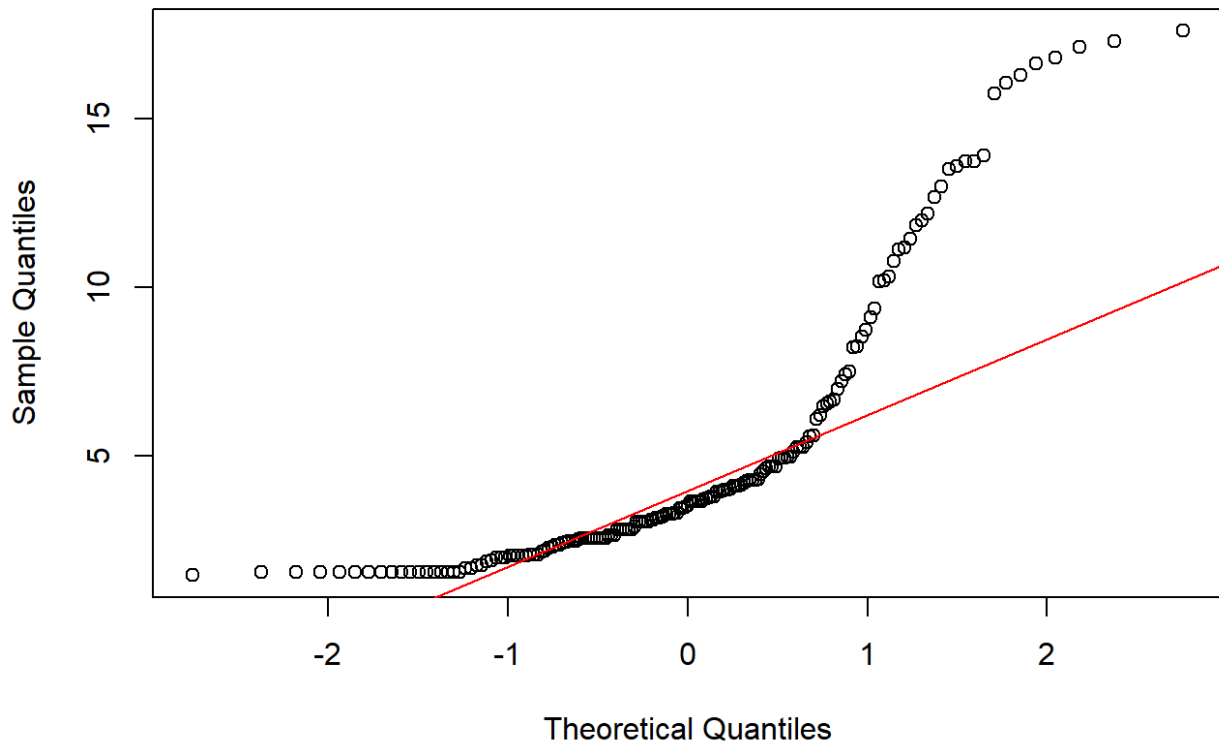
```
hist(db_cc$il1a, breaks=30)
```

Histogram of db_cc\$il1a



```
qqnorm(db_cc$il1a); qqline(db_cc$il1a, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$il1a)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.468   2.457   3.539   5.016   5.492   17.614
```

```
min(db_cc$il1a, na.rm=TRUE)
```

```
## [1] 1.468167
```

```
db$il1a_z <- scale(log(db$il1a))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$il1a_z <- scale(log(db_cc$il1a))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

il1aModel_hetero <- glmmTMB(
  il1a_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(il1aModel_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## illa_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion:                ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    459.1    518.8   -210.5    421.1     152
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id       (Intercept) 0.3208   0.5664
##   Residual              NA       NA
## Number of obs: 171, groups: id, 77
##
## Conditional model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.48381    2.55858   0.580   0.5620
## t1              0.58914    0.28353   2.078   0.0377 *
## t2              0.87970    0.45716   1.924   0.0543 .
## GROUP1         0.22092    0.42677   0.518   0.6047
## GROUP2         0.29457    0.38478   0.765   0.4439
## GROUP3        -0.24692    0.48702  -0.507   0.6122
## Age            -0.02077    0.03306  -0.628   0.5298
## SexM           -0.14239    0.16667  -0.854   0.3929
## CIRS_severity_0 -0.10612    0.17674  -0.601   0.5482
## t1:GROUP1      -0.72965    0.42353  -1.723   0.0849 .
## t2:GROUP1      -1.07673    0.52767  -2.041   0.0413 *
## t1:GROUP2      -0.18916    0.37199  -0.508   0.6111
## t2:GROUP2       0.01298    0.47155   0.028   0.9780
## t1:GROUP3      -0.23811    0.46900  -0.508   0.6117
## t2:GROUP3      -0.62914    0.55149  -1.141   0.2540
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.06342    0.11744  -0.540   0.58917
## t1           -0.78220    0.25224  -3.101   0.00193 **
## t2           -0.20685    0.16596  -1.246   0.21261
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

il1aModel_homo <- glmmTMB(
  il1a_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(il1aModel_homo, il1aModel_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## il1aModel_homo: il1a_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severi
ty_0 + , zi=~0, disp=~1
## il1aModel_homo:      (1 | id), zi=~0, disp=~t
## il1aModel_hetero: il1a_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## il1aModel_hetero:      (1 | id), zi=~0, disp=~t
##
##          Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Chi
sq)
## il1aModel_homo    17 468.69 522.10 -217.35   434.69
## il1aModel_hetero  19 459.09 518.78 -210.54   421.09 13.608      2  0.001
109 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

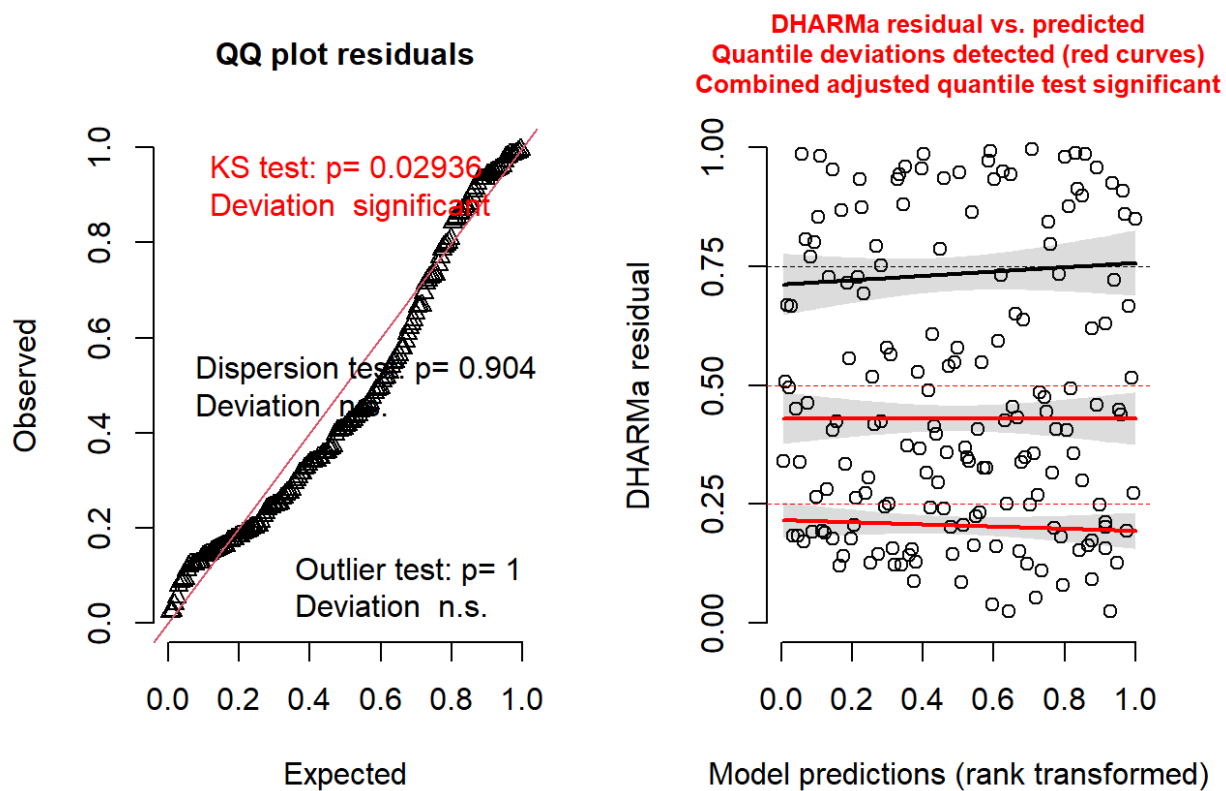
```

```

sim_res <- simulateResiduals(il1aModel_hetero, n = 1000)
plot(sim_res)

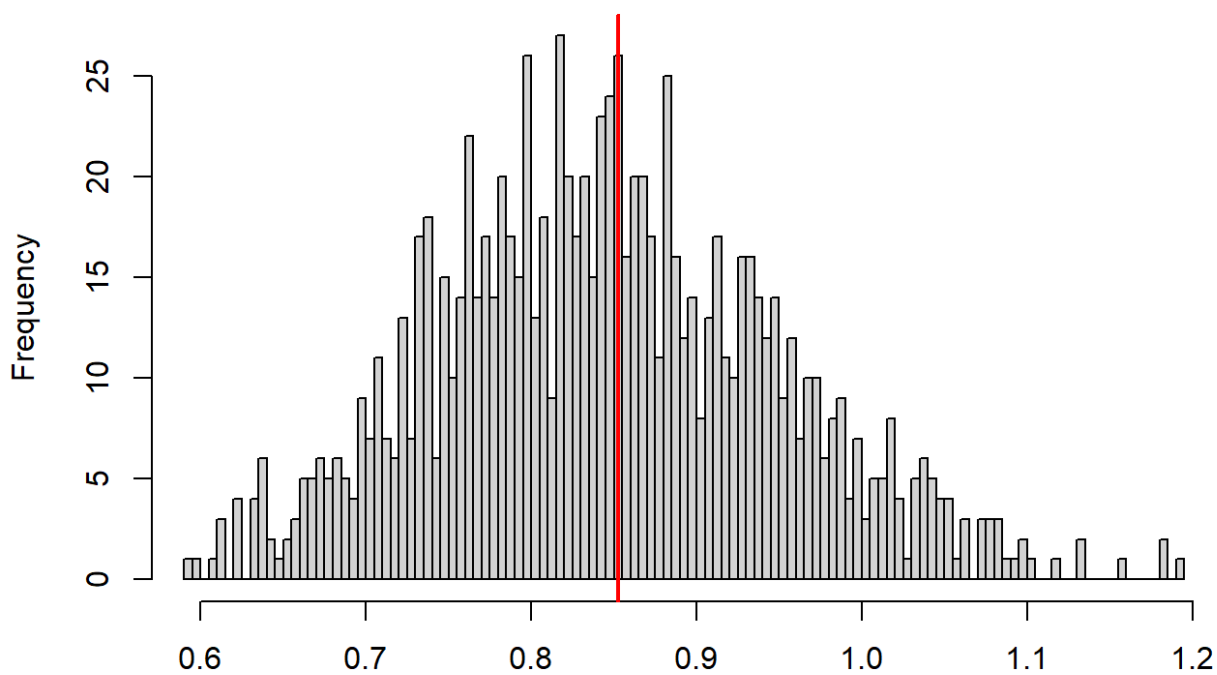
```

DHARMA residual



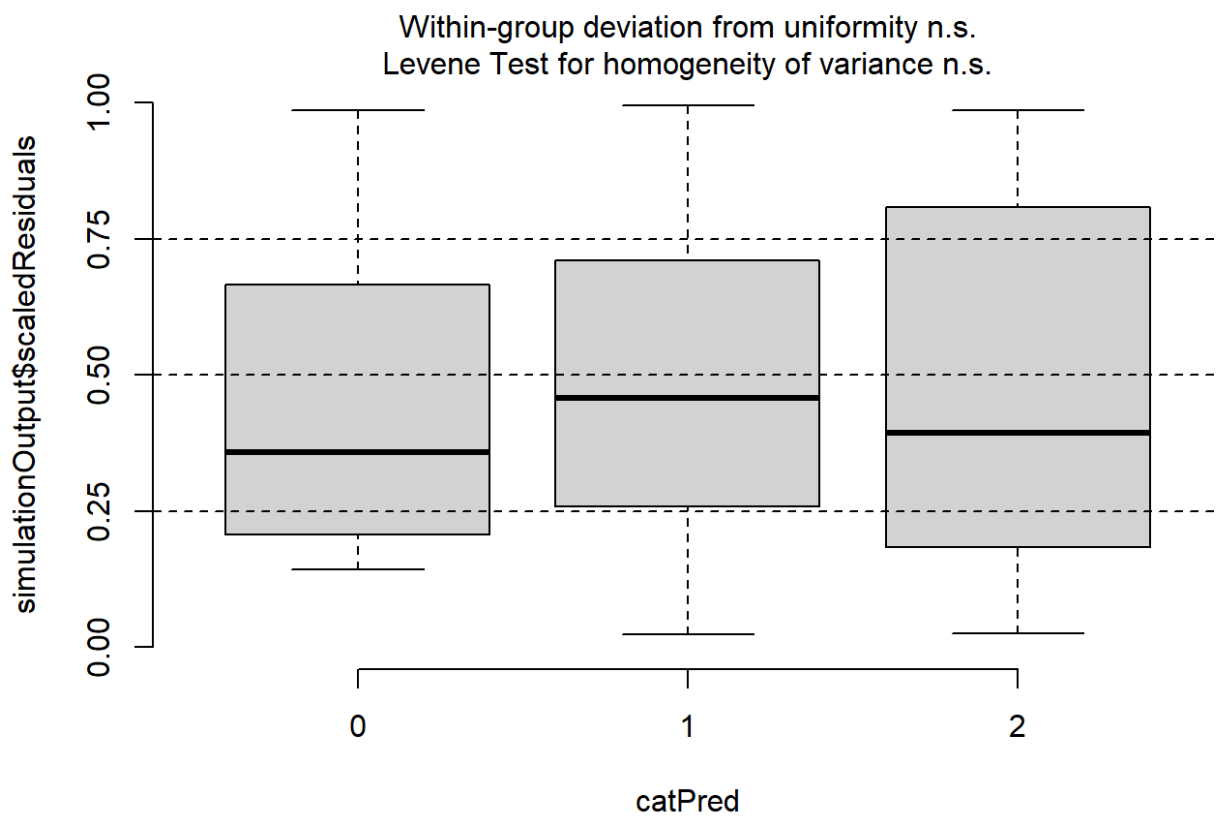
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 1.0083, p-value = 0.904
## alternative hypothesis: two.sided
```

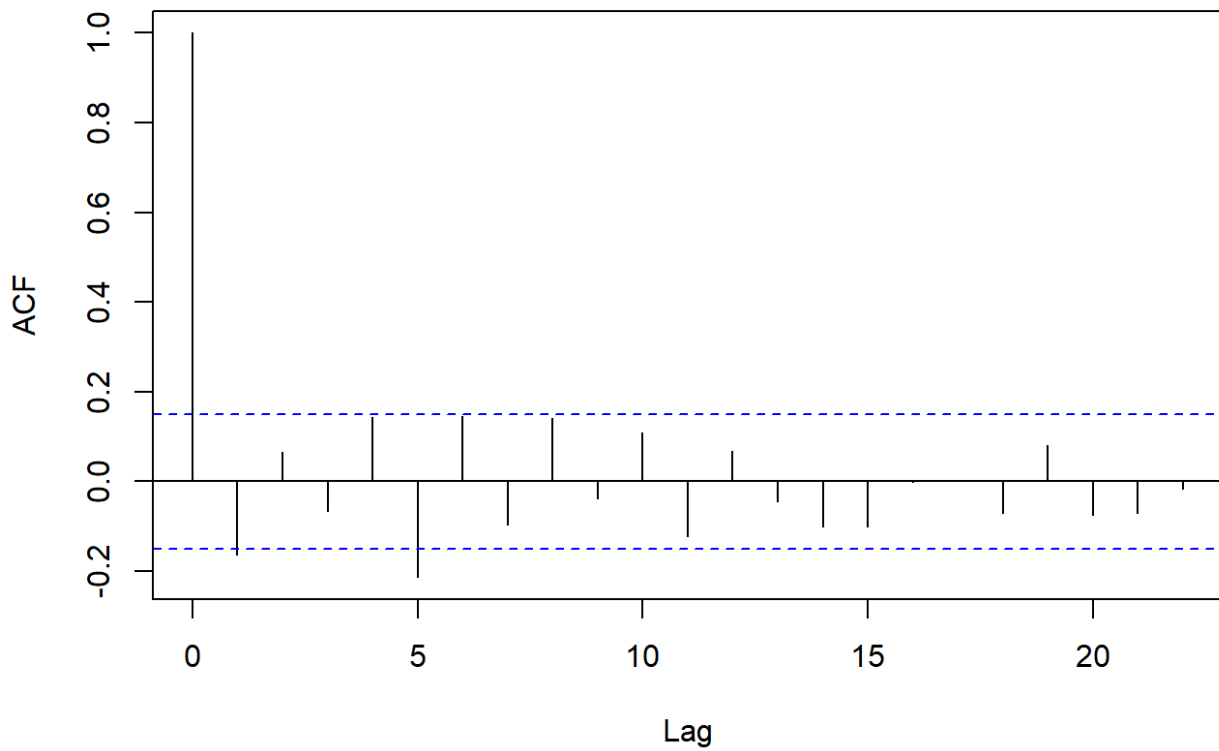
```
testQuantiles(sim_res, db_cc$t)
```



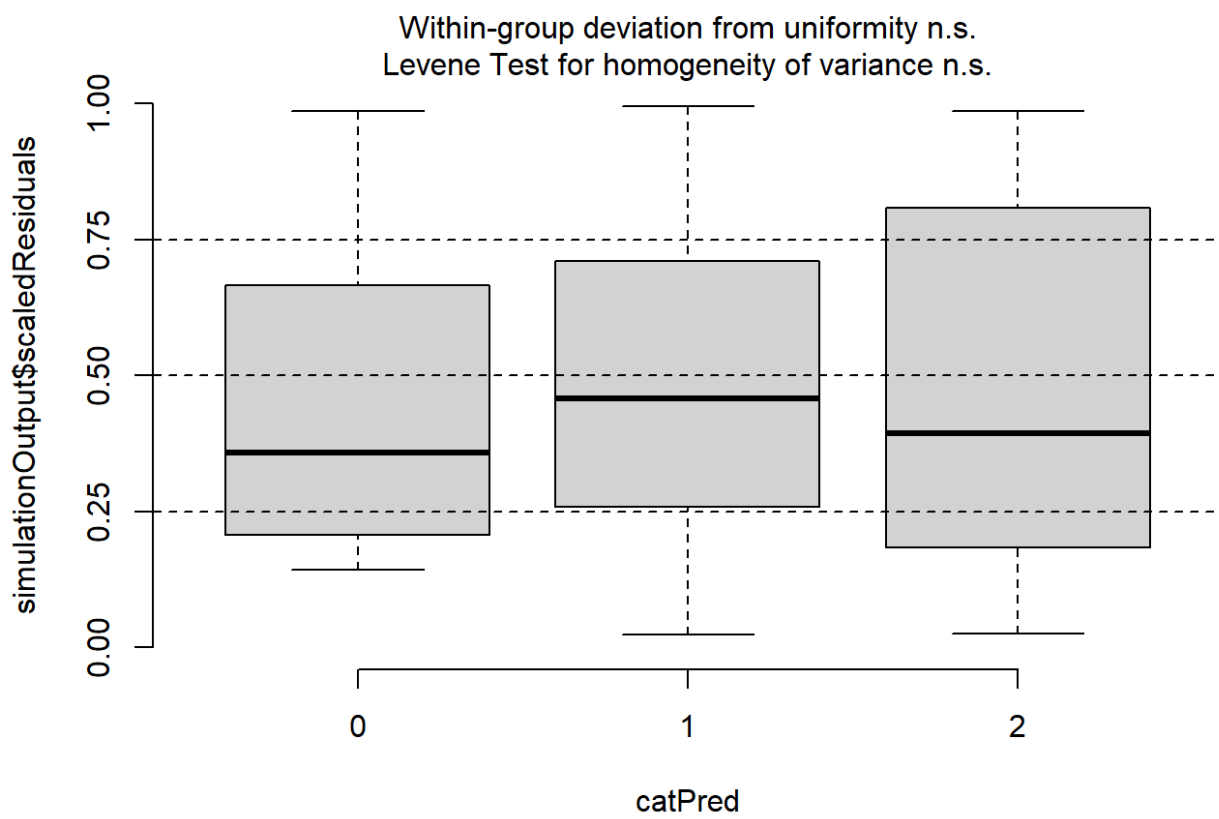
```
## NULL
```

```
acf(residuals(il1aModel_hetero))
```

Series residuals(il1aModel_hetero)



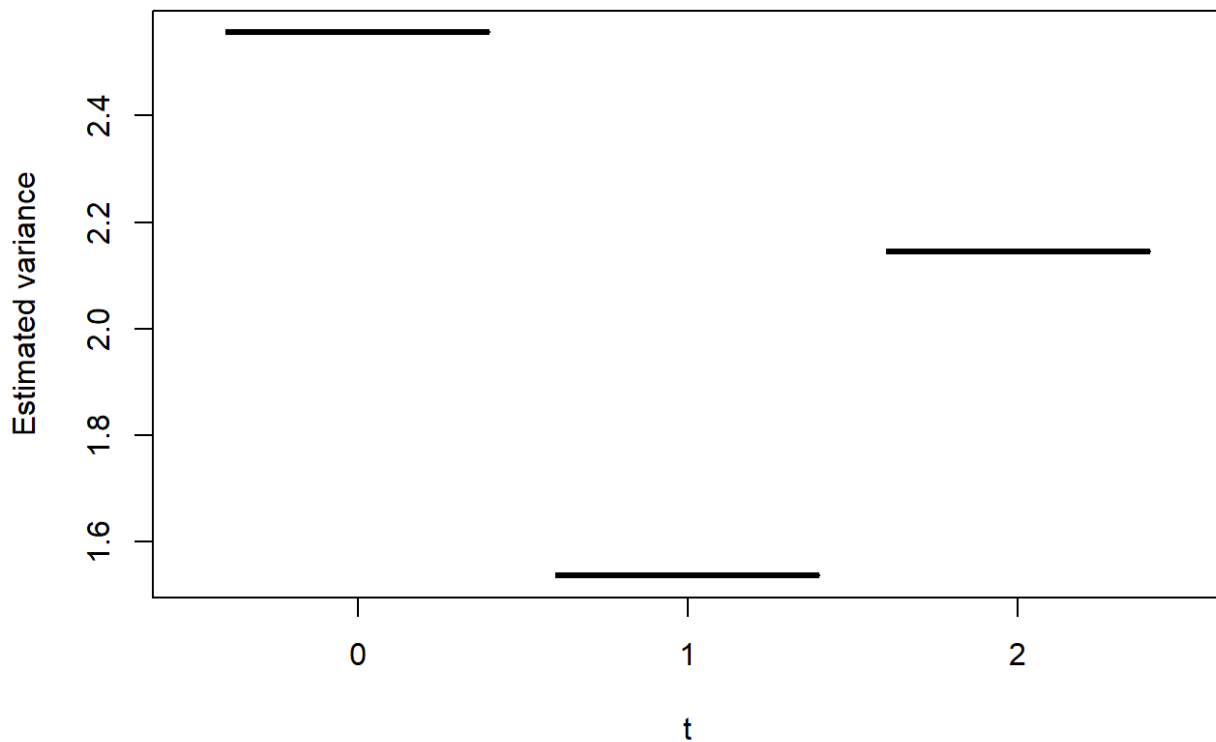
```
plotResiduals(sim_res, db_cc$t)
```




```

pred_disp <- predict(il1aModel_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")

```

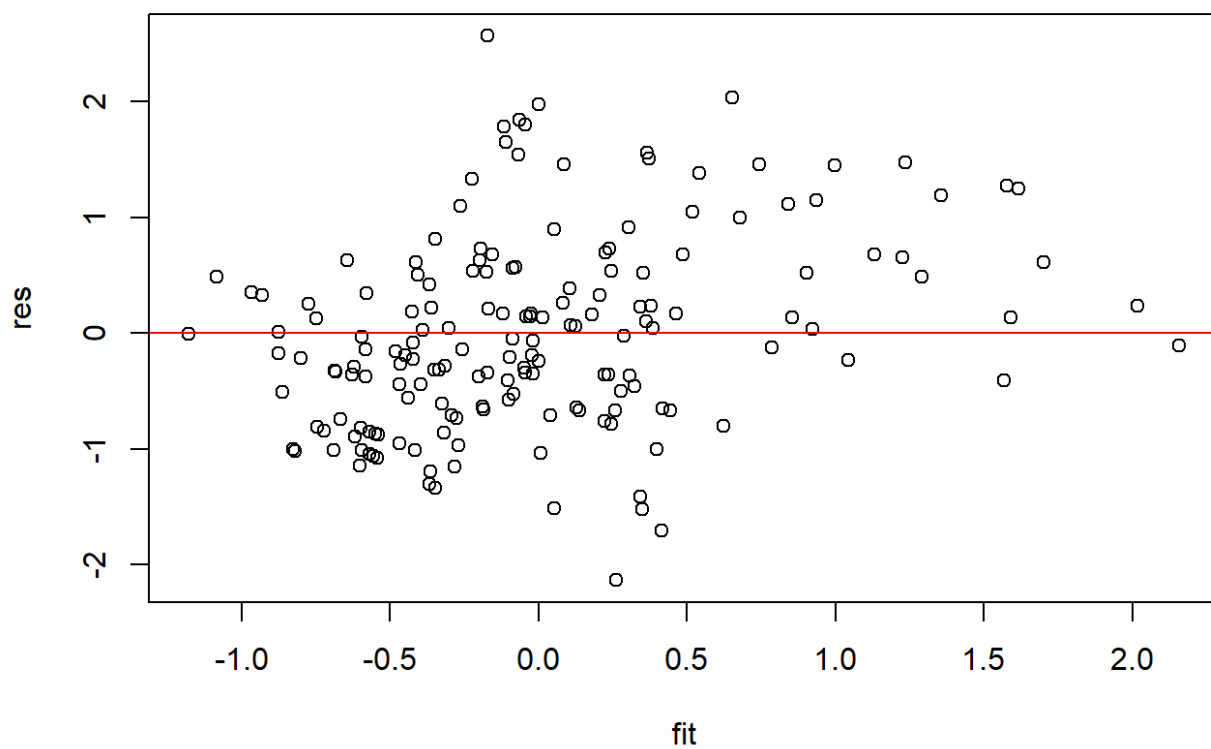


```

res <- residuals(il1aModel_hetero, type="pearson")
fit <- fitted(il1aModel_hetero)

plot(fit, res)
abline(h=0, col="red")

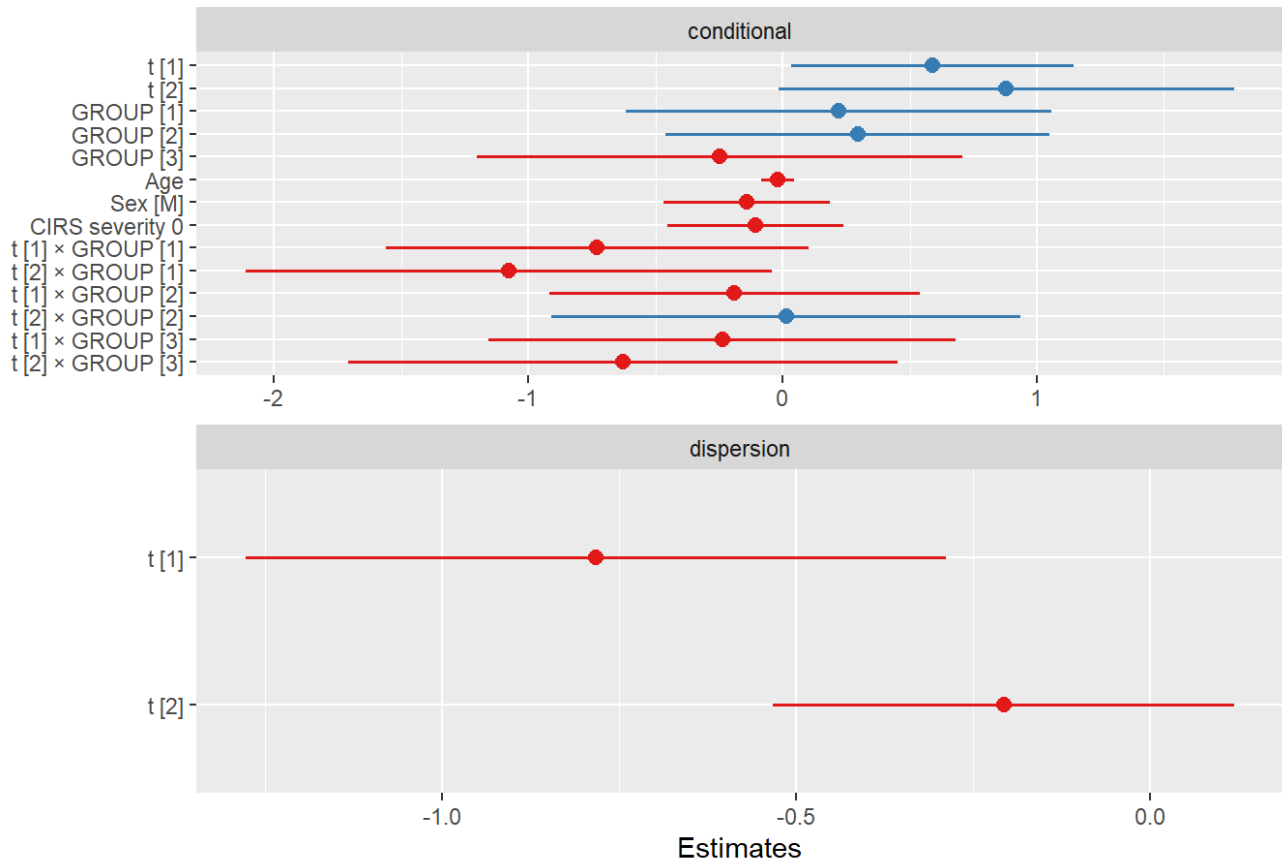
```



10.1 Visualizzazione del modello

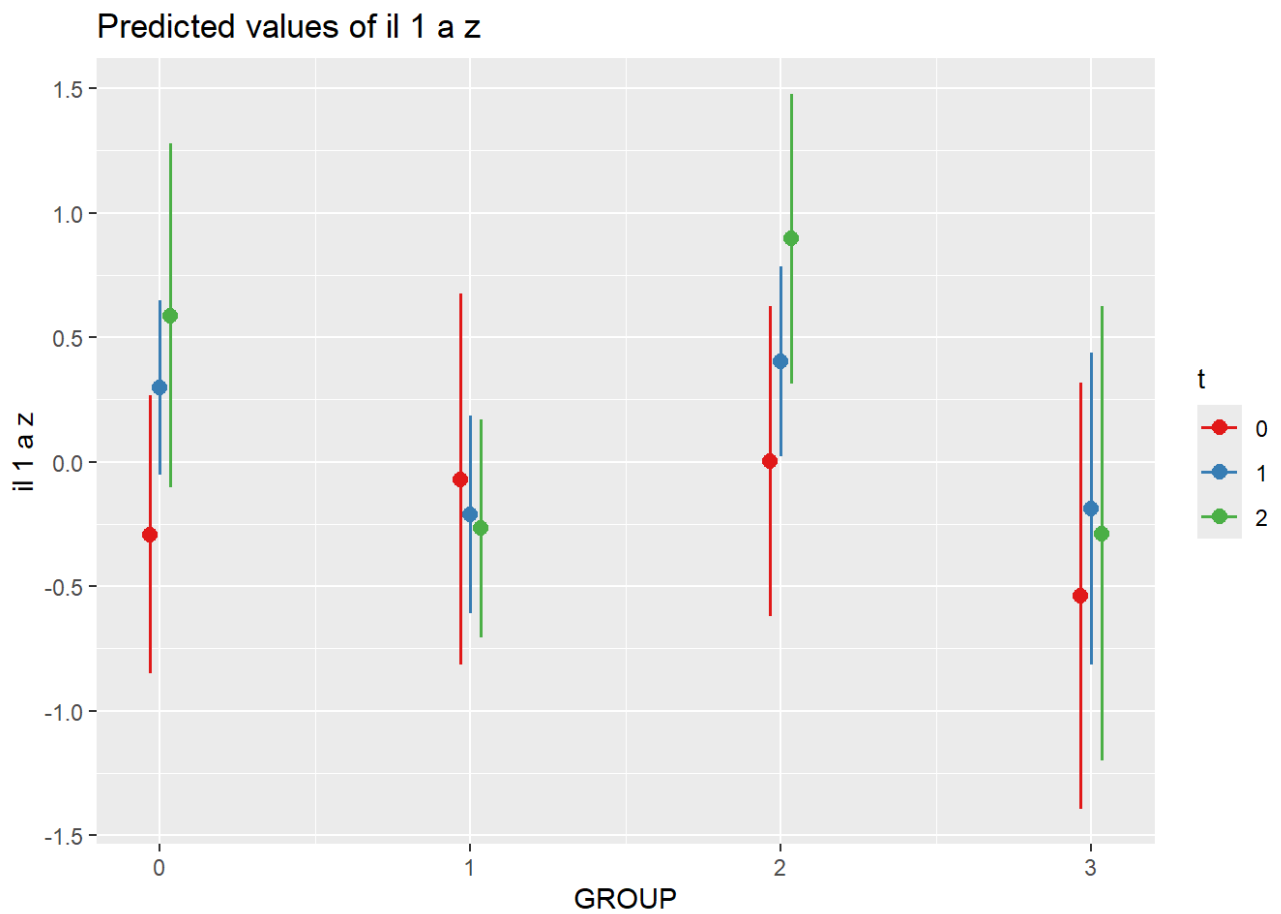
```
plot_model(i11aModel_hetero)
```

il 1 a z



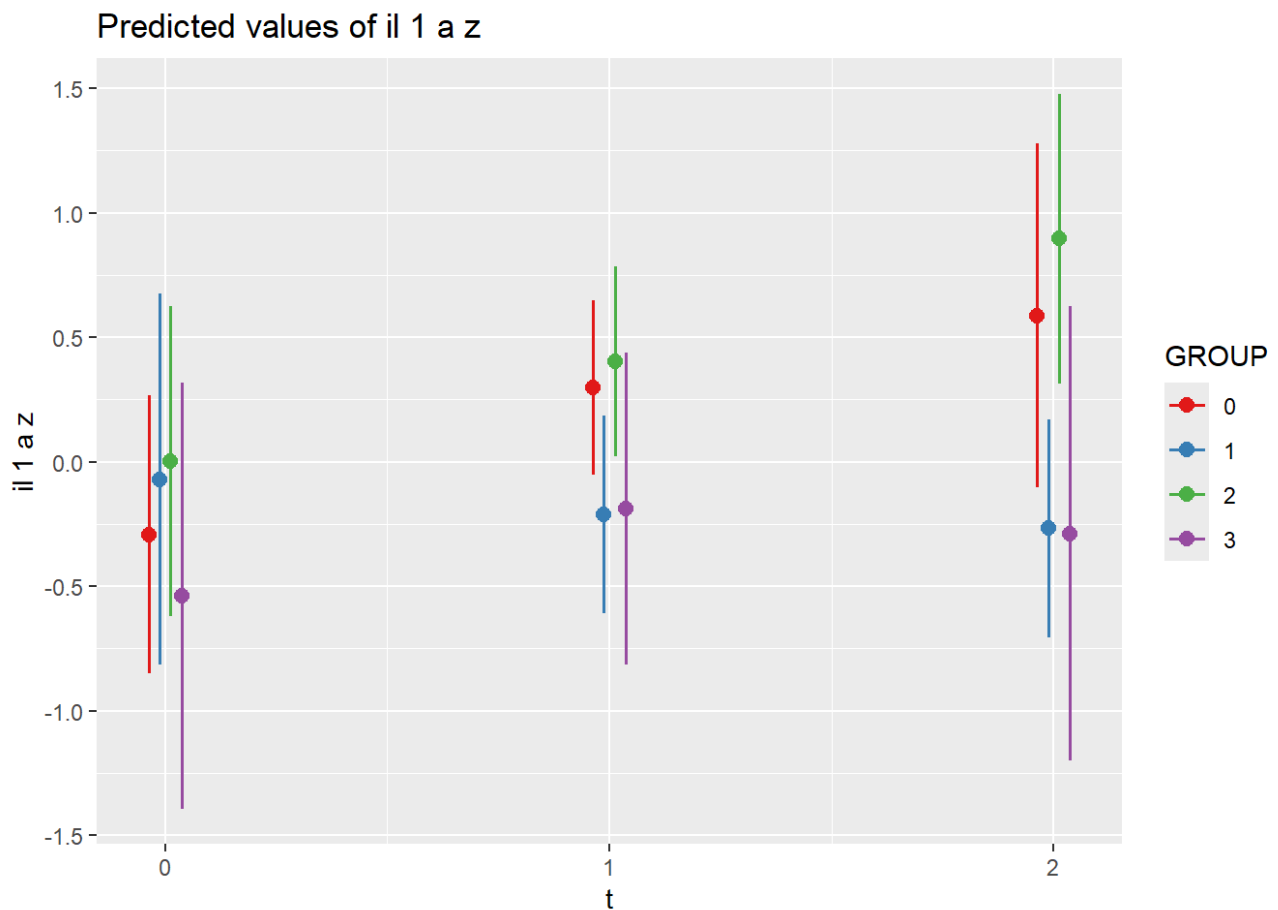
```
plot_model(il1aModel_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```



```
plot_model(il1aModel_hetero, type = "pred",  
           terms = c("t", "GROUP"),  
           pred.type = "fe")
```

```
## Ignoring unknown labels:  
## • linetype : "GROUP"  
## • shape : "GROUP"
```

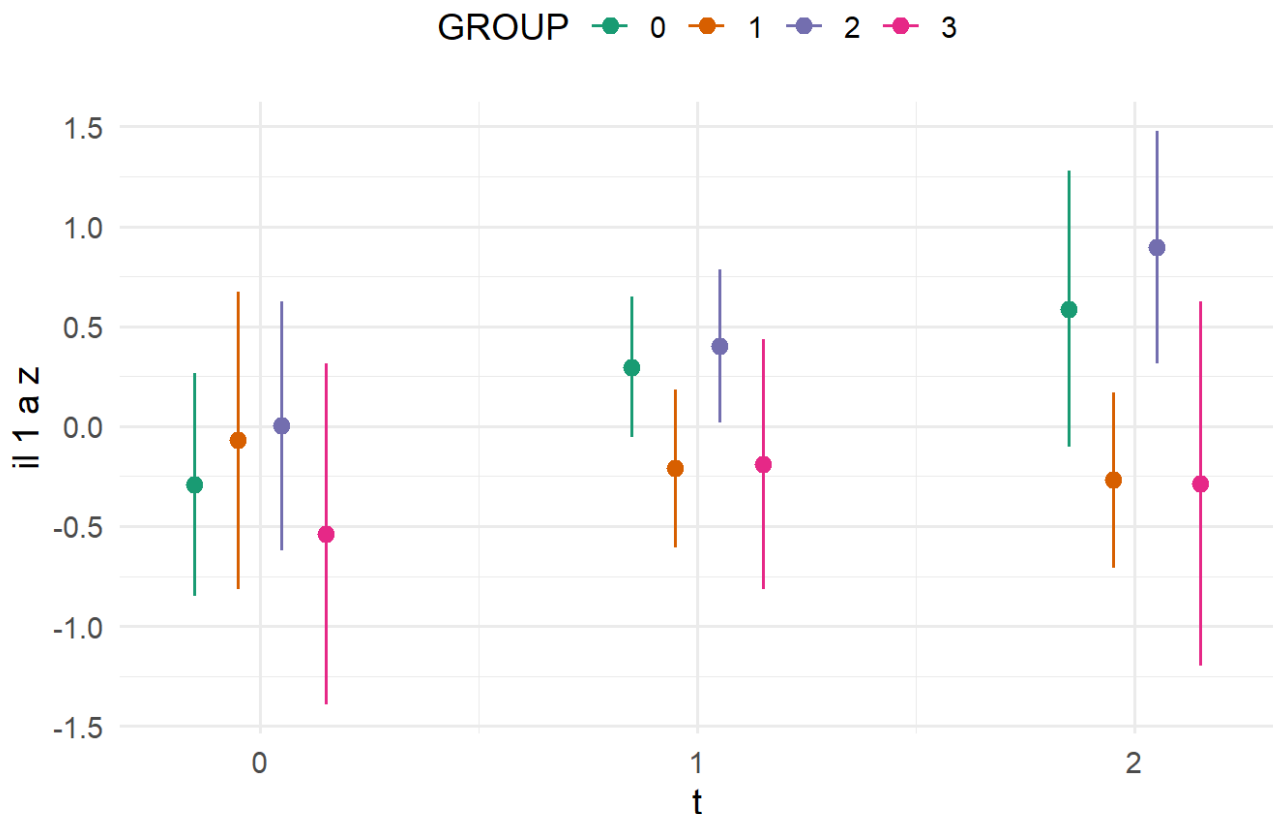


```
plot_model(il1aModel_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

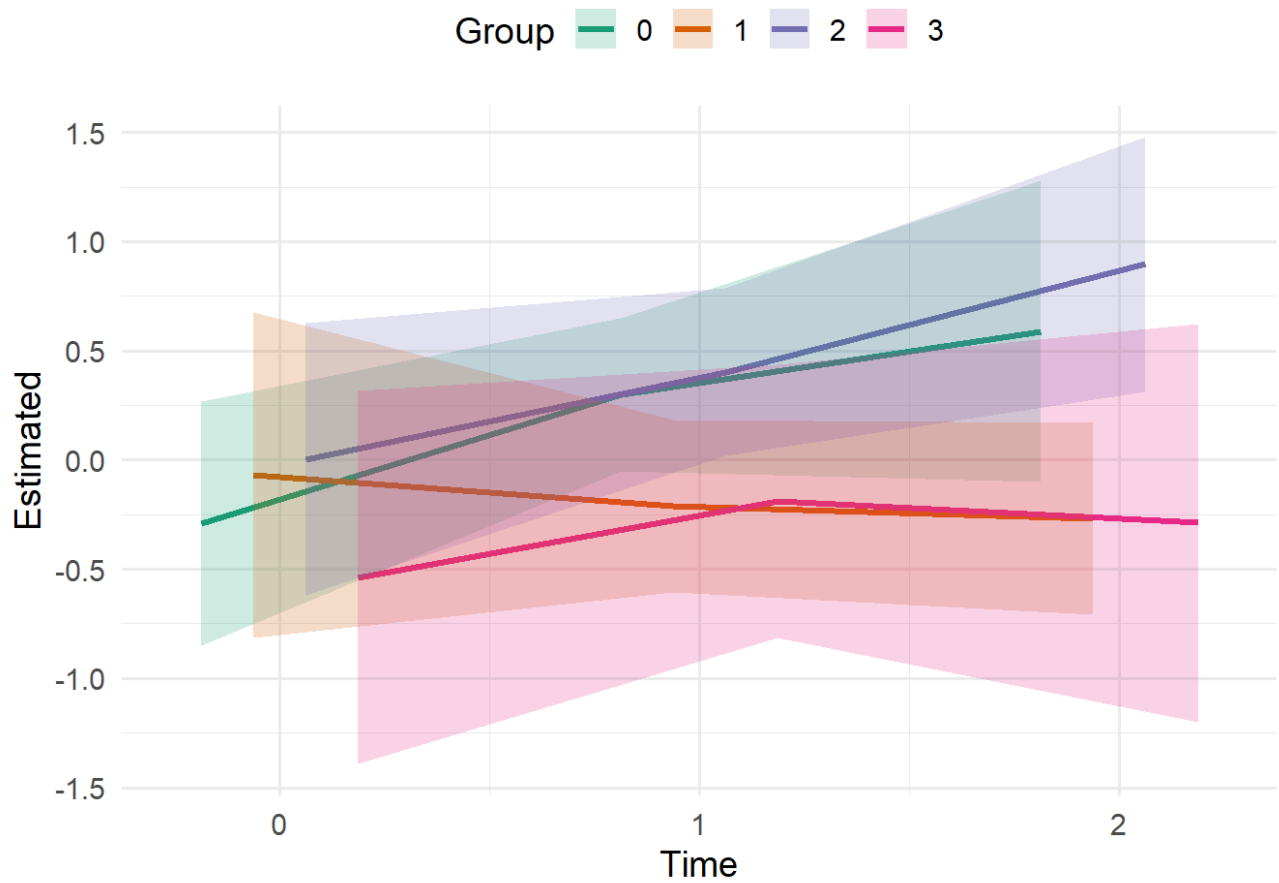
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of il 1 a z



```
# Personalizzazione per distanziare visivamente le linee (separazione per g
ruppo)
pred <- ggpredict(il1aModel_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = grou
p)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



10.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(il1aModel_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.3630 0.271 Inf   -0.8947    0.169
##  1      -0.1421 0.381 Inf   -0.8880    0.604
##  2      -0.0684 0.316 Inf   -0.6872    0.550
##  3      -0.6099 0.421 Inf   -1.4356    0.216
##
## t = 1:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.2262 0.154 Inf   -0.0762    0.529
##  1      -0.2826 0.196 Inf   -0.6660    0.101
##  2       0.3316 0.192 Inf   -0.0454    0.709
##  3      -0.2589 0.299 Inf   -0.8445    0.327
##
## t = 2:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.5167 0.340 Inf   -0.1501    1.184
##  1      -0.3391 0.217 Inf   -0.7651    0.087
##  2       0.8243 0.291 Inf    0.2537    1.395
##  3      -0.3593 0.449 Inf   -1.2403    0.522
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(ill1aModel_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  contrast          estimate    SE  df z.ratio p.value
##  GROUP0 - GROUP1      0.381 0.270 Inf   1.410  0.4933
##  GROUP0 - GROUP2     -0.236 0.244 Inf  -0.965  0.7695
##  GROUP0 - GROUP3      0.536 0.313 Inf   1.714  0.3164
##  GROUP1 - GROUP2     -0.617 0.263 Inf  -2.344  0.0883
##  GROUP1 - GROUP3      0.155 0.368 Inf   0.420  0.9750
##  GROUP2 - GROUP3      0.772 0.341 Inf   2.267  0.1059
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(ill1aModel_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```



```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.2209 0.427 Inf  -0.518 0.9549
## GROUP0 - GROUP2 -0.2946 0.385 Inf  -0.766 0.8700
## GROUP0 - GROUP3  0.2469 0.487 Inf   0.507 0.9574
## GROUP1 - GROUP2 -0.0736 0.444 Inf  -0.166 0.9984
## GROUP1 - GROUP3  0.4678 0.549 Inf   0.851 0.8297
## GROUP2 - GROUP3  0.5415 0.513 Inf   1.056 0.7160
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.5087 0.265 Inf   1.916 0.2211
## GROUP0 - GROUP2 -0.1054 0.234 Inf  -0.450 0.9697
## GROUP0 - GROUP3  0.4850 0.319 Inf   1.521 0.4251
## GROUP1 - GROUP2 -0.6141 0.270 Inf  -2.276 0.1035
## GROUP1 - GROUP3 -0.0237 0.381 Inf  -0.062 0.9999
## GROUP2 - GROUP3  0.5904 0.351 Inf   1.684 0.3320
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.8558 0.409 Inf   2.093 0.1552
## GROUP0 - GROUP2 -0.3075 0.372 Inf  -0.827 0.8419
## GROUP0 - GROUP3  0.8761 0.430 Inf   2.037 0.1745
## GROUP1 - GROUP2 -1.1634 0.360 Inf  -3.229 0.0068
## GROUP1 - GROUP3  0.0202 0.505 Inf   0.040 1.0000
## GROUP2 - GROUP3  1.1836 0.469 Inf   2.522 0.0566
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(ill1aModel_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.5891 0.284 Inf  -2.078  0.0944
## t0 - t2      -0.8797 0.457 Inf  -1.924  0.1318
## t1 - t2      -0.2906 0.343 Inf  -0.848  0.6731
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1       0.1405 0.343 Inf   0.409  0.9118
## t0 - t2       0.1970 0.417 Inf   0.473  0.8841
## t1 - t2       0.0565 0.242 Inf   0.234  0.9704
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.4000 0.310 Inf  -1.292  0.3999
## t0 - t2      -0.8927 0.428 Inf  -2.084  0.0932
## t1 - t2      -0.4927 0.297 Inf  -1.657  0.2220
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.3510 0.424 Inf  -0.828  0.6856
## t0 - t2      -0.2506 0.565 Inf  -0.443  0.8974
## t1 - t2       0.1005 0.427 Inf   0.235  0.9699
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

11 Model il1b

```
bptest(lm(il1b ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(il1b ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 24.216, df = 14, p-value = 0.04314
```

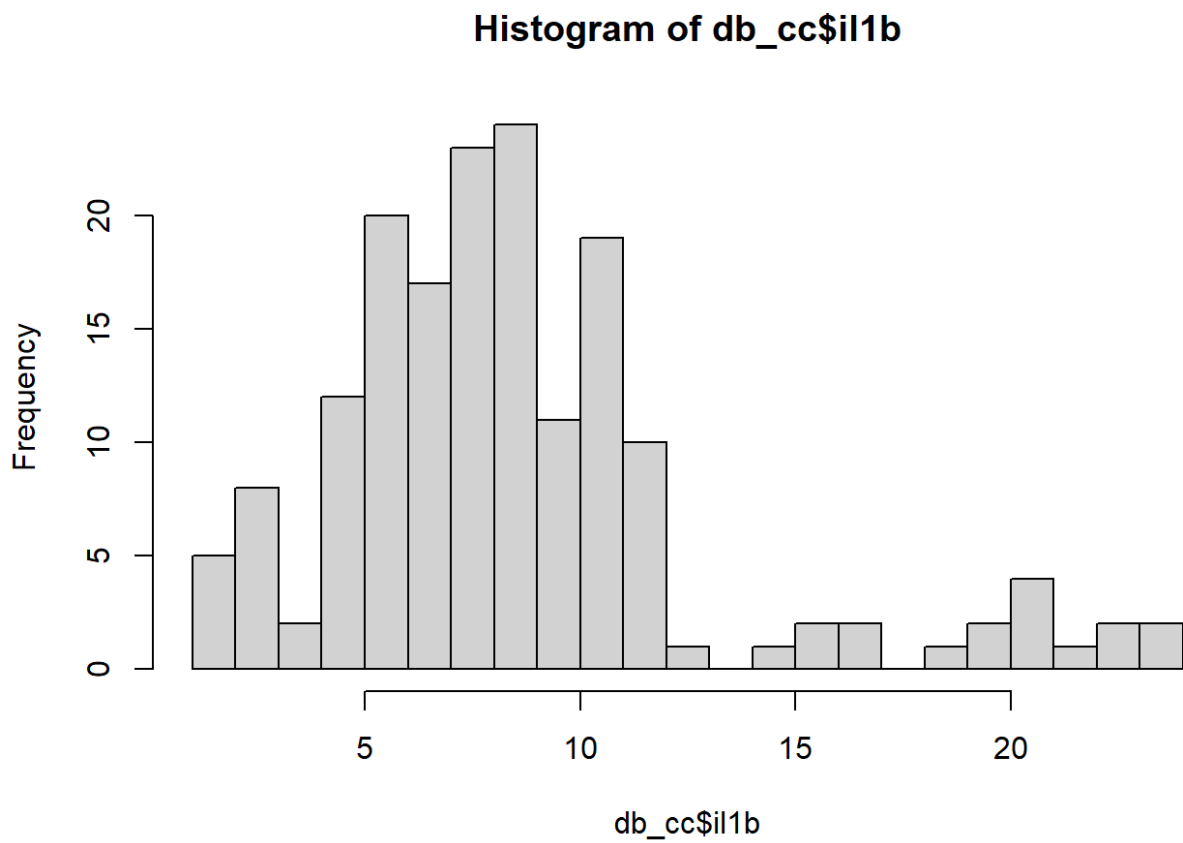
```
vars <- c("il1b","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

nrow(db); nrow(db_cc)
```

```
## [1] 243
```

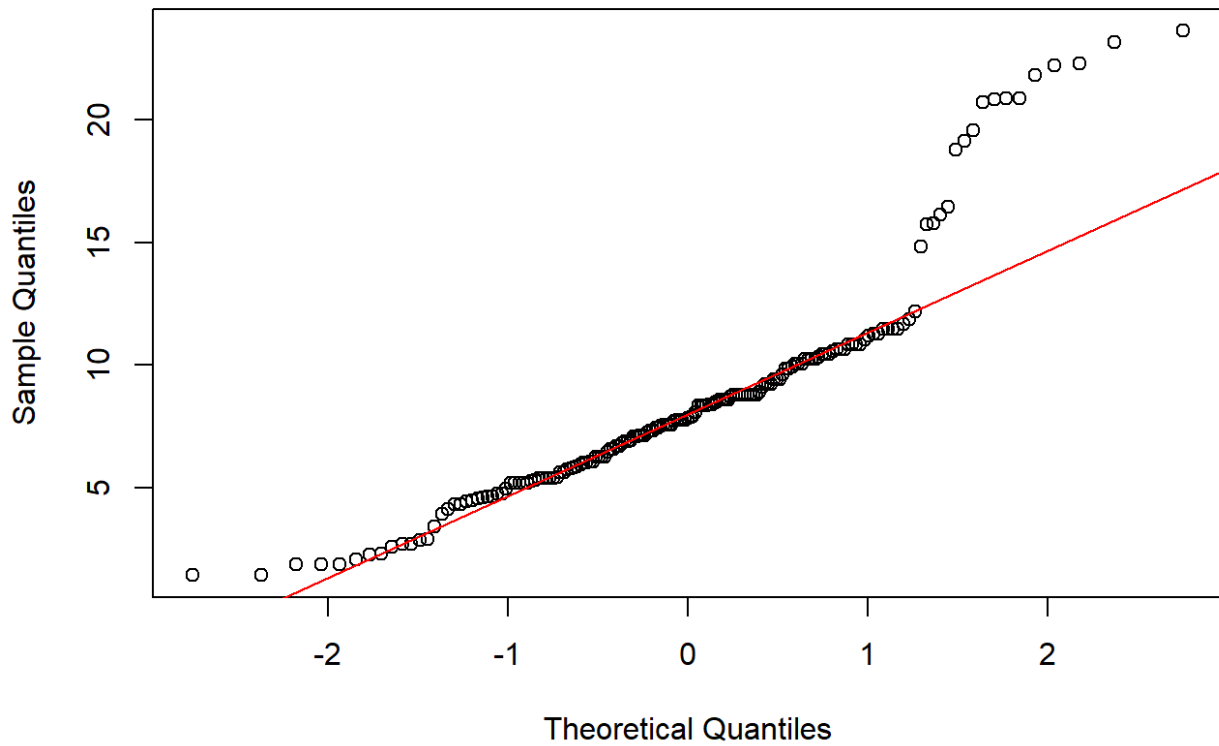
```
## [1] 169
```

```
hist(db_cc$il1b, breaks=30)
```



```
qqnorm(db_cc$il1b); qqline(db_cc$il1b, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$il1b)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  1.441   5.748   7.860   8.592  10.252   23.629
```

```
min(db_cc$il1b, na.rm=TRUE)
```

```
## [1] 1.441301
```

```
db$il1b_z <- scale(log(db$il1b))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$il1b_z <- scale(log(db_cc$il1b))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

il1bModel_hetero <- glmmTMB(
  il1b_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(il1bModel_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## il1b_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion:                ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    439.5    499.0    -200.8    401.5      150
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id       (Intercept) 0.1227   0.3503
##   Residual              NA       NA
## Number of obs: 169, groups: id, 78
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.502296   1.958495   1.278 0.201368
## t1              0.848321   0.252891   3.354 0.000795 ***
## t2              1.004903   0.474559   2.118 0.034213 *
## GROUP1         -0.121946   0.397608  -0.307 0.759072
## GROUP2          0.459418   0.346441   1.326 0.184804
## GROUP3          0.187161   0.434112   0.431 0.666370
## Age            -0.038756   0.025038  -1.548 0.121645
## SexM           -0.224768   0.126034  -1.783 0.074522 .
## CIRS_severity_0 -0.001856   0.183976  -0.010 0.991953
## t1:GROUP1       0.088234   0.410871   0.215 0.829964
## t2:GROUP1      -0.687956   0.601446  -1.144 0.252692
## t1:GROUP2      -0.128461   0.353539  -0.363 0.716338
## t2:GROUP2      -0.494415   0.558686  -0.885 0.376177
## t1:GROUP3       0.220405   0.447718   0.492 0.622518
## t2:GROUP3      -0.200862   0.658271  -0.305 0.760262
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.08947    0.11544  -0.775 0.43829
## t1          -0.84640    0.27189  -3.113 0.00185 **
## t2           0.19441    0.16581   1.173 0.24101
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

il1bModel_homo <- glmmTMB(
  il1b_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(il1bModel_homo, il1bModel_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## il1bModel_homo: il1b_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severi
ty_0 + , zi=~0, disp=~1
## il1bModel_homo:      (1 | id), zi=~0, disp=~t
## il1bModel_hetero: il1b_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## il1bModel_hetero:      (1 | id), zi=~0, disp=~t
##
##          Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Chi
sq)
## il1bModel_homo    17 472.35 525.56 -219.17   438.35
## il1bModel_hetero  19 439.51 498.98 -200.76   401.51 36.834      2 1.004e
-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

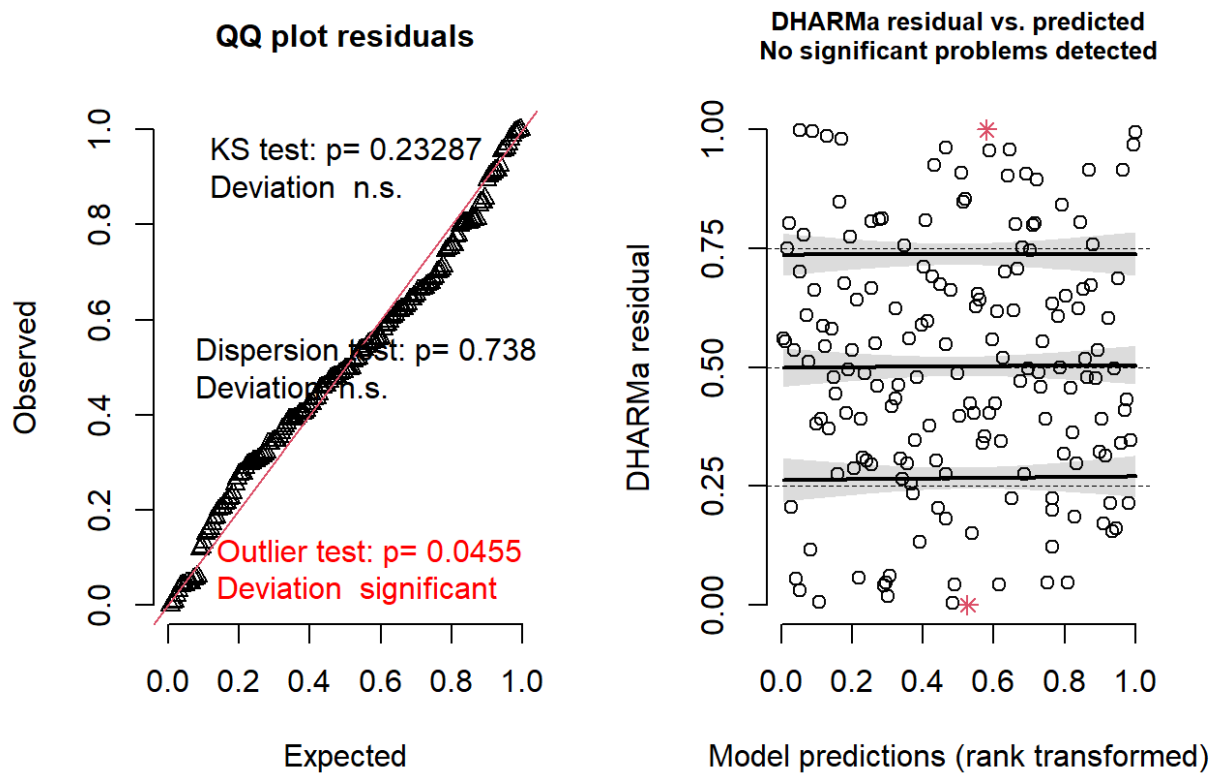
```

```

sim_res <- simulateResiduals(il1bModel_hetero, n = 1000)
plot(sim_res)

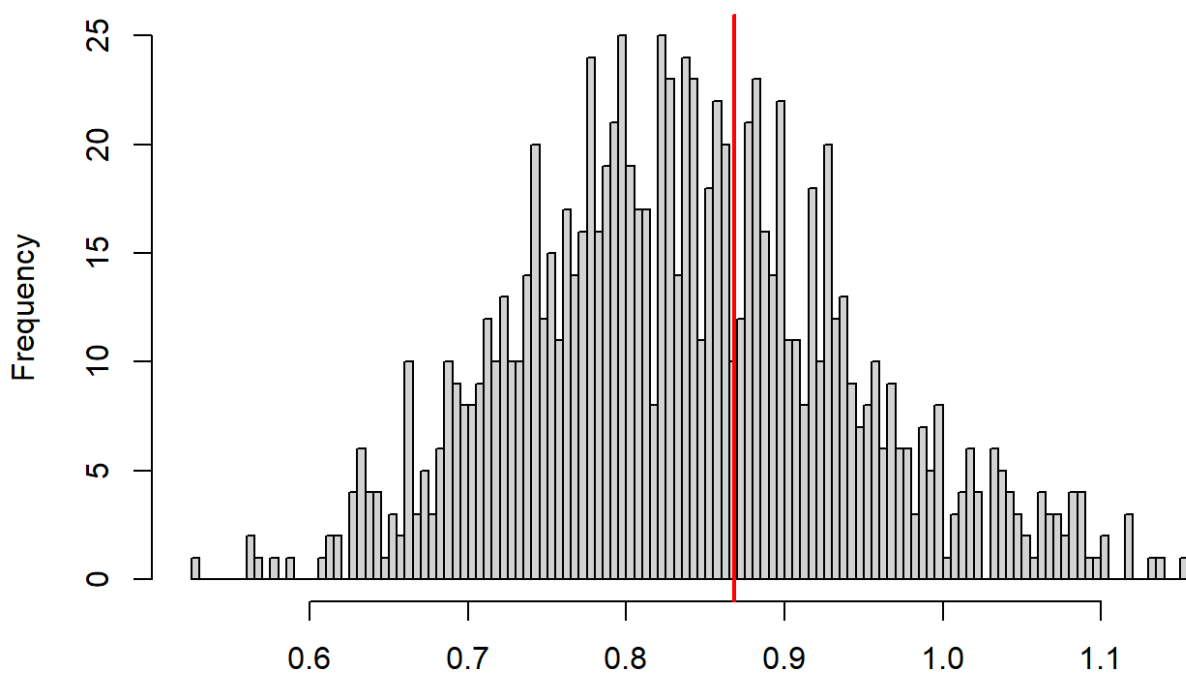
```

DHARMA residual



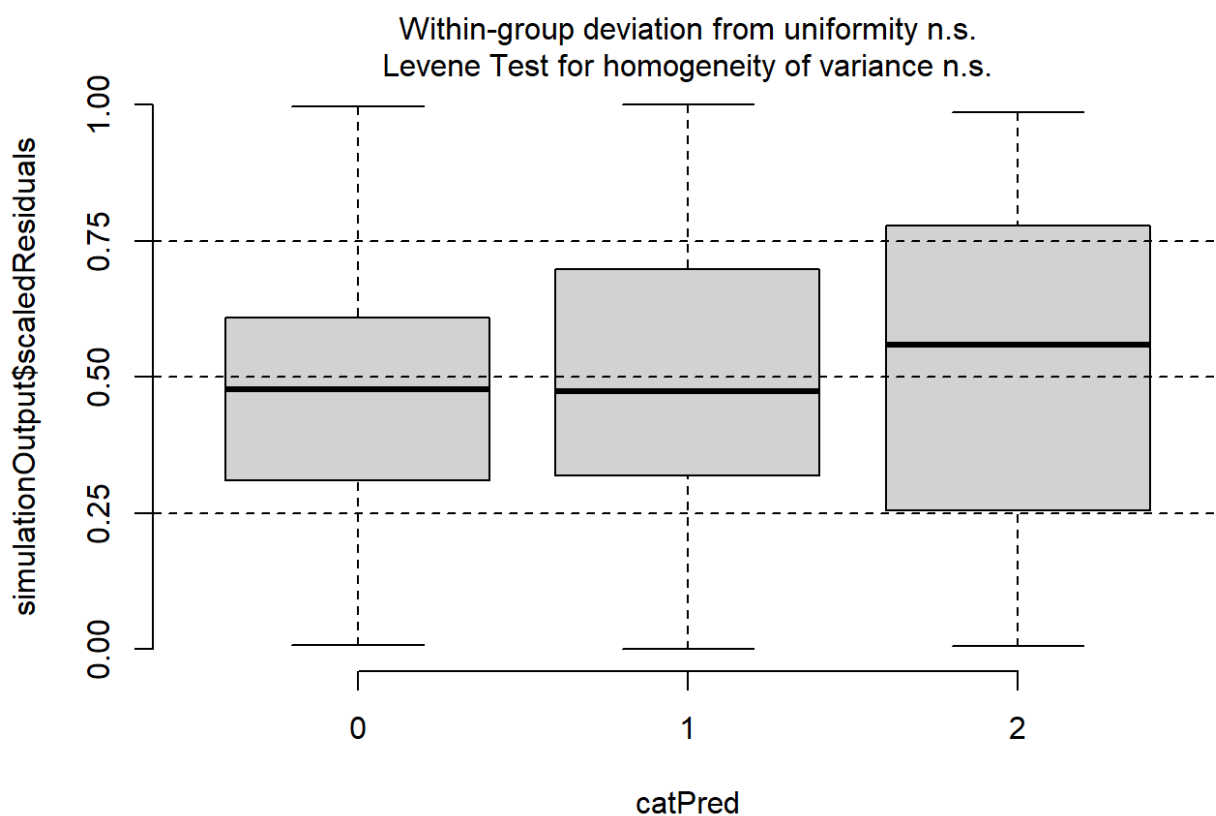
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated




```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 1.037, p-value = 0.738
## alternative hypothesis: two.sided
```

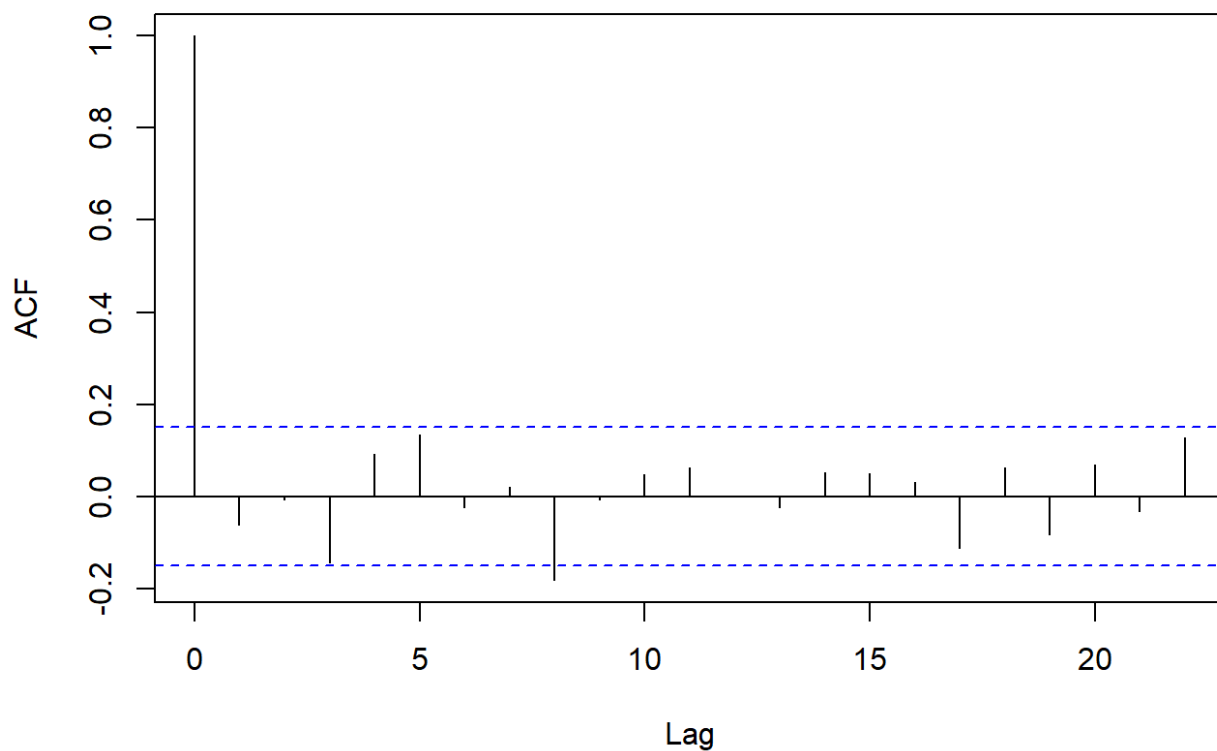
```
testQuantiles(sim_res, db_cc$t)
```



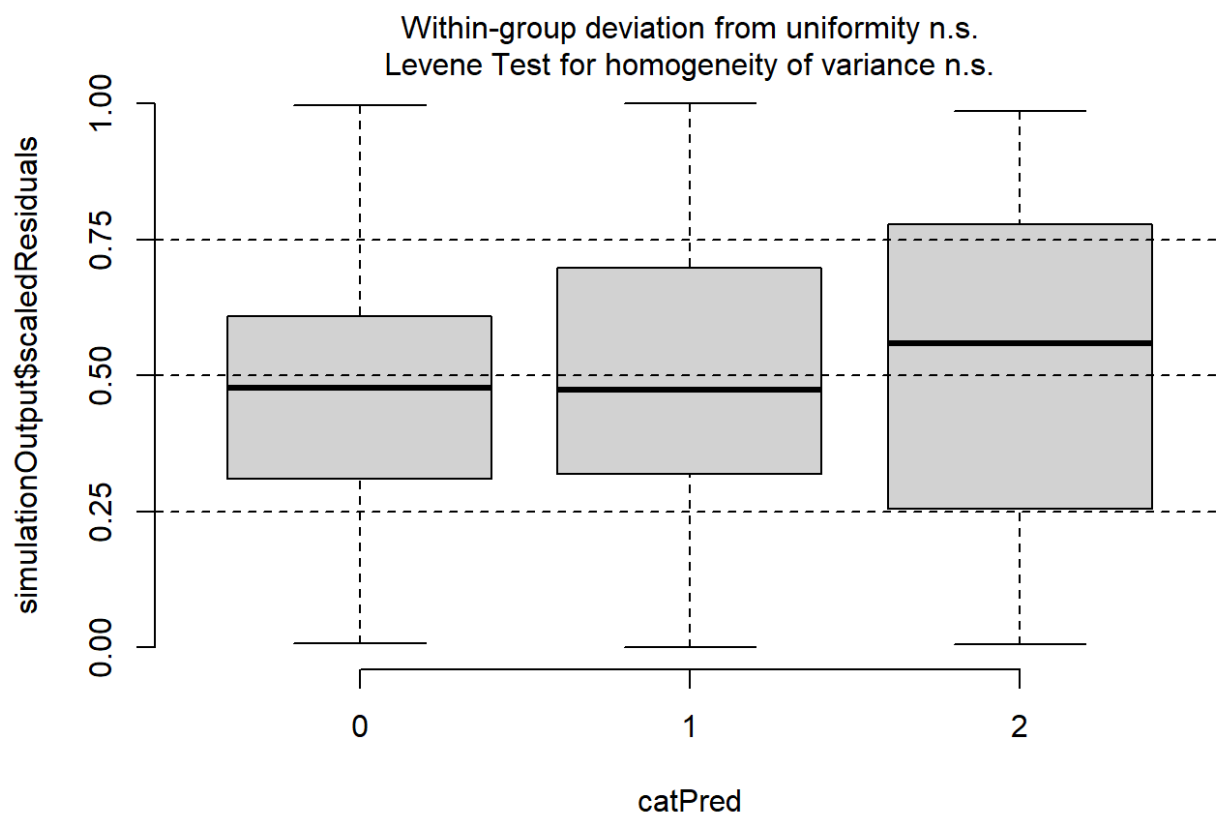
```
## NULL
```

```
acf(residuals(il1bModel_hetero))
```

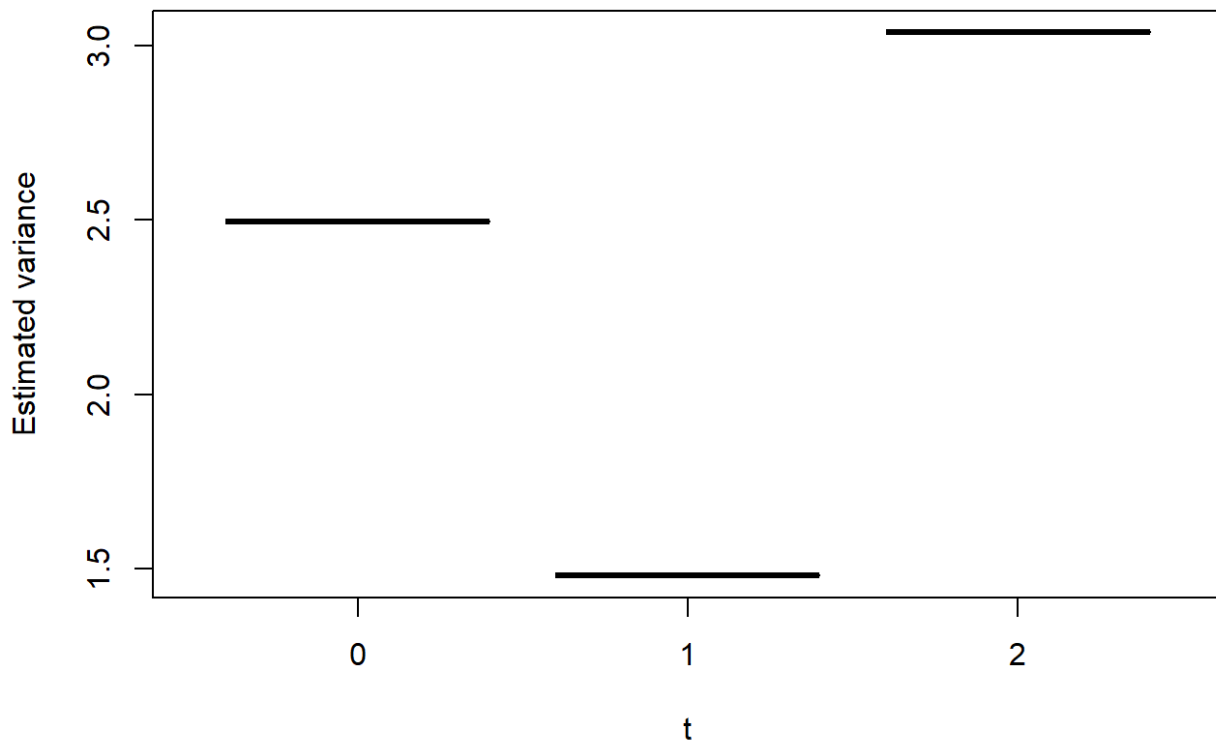
Series residuals(il1bModel_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

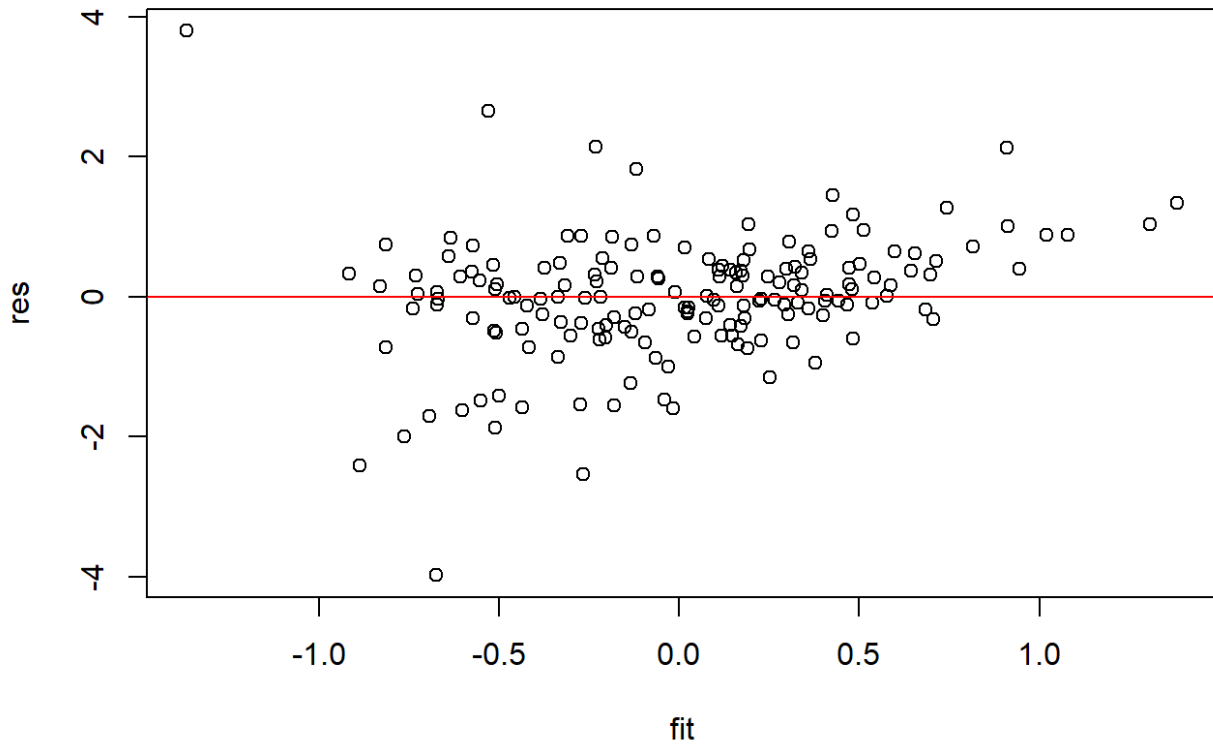


```
pred_disp <- predict(il1bModel_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(il1bModel_hetero, type="pearson")
fit <- fitted(il1bModel_hetero)

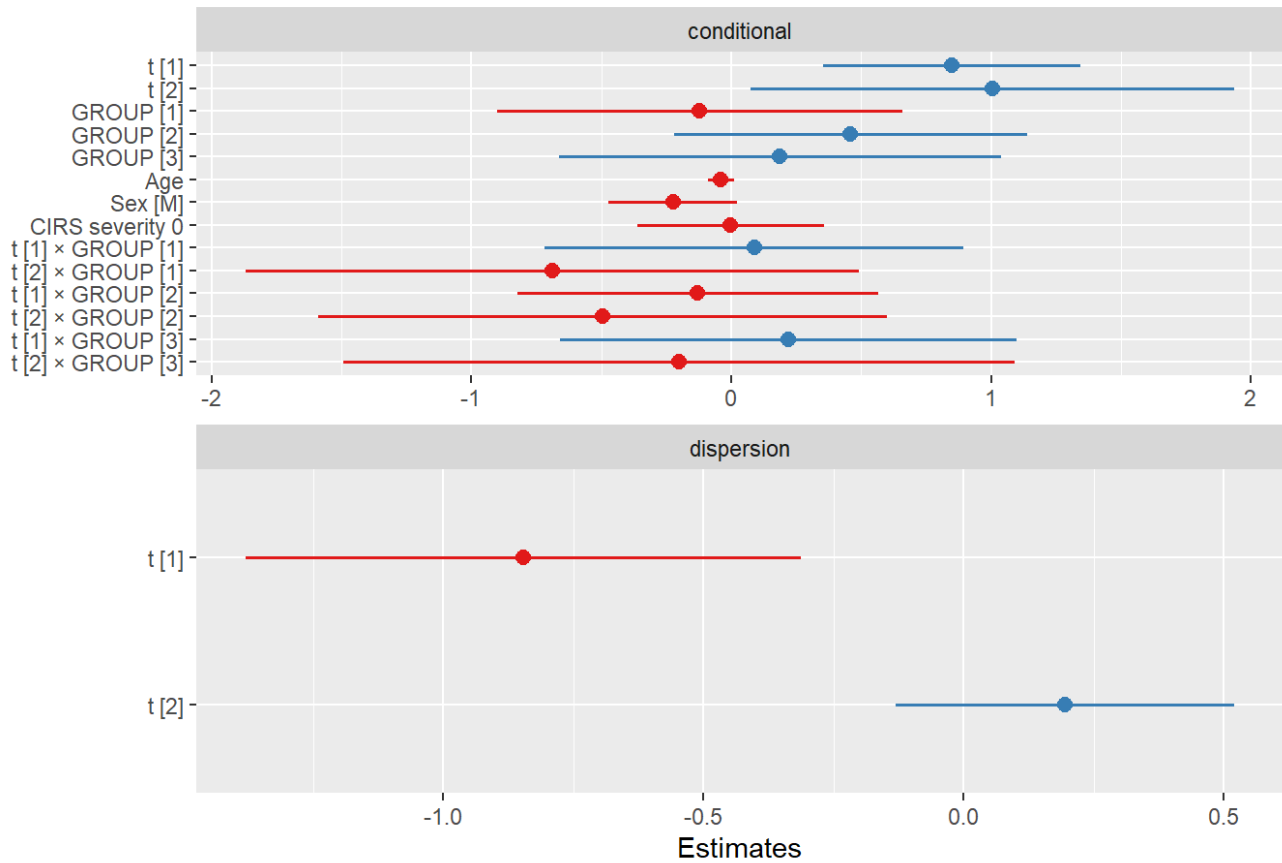
plot(fit, res)
abline(h=0, col="red")
```



11.1 Visualizzazione del modello

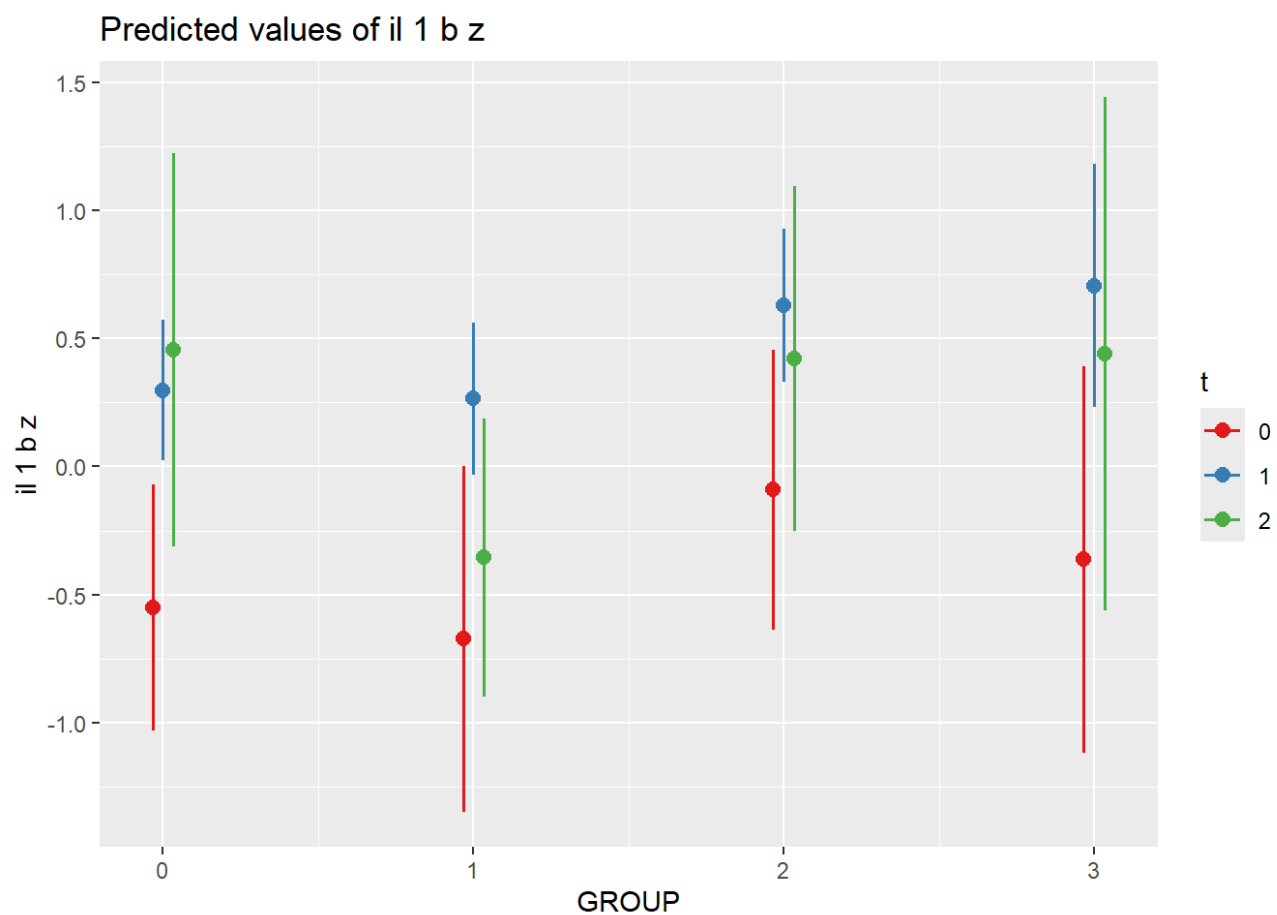
```
plot_model(il1bModel_hetero)
```

il 1 b z



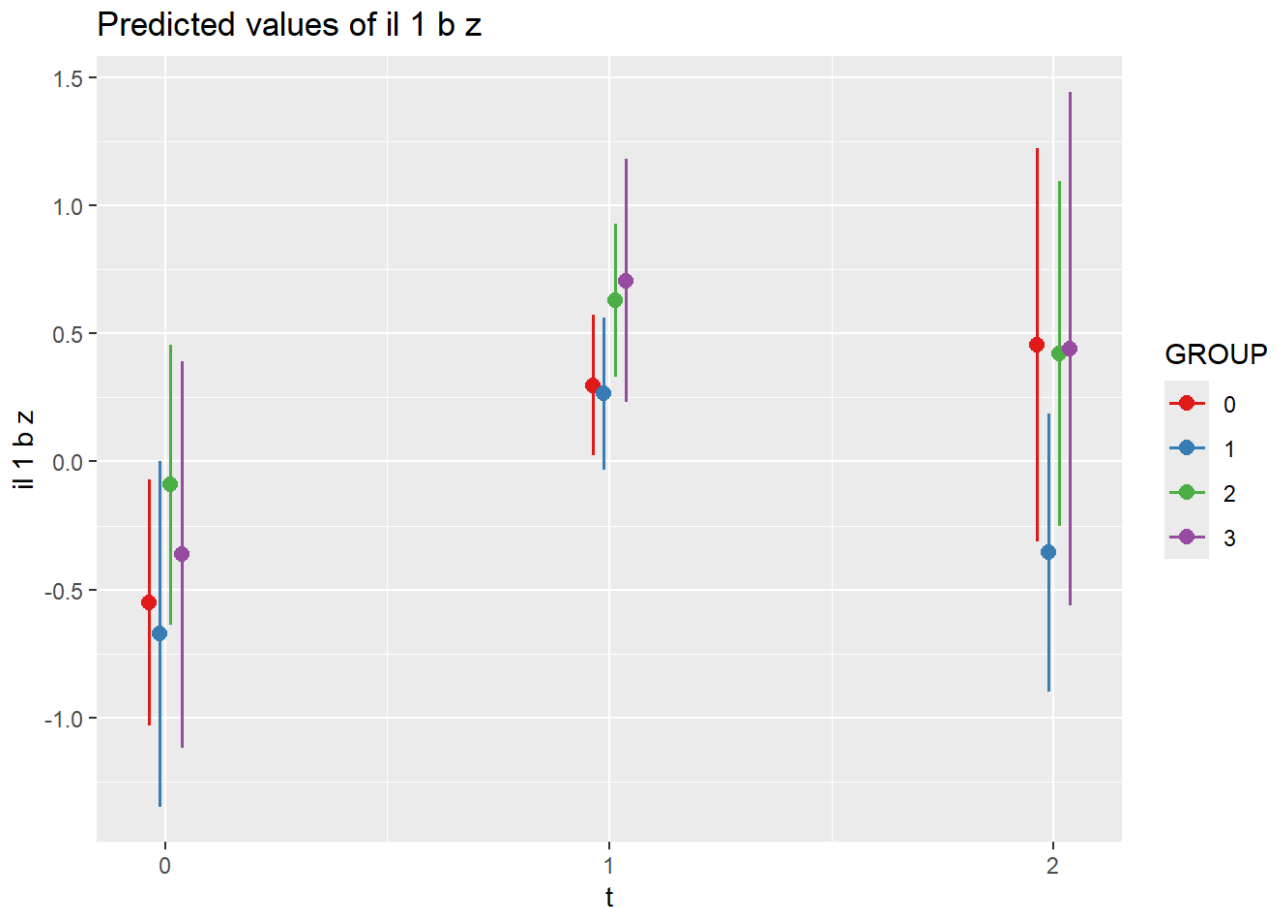
```
plot_model(il1bModel_hetero, type = "pred",
           terms = c("GROUP", "t"),
           pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```



```
plot_model(il1bModel_hetero, type = "pred",  
           terms = c("t", "GROUP"),  
           pred.type = "fe")
```

```
## Ignoring unknown labels:  
## • linetype : "GROUP"  
## • shape : "GROUP"
```

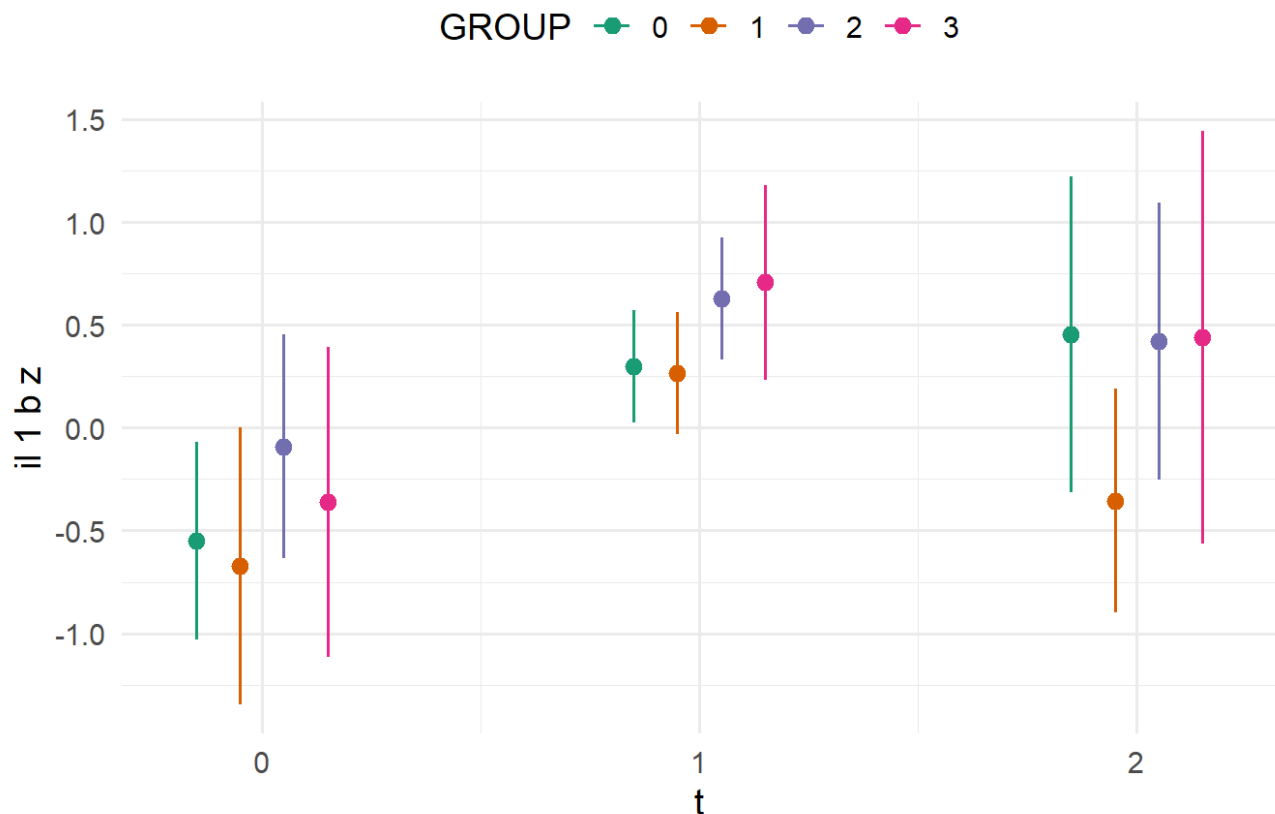


```
plot_model(il1bModel_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

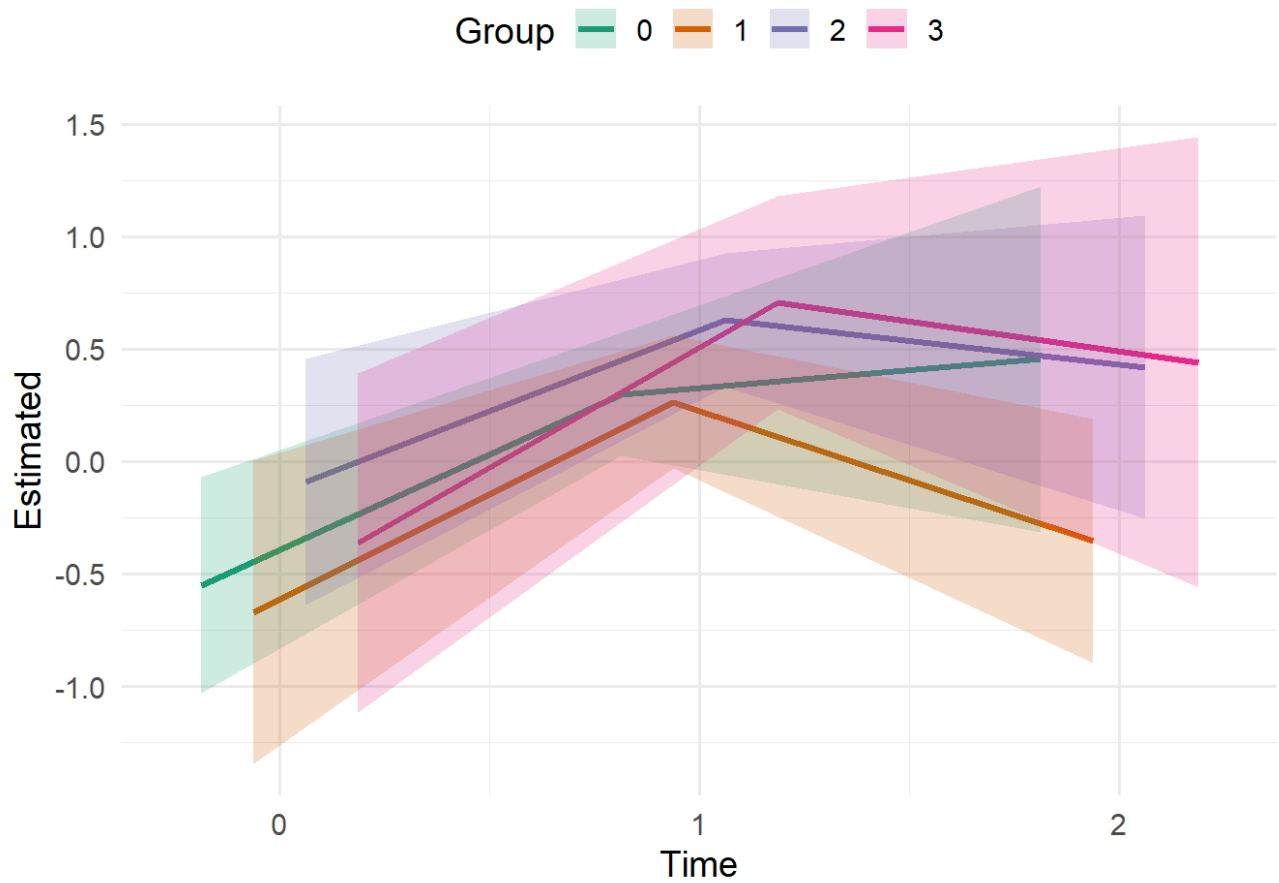
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of il 1 b z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(il1bModel_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```

11.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(il1bModel_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.662 0.236 Inf   -1.1243   -0.2003
##  1      -0.784 0.346 Inf   -1.4617   -0.1068
##  2      -0.203 0.278 Inf   -0.7473    0.3416
##  3      -0.475 0.375 Inf   -1.2107    0.2605
##
## t = 1:
##  GROUP emmean    SE  df asymp.LCL asymp.UCL
##  0       0.186 0.123 Inf   -0.0552    0.4273
##  1       0.152 0.147 Inf   -0.1354    0.4401
##  2       0.517 0.151 Inf    0.2214    0.8126
##  3       0.594 0.227 Inf    0.1477    1.0395
##
## t = 2:
##  GROUP emmean    SE  df asymp.LCL asymp.UCL
##  0       0.343 0.387 Inf   -0.4167    1.1019
##  1      -0.467 0.275 Inf   -1.0064    0.0718
##  2       0.308 0.340 Inf   -0.3595    0.9748
##  3       0.329 0.504 Inf   -0.6580    1.3159
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(illbModel_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  contrast          estimate    SE  df z.ratio p.value
##  GROUP0 - GROUP1    0.3219 0.242 Inf   1.331  0.5428
##  GROUP0 - GROUP2   -0.2518 0.223 Inf  -1.127  0.6729
##  GROUP0 - GROUP3   -0.1937 0.274 Inf  -0.707  0.8944
##  GROUP1 - GROUP2   -0.5736 0.234 Inf  -2.451  0.0678
##  GROUP1 - GROUP3   -0.5155 0.314 Inf  -1.641  0.3553
##  GROUP2 - GROUP3    0.0581 0.294 Inf   0.197  0.9973
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(illbModel_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1   0.1219 0.398 Inf   0.307  0.9900
## GROUP0 - GROUP2  -0.4594 0.346 Inf  -1.326  0.5462
## GROUP0 - GROUP3  -0.1872 0.434 Inf  -0.431  0.9731
## GROUP1 - GROUP2  -0.5814 0.416 Inf  -1.398  0.5004
## GROUP1 - GROUP3  -0.3091 0.500 Inf  -0.618  0.9264
## GROUP2 - GROUP3   0.2723 0.458 Inf   0.595  0.9336
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1   0.0337 0.199 Inf   0.169  0.9983
## GROUP0 - GROUP2  -0.3310 0.176 Inf  -1.883  0.2352
## GROUP0 - GROUP3  -0.4076 0.241 Inf  -1.695  0.3265
## GROUP1 - GROUP2  -0.3647 0.203 Inf  -1.793  0.2768
## GROUP1 - GROUP3  -0.4413 0.287 Inf  -1.535  0.4165
## GROUP2 - GROUP3  -0.0766 0.264 Inf  -0.291  0.9915
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1   0.8099 0.480 Inf   1.686  0.3309
## GROUP0 - GROUP2   0.0350 0.468 Inf   0.075  0.9998
## GROUP0 - GROUP3   0.0137 0.539 Inf   0.025  1.0000
## GROUP1 - GROUP2  -0.7749 0.438 Inf  -1.770  0.2876
## GROUP1 - GROUP3  -0.7962 0.579 Inf  -1.374  0.5157
## GROUP2 - GROUP3  -0.0213 0.566 Inf  -0.038  1.0000
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(illbModel_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.848 0.253 Inf  -3.354  0.0023
## t0 - t2      -1.005 0.475 Inf  -2.118  0.0863
## t1 - t2      -0.157 0.409 Inf   -0.383  0.9225
##
## GROUP = 1:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.937 0.333 Inf  -2.810  0.0137
## t0 - t2      -0.317 0.434 Inf  -0.730  0.7458
## t1 - t2       0.620 0.292 Inf   2.121  0.0856
##
## GROUP = 2:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.720 0.286 Inf  -2.521  0.0314
## t0 - t2      -0.510 0.447 Inf  -1.143  0.4875
## t1 - t2       0.209 0.359 Inf   0.583  0.8290
##
## GROUP = 3:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -1.069 0.399 Inf  -2.682  0.0200
## t0 - t2      -0.804 0.620 Inf  -1.297  0.3969
## t1 - t2       0.265 0.517 Inf   0.512  0.8655
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

12 Model il33

```
bptest(lm(il33 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(il33 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 33.875, df = 14, p-value = 0.002151
```

```
vars <- c("il33","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

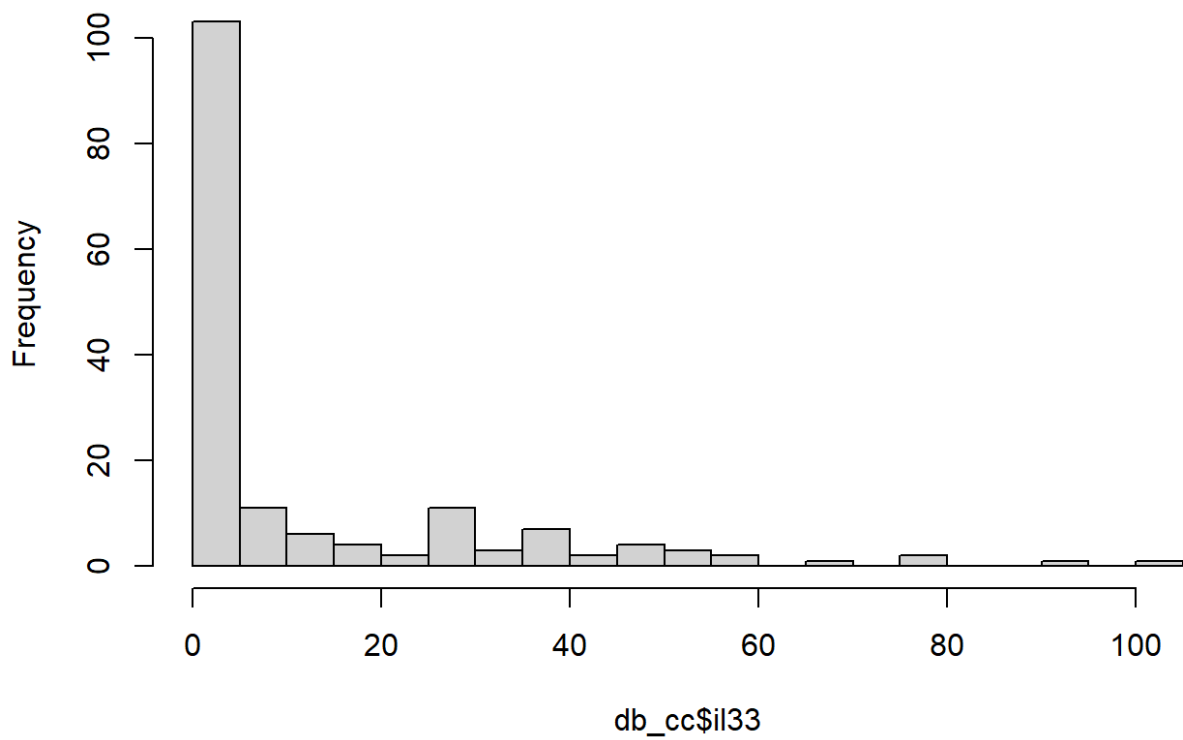
nrow(db); nrow(db_cc)
```

```
## [1] 243
```

```
## [1] 163
```

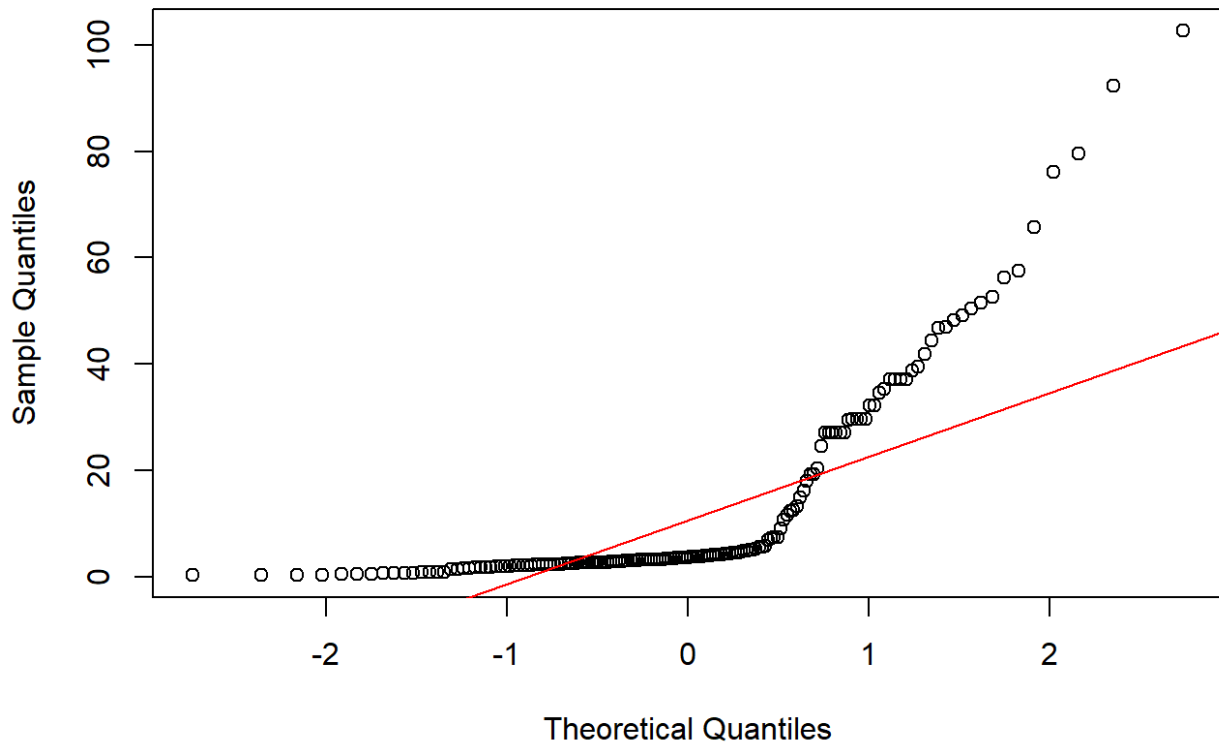
```
hist(db_cc$il33, breaks=30)
```

Histogram of db_cc\$il33



```
qqnorm(db_cc$il33); qqline(db_cc$il33, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$il33)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.2982  2.5630   3.6767  13.3378  18.6874 102.6384
```

```
min(db_cc$il33, na.rm=TRUE)
```

```
## [1] 0.2982189
```

```
db$il33_z <- scale(log(db$il33))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$il33_z <- scale(log(db_cc$il33))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

il33Model_hetero <- glmmTMB(
  il33_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(il33Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## il33_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion: ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    349.1    407.8   -155.5    311.1     144
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id      (Intercept) 0.2579   0.5079
##   Residual              NA      NA
## Number of obs: 163, groups: id, 77
##
## Conditional model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    7.00323    2.28158   3.069 0.002144 **
## t1             -0.14678    0.20072  -0.731 0.464624
## t2              0.64057    0.38477   1.665 0.095948 .
## GROUP1         1.18915    0.32716   3.635 0.000278 ***
## GROUP2         1.35833    0.29017   4.681 2.85e-06 ***
## GROUP3         1.32385    0.35349   3.745 0.000180 ***
## Age            -0.08139    0.02922  -2.786 0.005343 **
## SexM           -0.24933    0.14176  -1.759 0.078609 .
## CIRS_severity_0 -0.33688    0.15531  -2.169 0.030076 *
## t1:GROUP1      -1.73828    0.31016  -5.604 2.09e-08 ***
## t2:GROUP1      -2.30448    0.42329  -5.444 5.20e-08 ***
## t1:GROUP2      -1.57888    0.27279  -5.788 7.13e-09 ***
## t2:GROUP2      -2.17456    0.37627  -5.779 7.50e-09 ***
## t1:GROUP3      -1.82790    0.32184  -5.679 1.35e-08 ***
## t2:GROUP3      -2.47674    0.39724  -6.235 4.52e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.4595    0.1443  -3.183 0.00146 **
## t1            -0.6453    0.3499  -1.845 0.06511 .
## t2            -0.2070    0.2191  -0.945 0.34481
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```



```

il33Model_homo <- glmmTMB(
  il33_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(il33Model_homo, il33Model_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## il33Model_homo: il33_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severi
ty_0 + , zi=~0, disp=~1
## il33Model_homo:      (1 | id), zi=~0, disp=~t
## il33Model_hetero: il33_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## il33Model_hetero:      (1 | id), zi=~0, disp=~t
##
##          Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Chi
sq)
## il33Model_homo    17 350.64 403.24 -158.32   316.64
## il33Model_hetero  19 349.06 407.84 -155.53   311.06 5.5796      2    0.06
143 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

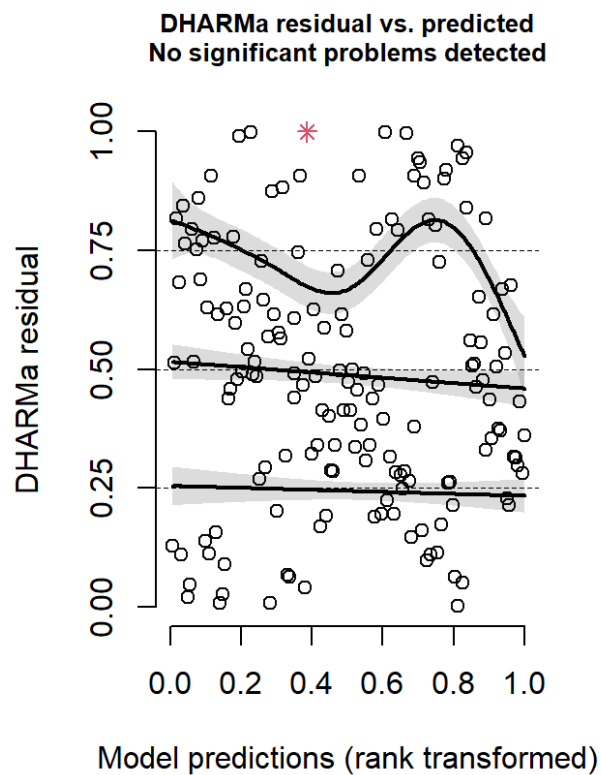
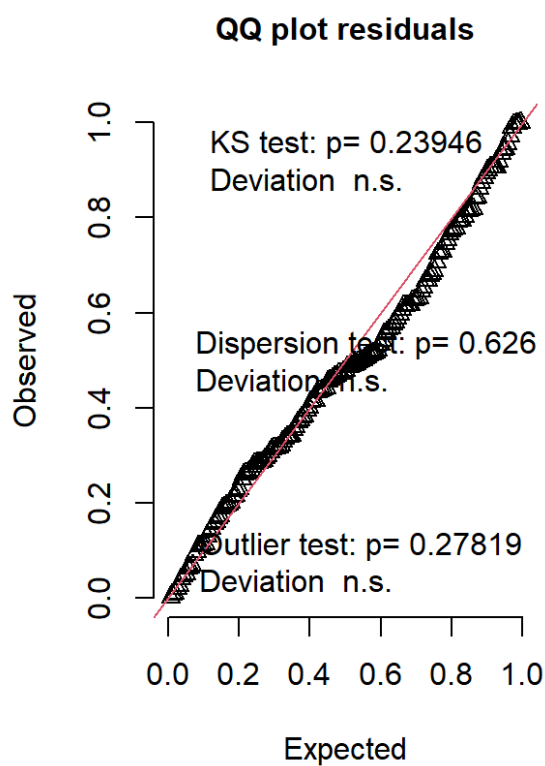
```

```

sim_res <- simulateResiduals(il33Model_hetero, n = 1000)
plot(sim_res)

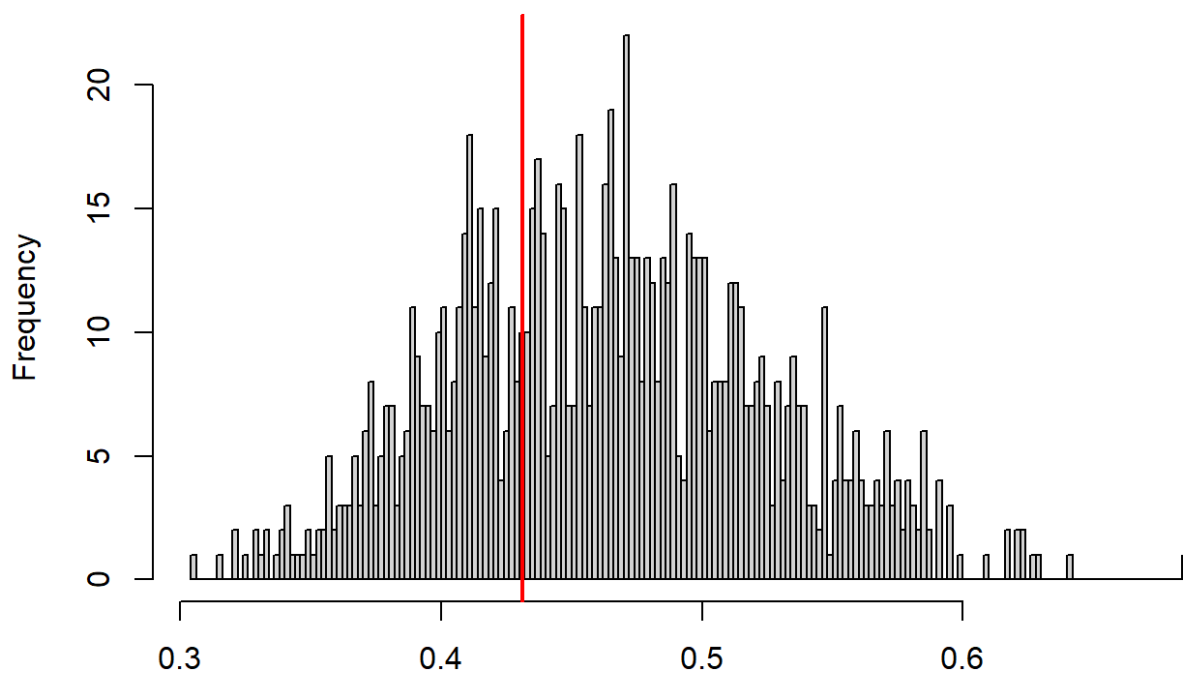
```

DHARMA residual



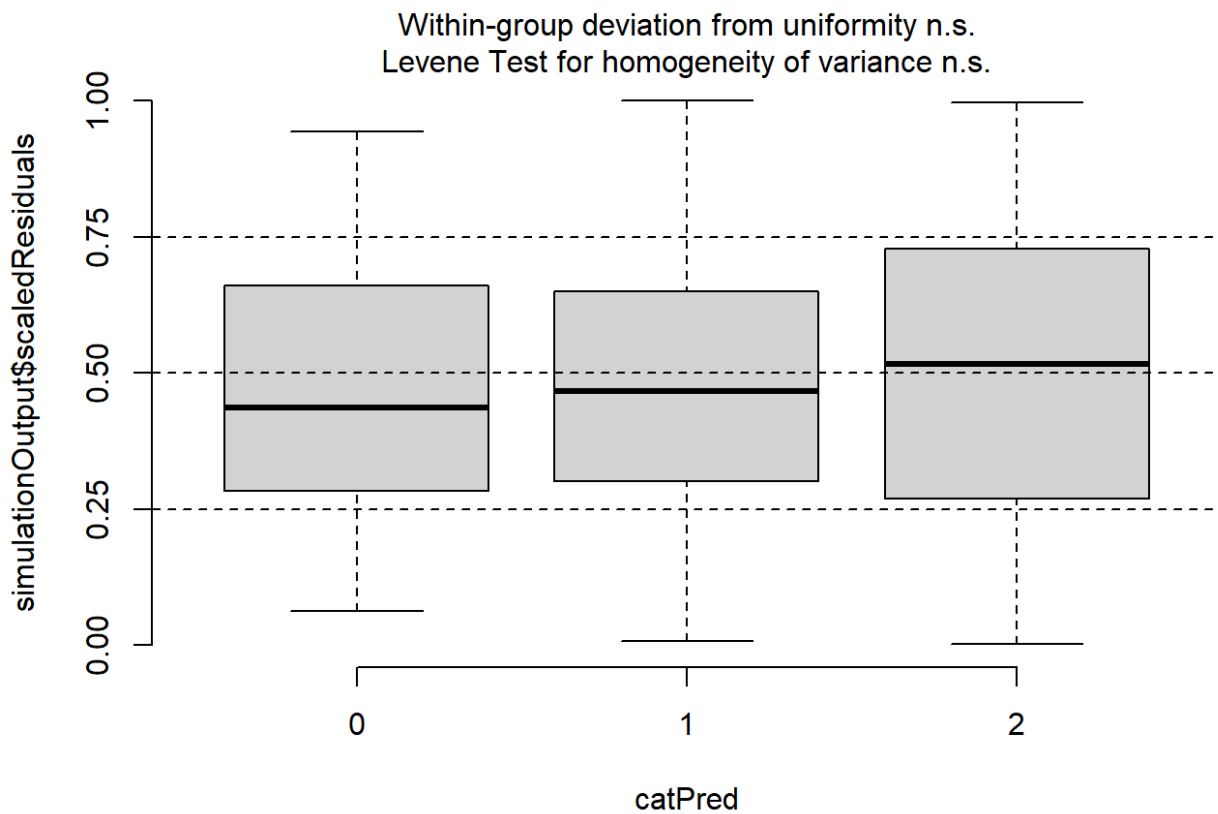
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 0.92917, p-value = 0.626
## alternative hypothesis: two.sided
```

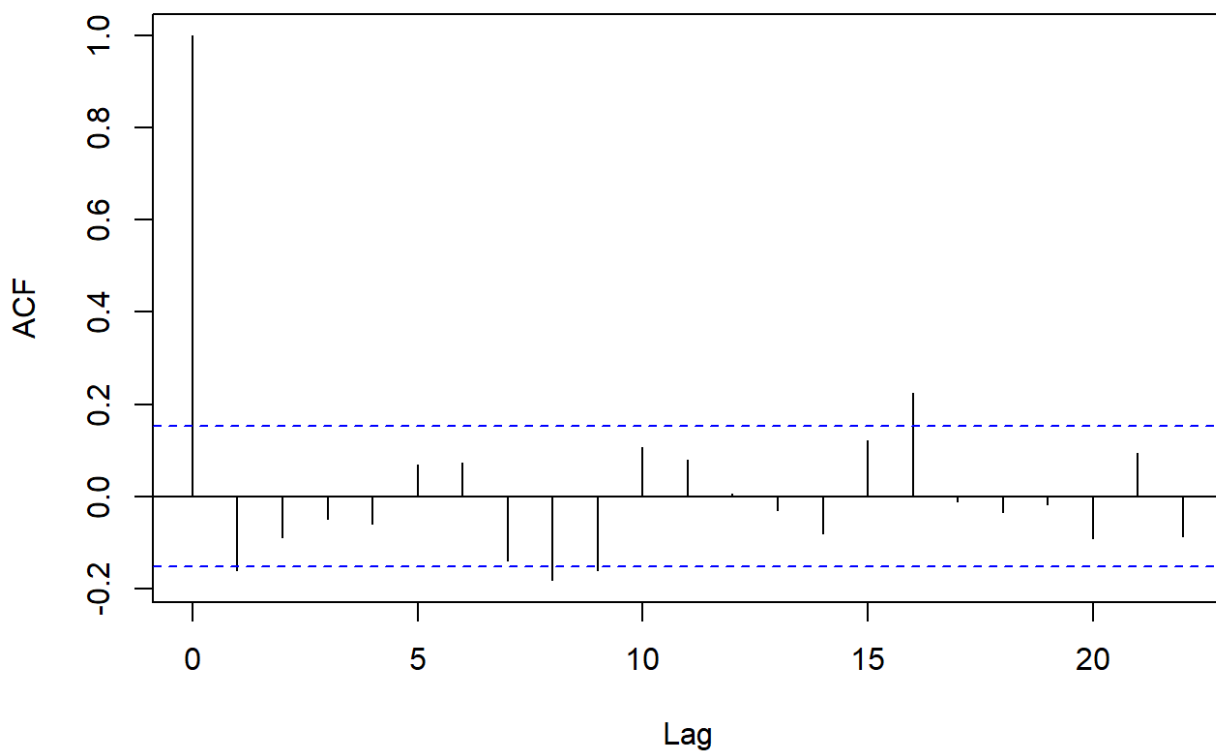
```
testQuantiles(sim_res, db_cc$t)
```



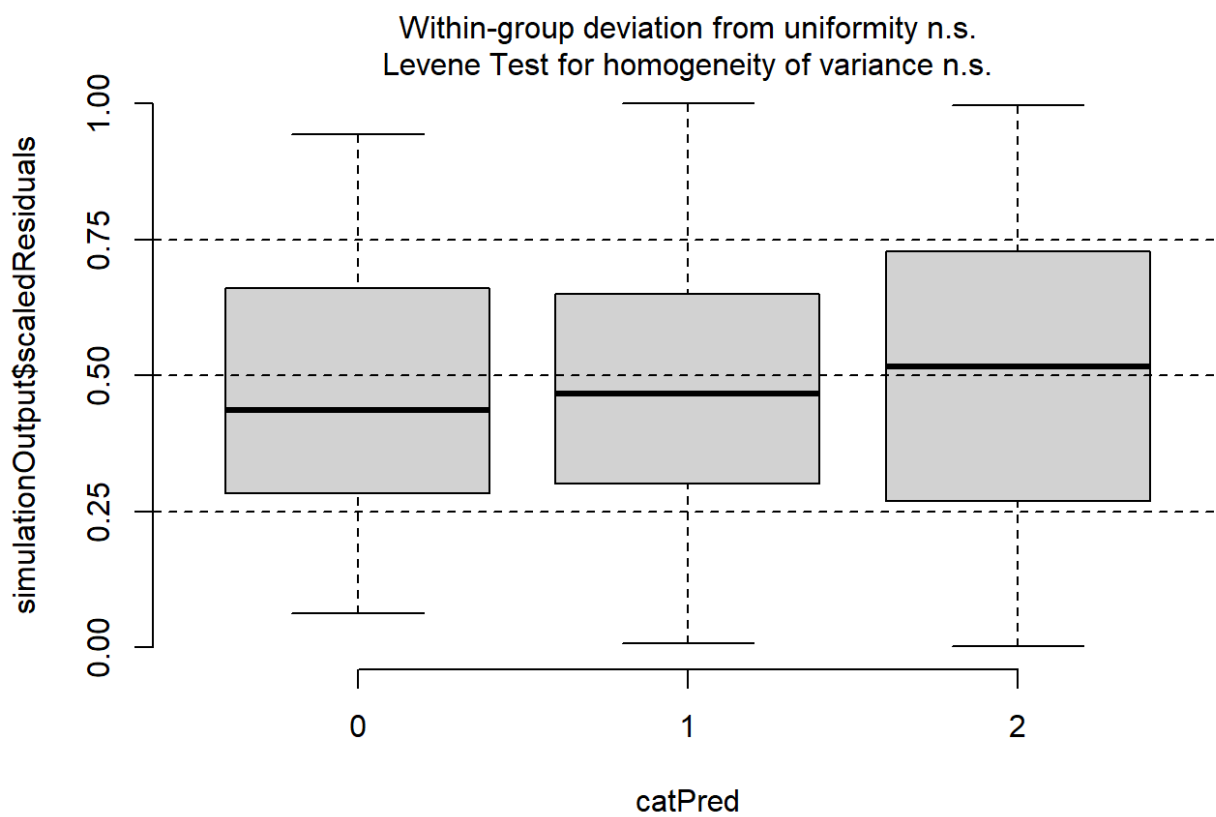
```
## NULL
```

```
acf(residuals(il33Model_hetero))
```

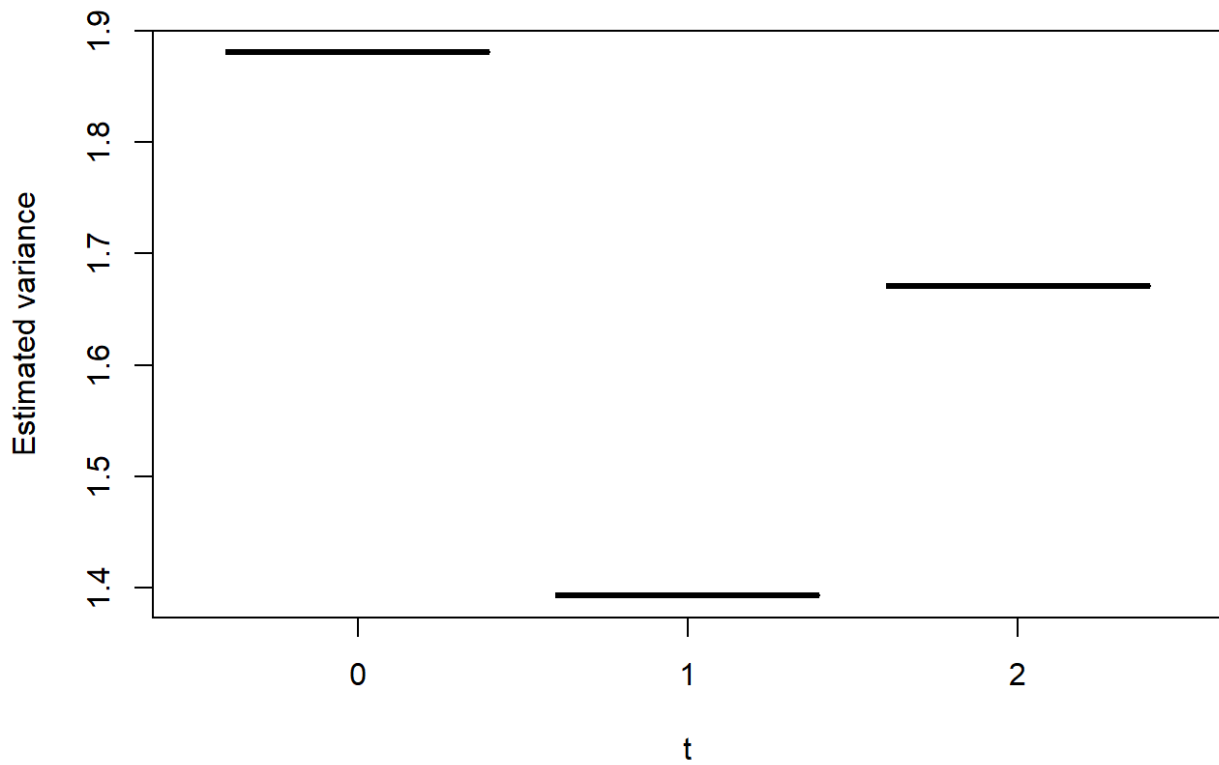
Series residuals(il33Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

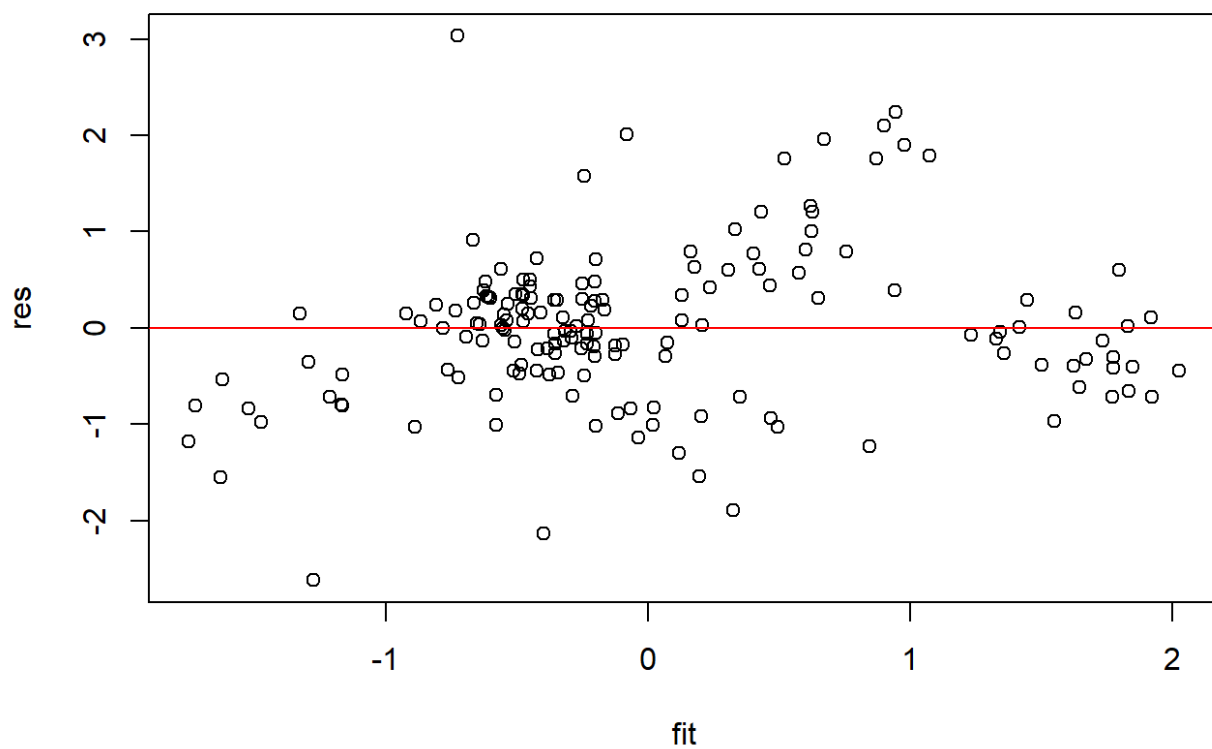


```
pred_disp <- predict(il33Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(il33Model_hetero, type="pearson")
fit <- fitted(il33Model_hetero)

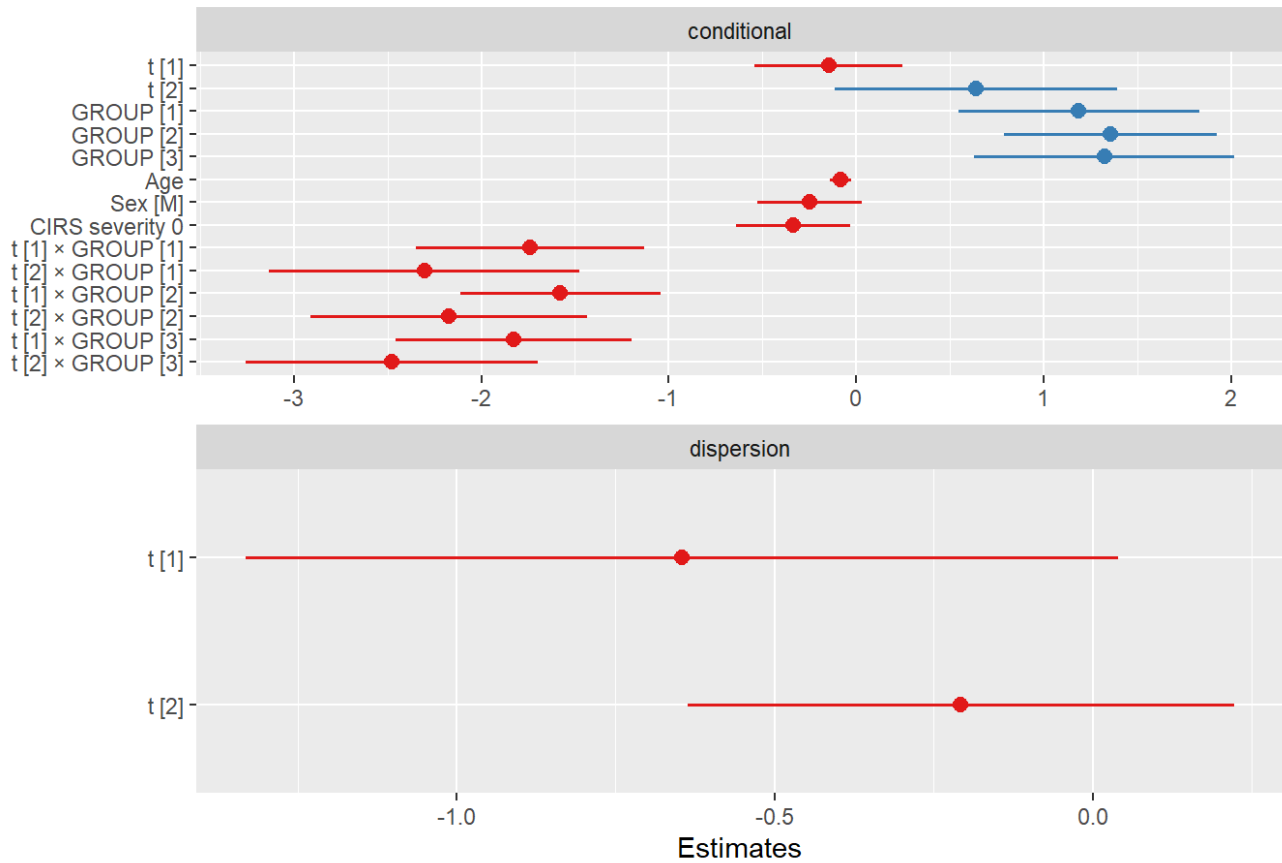
plot(fit, res)
abline(h=0, col="red")
```



12.1 Visualizzazione del modello

```
plot_model(il33Model_hetero)
```

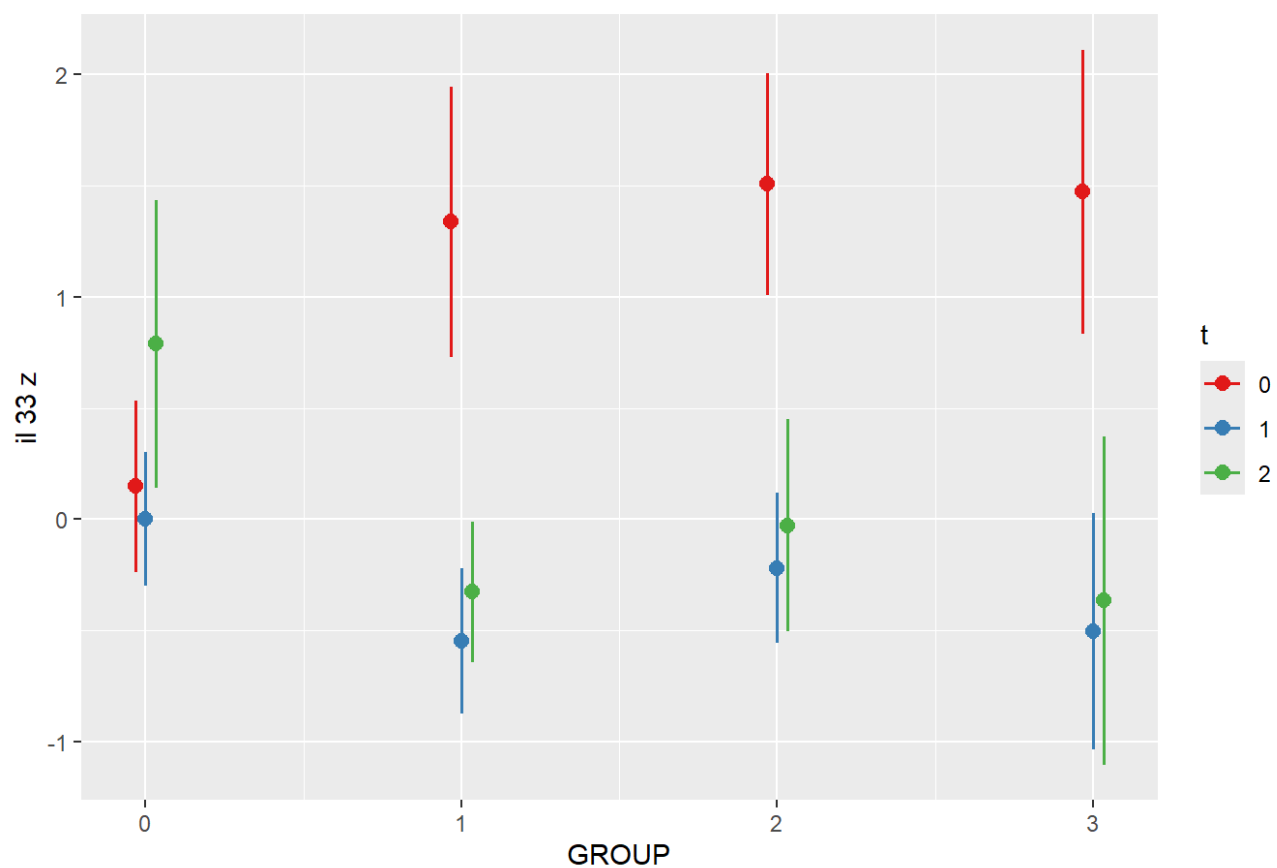
il 33 z



```
plot_model(il33Model_hetero, type = "pred",
           terms = c("GROUP", "t"),
           pred.type = "fe")
```

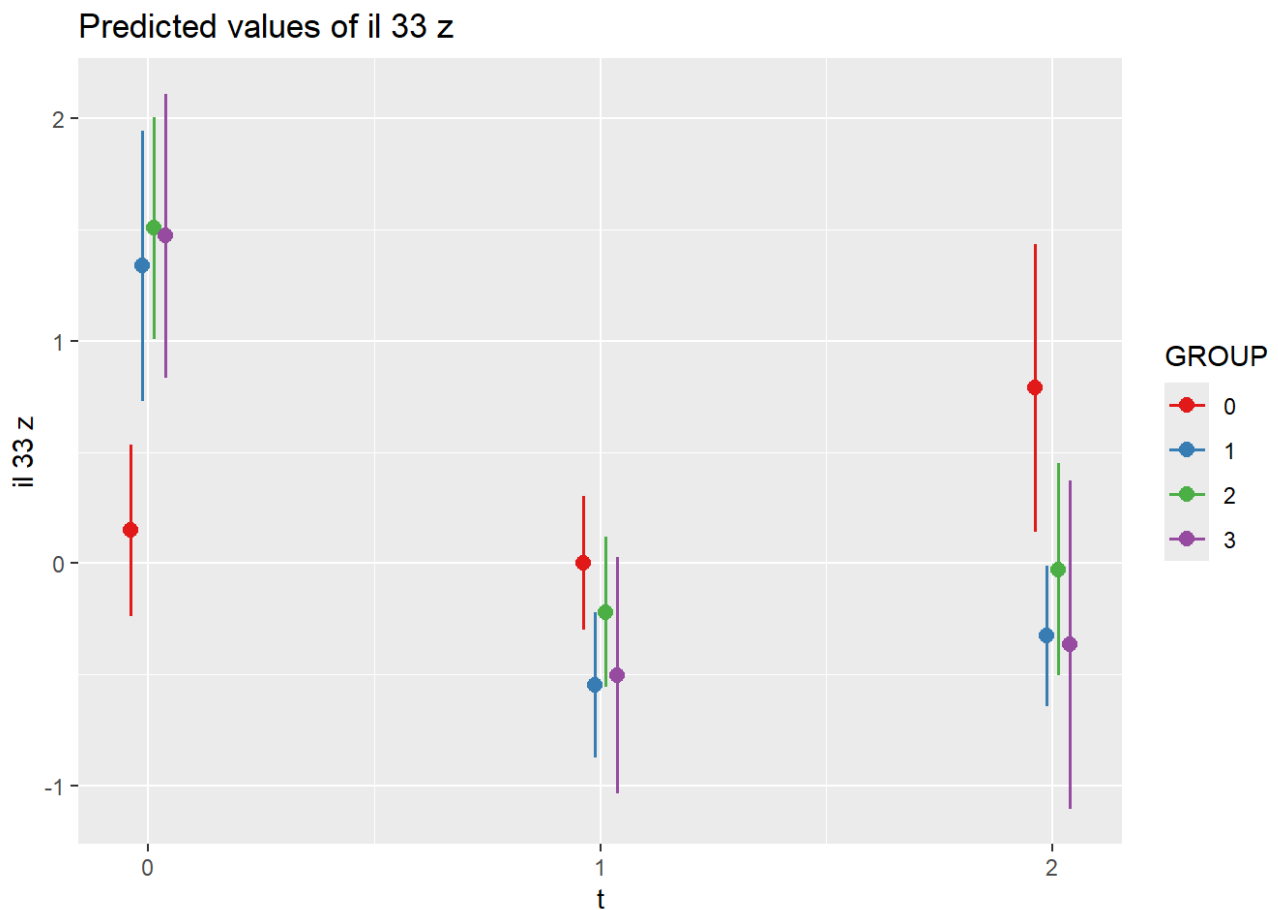
```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

Predicted values of il 33 z



```
plot_model(il33Model_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

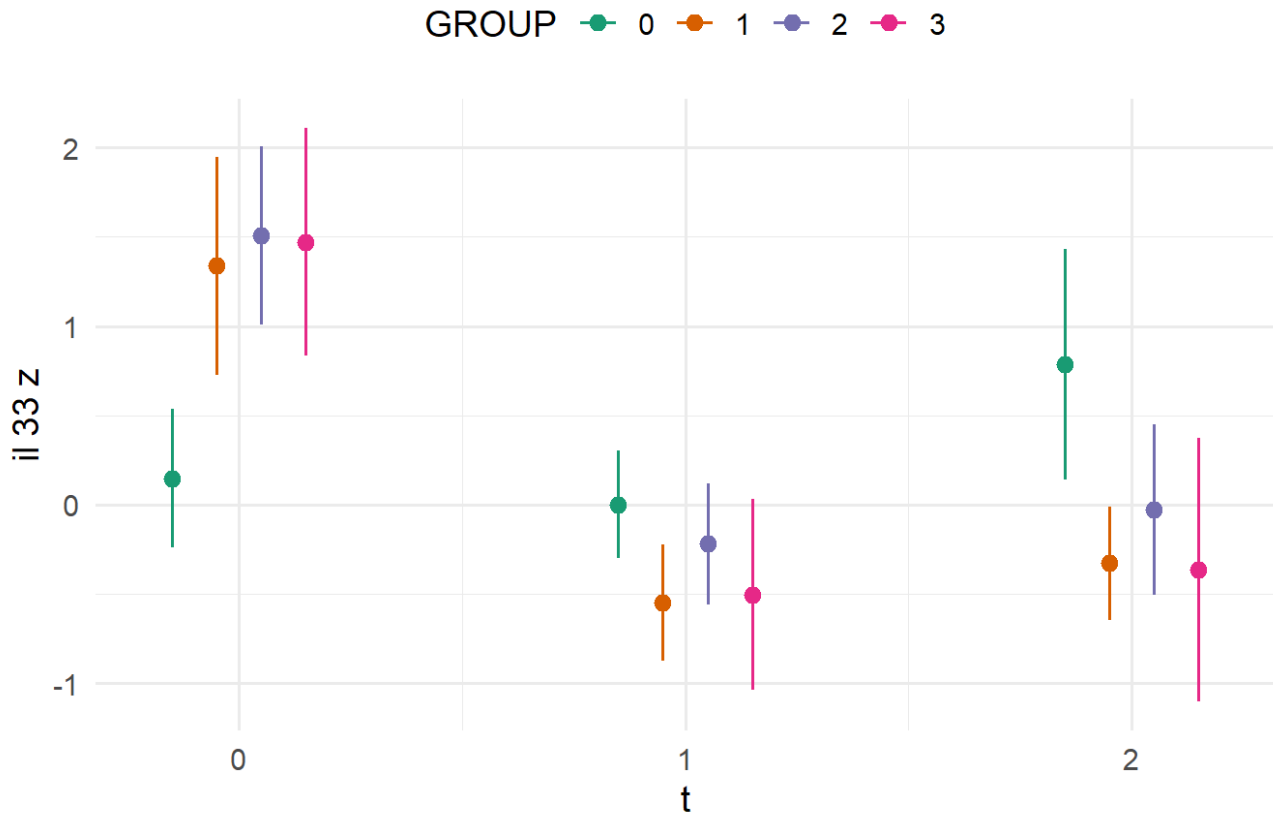



```
plot_model(il33Model_hetero,  
           type = "pred",  
           terms = c("t", "GROUP"),  
           dodge = 0.4) +  
  scale_color_brewer(palette = "Dark2") +  
  theme_minimal(base_size = 14) +  
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg  
e
```

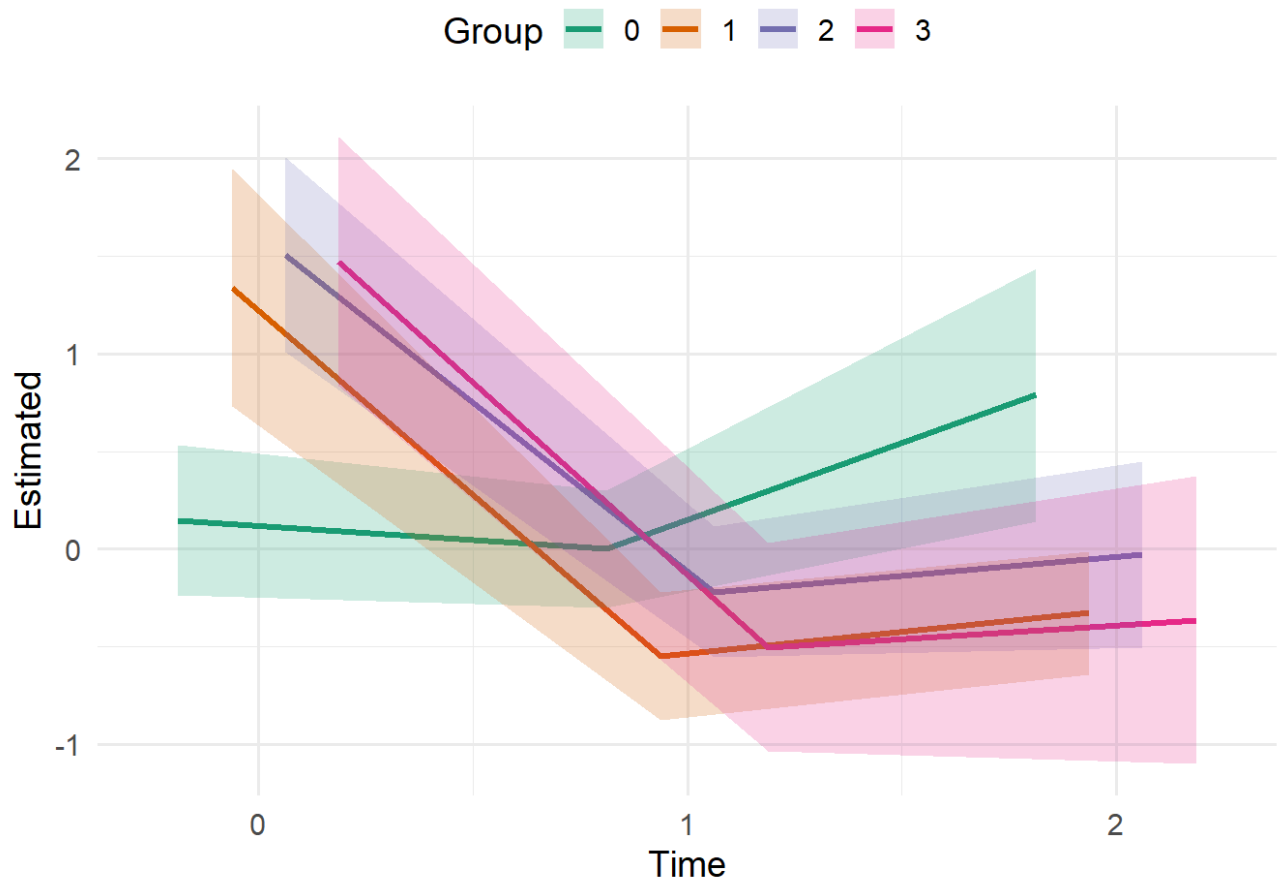
```
## Scale for colour is already present.  
## Adding another scale for colour, which will replace the existing scale.  
## Ignoring unknown labels:  
## • linetype : "GROUP"  
## • shape : "GROUP"
```

Predicted values of il 33 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per g
ruppo)
pred <- ggpredict(il33Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = grou
p)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



12.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(il33Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##   GROUP   emmean    SE  df asymp.LCL asymp.UCL
##   0       0.0231 0.188 Inf   -0.3445    0.391
##   1       1.2123 0.313 Inf    0.5995    1.825
##   2       1.3814 0.253 Inf    0.8854    1.878
##   3       1.3470 0.312 Inf    0.7352    1.959
##
## t = 1:
##   GROUP   emmean    SE  df asymp.LCL asymp.UCL
##   0      -0.1237 0.133 Inf   -0.3849    0.138
##   1      -0.6728 0.164 Inf   -0.9938   -0.352
##   2      -0.3442 0.167 Inf   -0.6715   -0.017
##   3      -0.6277 0.253 Inf   -1.1245   -0.131
##
## t = 2:
##   GROUP   emmean    SE  df asymp.LCL asymp.UCL
##   0       0.6637 0.320 Inf    0.0372    1.290
##   1      -0.4516 0.156 Inf   -0.7567   -0.147
##   2      -0.1525 0.234 Inf   -0.6116    0.307
##   3      -0.4892 0.361 Inf   -1.1966    0.218
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(il33Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##   contrast          estimate    SE  df z.ratio p.value
##   GROUP0 - GROUP1    0.1584 0.230 Inf    0.689  0.9015
##   GROUP0 - GROUP2   -0.1072 0.208 Inf   -0.515  0.9555
##   GROUP0 - GROUP3    0.1110 0.260 Inf    0.426  0.9740
##   GROUP1 - GROUP2   -0.2656 0.226 Inf   -1.176  0.6421
##   GROUP1 - GROUP3   -0.0474 0.309 Inf   -0.153  0.9987
##   GROUP2 - GROUP3    0.2182 0.283 Inf    0.770  0.8680
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(il33Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -1.1892 0.327 Inf  -3.635  0.0016
## GROUP0 - GROUP2 -1.3583 0.290 Inf  -4.681 <0.0001
## GROUP0 - GROUP3 -1.3238 0.353 Inf  -3.745  0.0010
## GROUP1 - GROUP2 -0.1692 0.359 Inf  -0.471  0.9655
## GROUP1 - GROUP3 -0.1347 0.425 Inf  -0.317  0.9890
## GROUP2 - GROUP3  0.0345 0.391 Inf   0.088  0.9998
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.5491 0.228 Inf   2.404  0.0763
## GROUP0 - GROUP2  0.2206 0.201 Inf   1.098  0.6908
## GROUP0 - GROUP3  0.5041 0.271 Inf   1.860  0.2457
## GROUP1 - GROUP2 -0.3286 0.236 Inf  -1.391  0.5051
## GROUP1 - GROUP3 -0.0451 0.325 Inf  -0.139  0.9991
## GROUP2 - GROUP3  0.2835 0.297 Inf   0.954  0.7753
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  1.1153 0.359 Inf   3.110  0.0101
## GROUP0 - GROUP2  0.8162 0.330 Inf   2.476  0.0638
## GROUP0 - GROUP3  1.1529 0.356 Inf   3.236  0.0066
## GROUP1 - GROUP2 -0.2991 0.276 Inf  -1.084  0.6996
## GROUP1 - GROUP3  0.0376 0.396 Inf   0.095  0.9997
## GROUP2 - GROUP3  0.3367 0.366 Inf   0.920  0.7944
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(il33Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      0.147 0.201 Inf   0.731  0.7449
## t0 - t2     -0.641 0.385 Inf  -1.665  0.2188
## t1 - t2     -0.787 0.319 Inf  -2.467  0.0363
##
## GROUP = 1:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      1.885 0.269 Inf   6.998 <0.0001
## t0 - t2      1.664 0.327 Inf   5.081 <0.0001
## t1 - t2     -0.221 0.175 Inf  -1.260  0.4178
##
## GROUP = 2:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      1.726 0.246 Inf   7.007 <0.0001
## t0 - t2      1.534 0.338 Inf   4.534 <0.0001
## t1 - t2     -0.192 0.226 Inf  -0.850  0.6720
##
## GROUP = 3:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      1.975 0.303 Inf   6.510 <0.0001
## t0 - t2      1.836 0.418 Inf   4.394 <0.0001
## t1 - t2     -0.139 0.323 Inf  -0.429  0.9037
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

13 Model il6

```
bptest(lm(il6 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | i
d), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(il6 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
(1 | id), data = db)
## BP = 10.228, df = 14, p-value = 0.7453
```

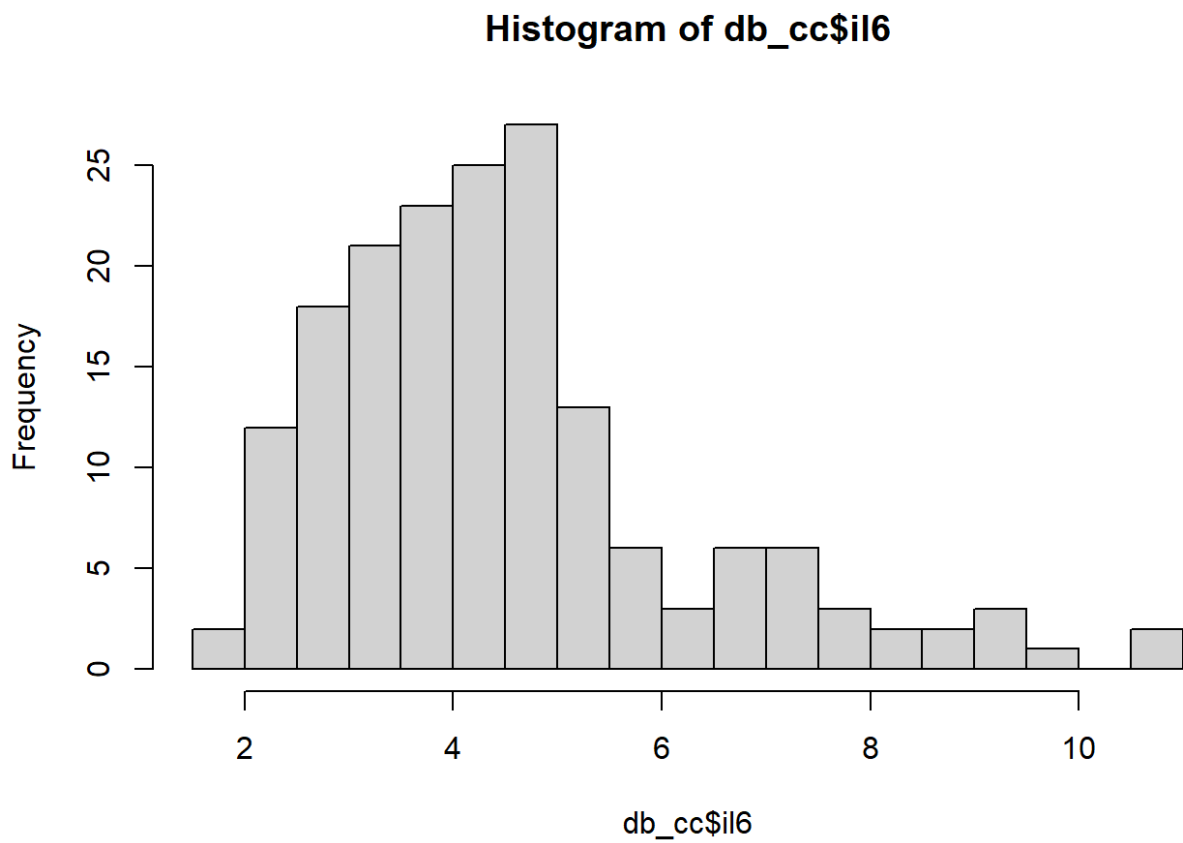
```
vars <- c("il6","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

nrow(db); nrow(db_cc)
```

```
## [1] 243
```

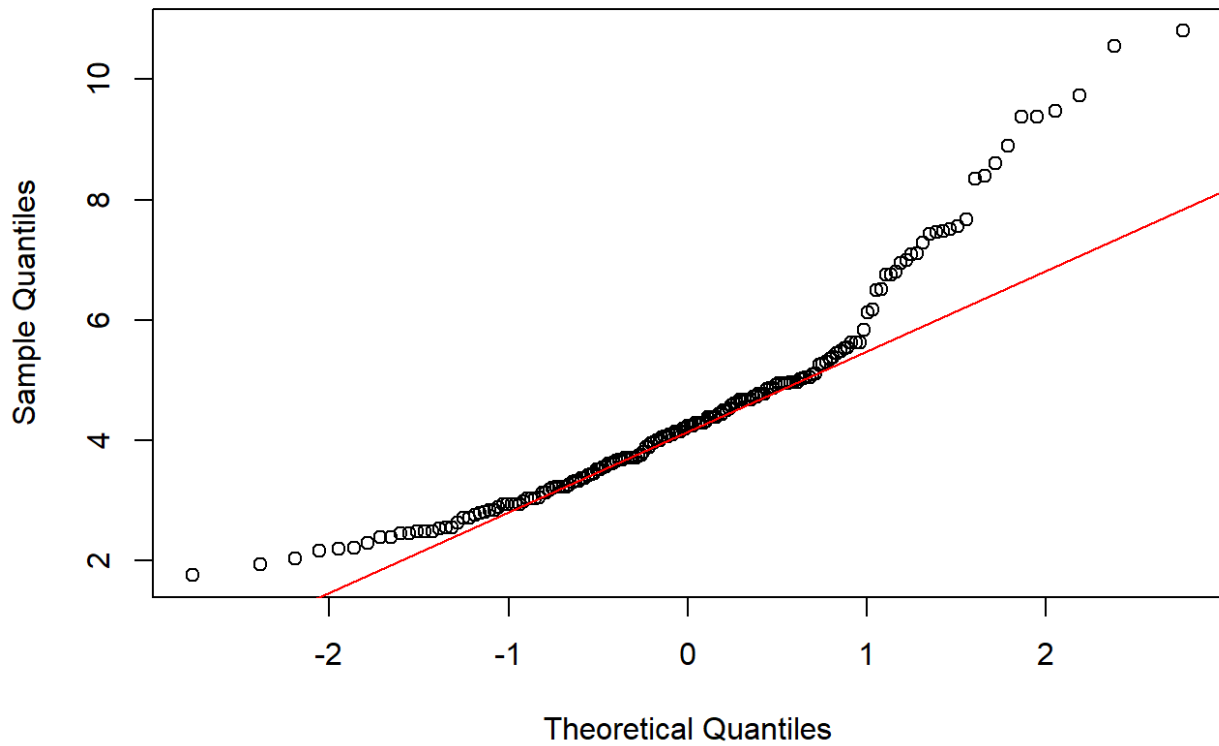
```
## [1] 175
```

```
hist(db_cc$il6, breaks=30)
```



```
qqnorm(db_cc$il6); qqline(db_cc$il6, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$il6)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.760   3.247   4.238   4.521   5.053  10.803
```

```
min(db_cc$il6, na.rm=TRUE)
```

```
## [1] 1.759832
```



```
db$il6_z <- scale(log(db$il6))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$il6_z <- scale(log(db_cc$il6))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

il6Model_hetero <- glmmTMB(
  il6_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(il6Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## il6_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion:                ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    450.2    510.4    -206.1    412.2      156
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id       (Intercept) 0.3219   0.5673
##   Residual                NA      NA
## Number of obs: 175, groups: id, 79
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    3.927253   2.546014   1.543 0.122950
## t1              0.799412   0.227131   3.520 0.000432 ***
## t2              1.106766   0.431371   2.566 0.010297 *
## GROUP1         -0.002110   0.360047  -0.006 0.995323
## GROUP2          0.487573   0.310729   1.569 0.116619
## GROUP3         -0.174663   0.402137  -0.434 0.664043
## Age            -0.055573   0.032889  -1.690 0.091085 .
## SexM           -0.158834   0.166431  -0.954 0.339905
## CIRS_severity_0 -0.148202   0.174549  -0.849 0.395851
## t1:GROUP1       0.005085   0.356347   0.014 0.988614
## t2:GROUP1      -0.813264   0.486025  -1.673 0.094269 .
## t1:GROUP2       0.048266   0.295843   0.163 0.870404
## t2:GROUP2      -0.082686   0.425170  -0.194 0.845801
## t1:GROUP3       0.786612   0.380811   2.066 0.038864 *
## t2:GROUP3       0.553839   0.488173   1.135 0.256579
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.30646   0.12411  -2.469   0.0135 *
## t1            -0.54222   0.25146  -2.156   0.0311 *
## t2             0.04764   0.16432   0.290   0.7719
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

il6Model_homo <- glmmTMB(
  il6_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(il6Model_homo, il6Model_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## il6Model_homo: il6_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## il6Model_homo:      (1 | id), zi=~0, disp=~t
## il6Model_hetero: il6_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## il6Model_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## il6Model_homo   17 453.72 507.52 -209.86   419.72
## il6Model_hetero  19 450.22 510.35 -206.11   412.22 7.5002      2    0.02352 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

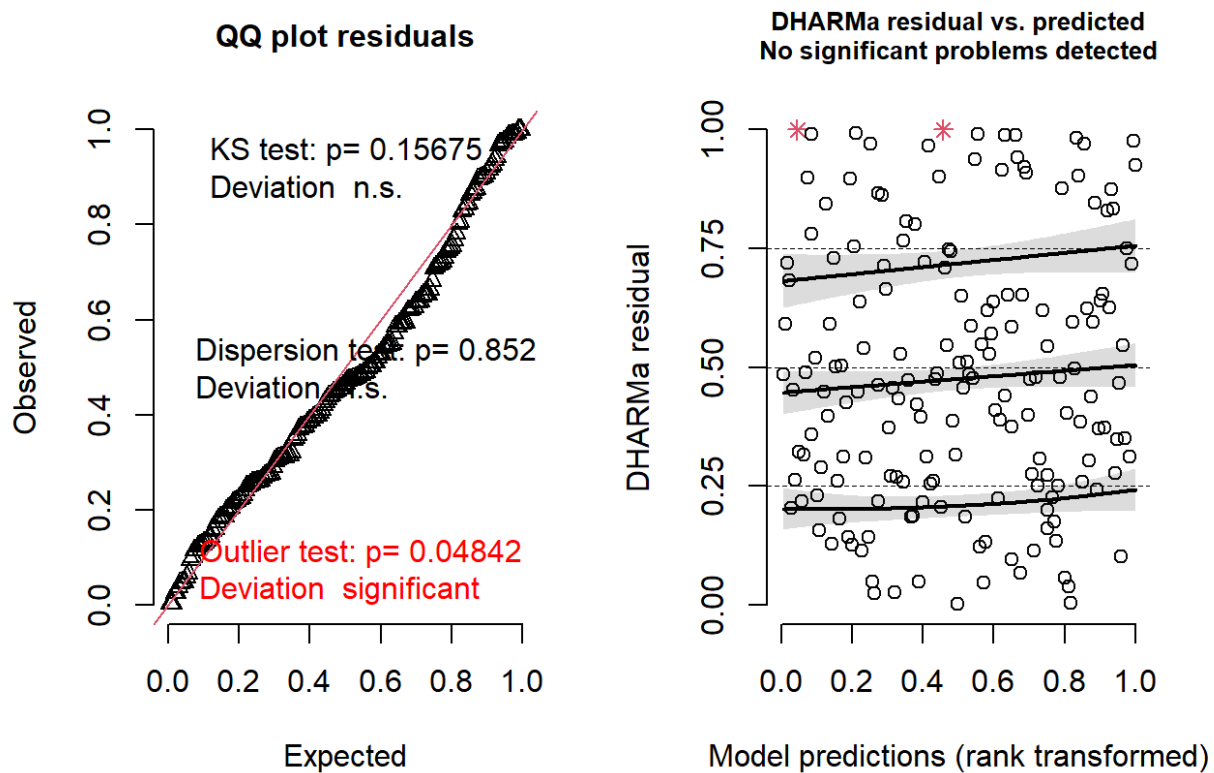
```

```

sim_res <- simulateResiduals(il6Model_hetero, n = 1000)
plot(sim_res)

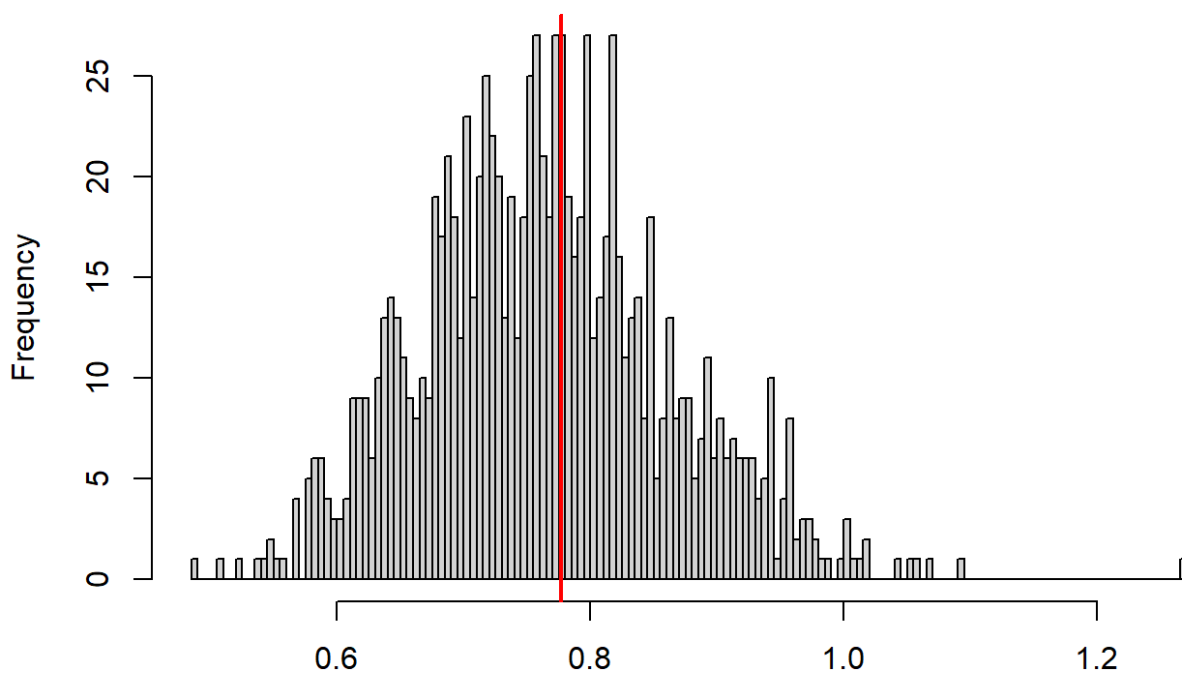
```

DHARMA residual



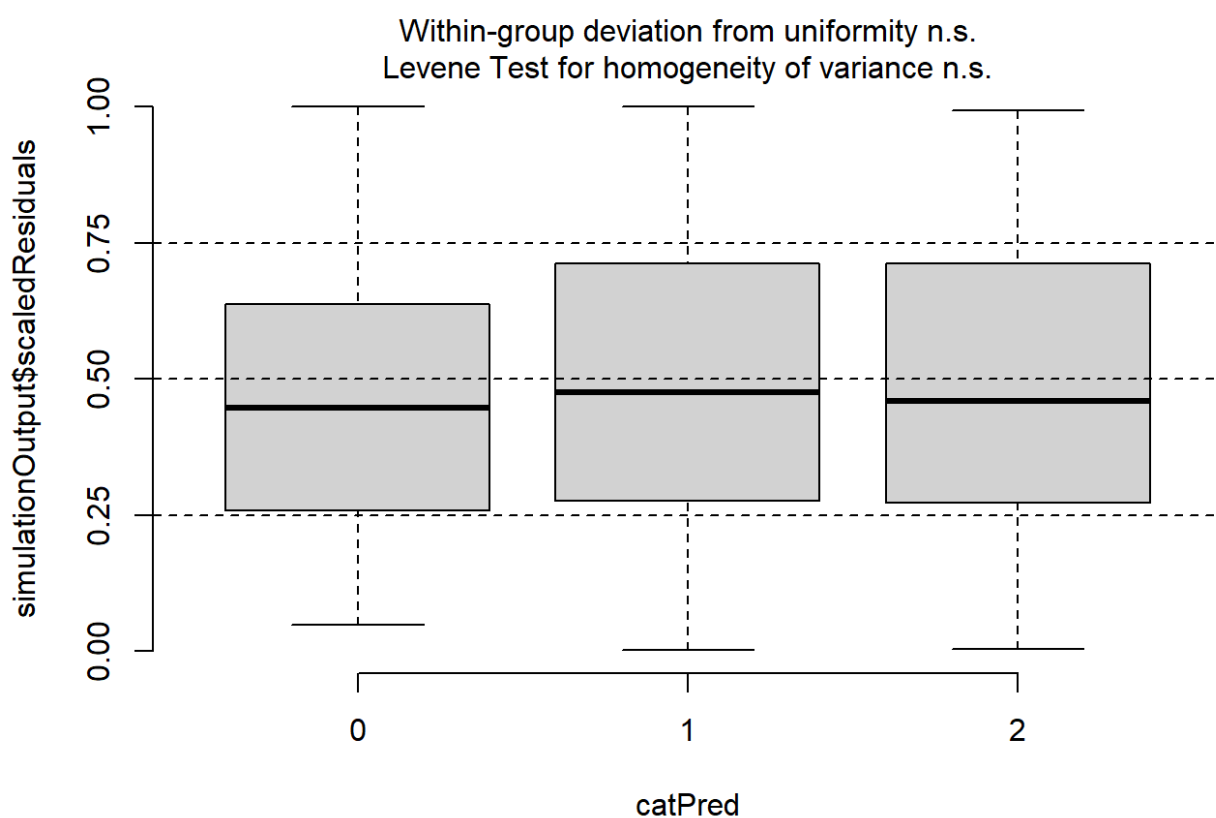
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 1.0164, p-value = 0.852
## alternative hypothesis: two.sided
```

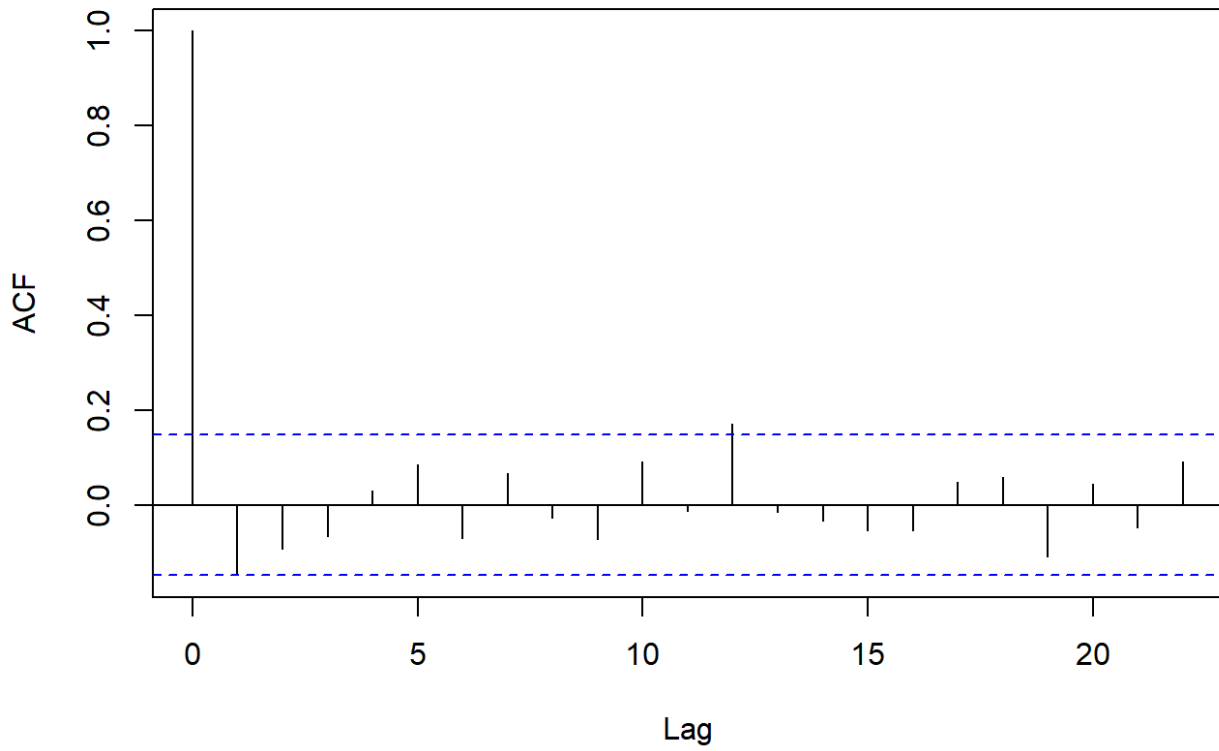
```
testQuantiles(sim_res, db_cc$t)
```



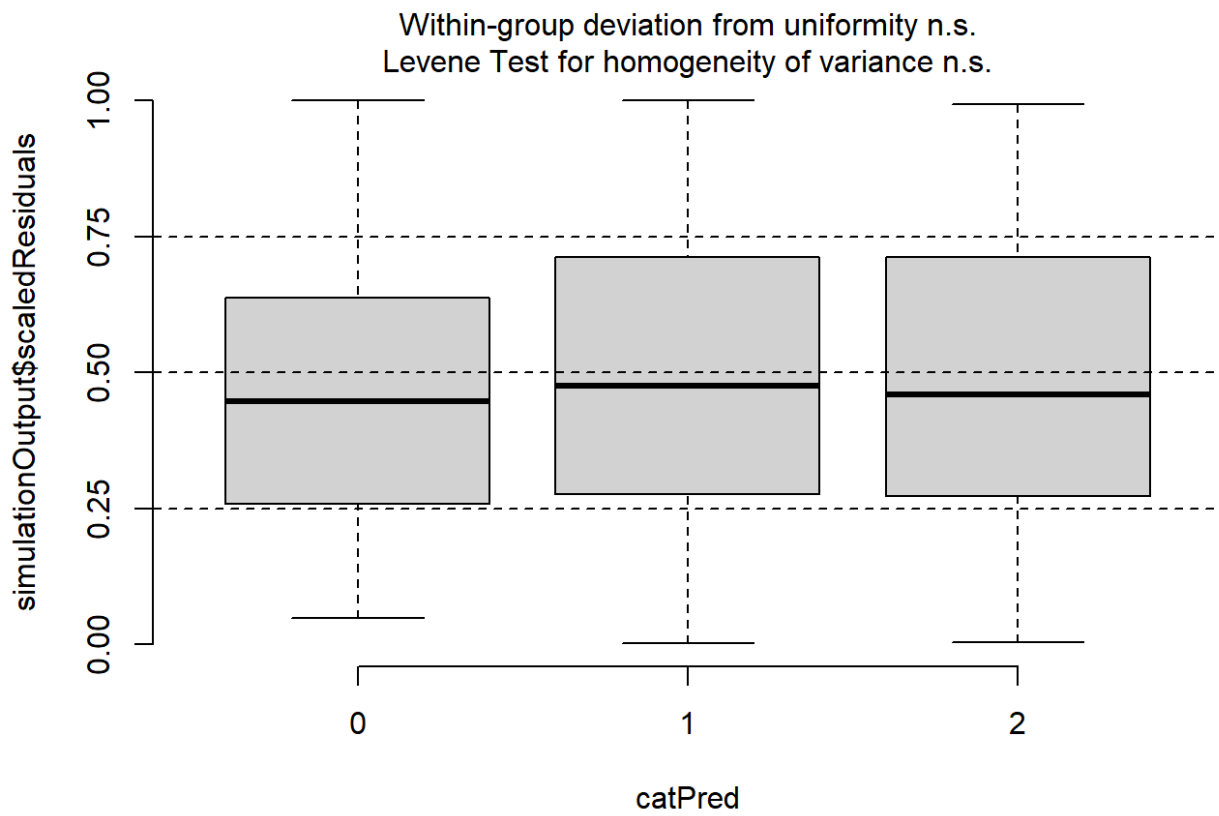
```
## NULL
```

```
acf(residuals(il6Model_hetero))
```

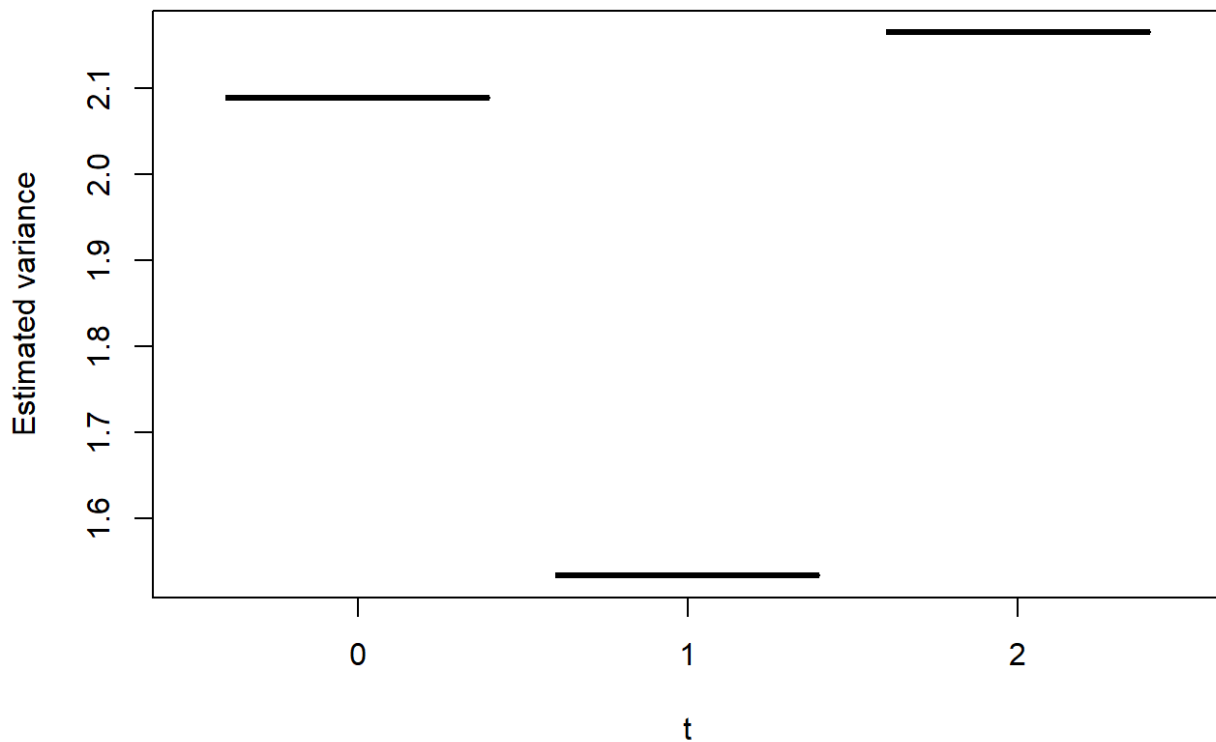
Series residuals(il6Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

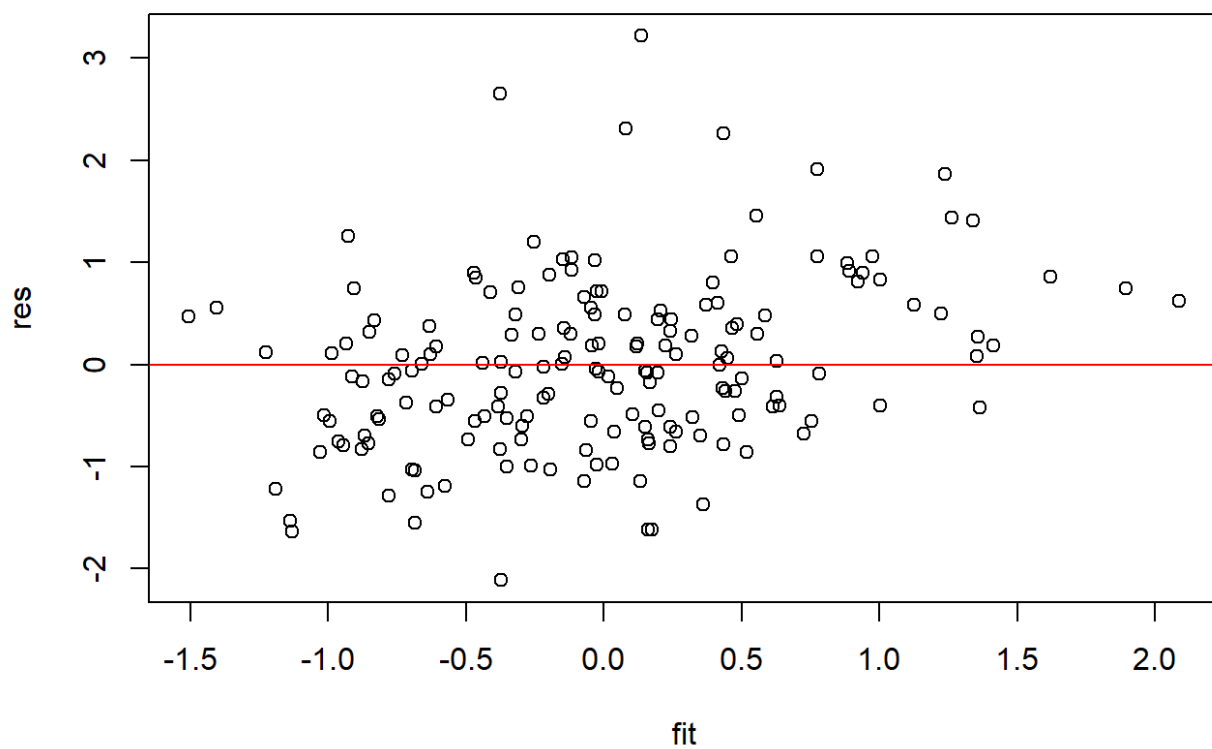


```
pred_disp <- predict(il6Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(il6Model_hetero, type="pearson")
fit <- fitted(il6Model_hetero)

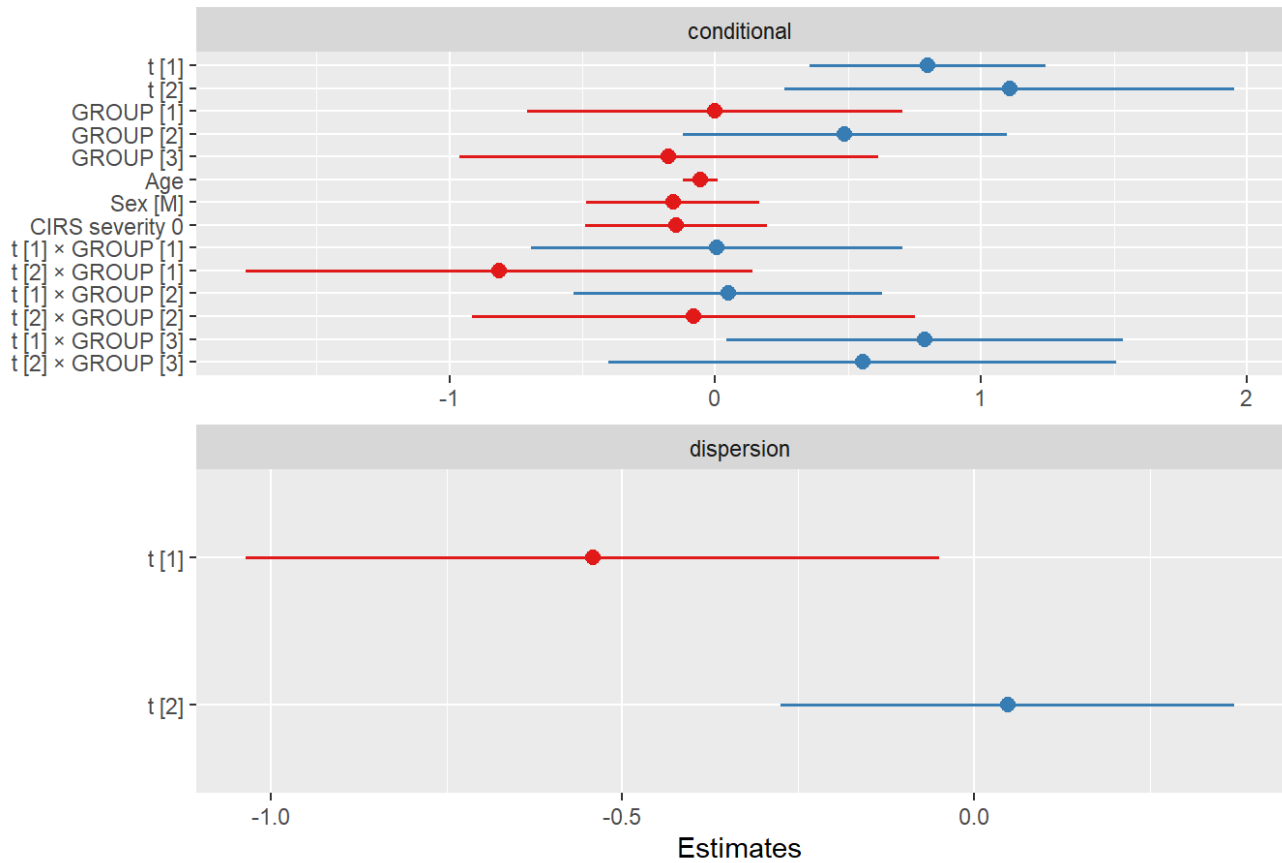
plot(fit, res)
abline(h=0, col="red")
```



13.1 Visualizzazione del modello

```
plot_model(il6Model_hetero)
```

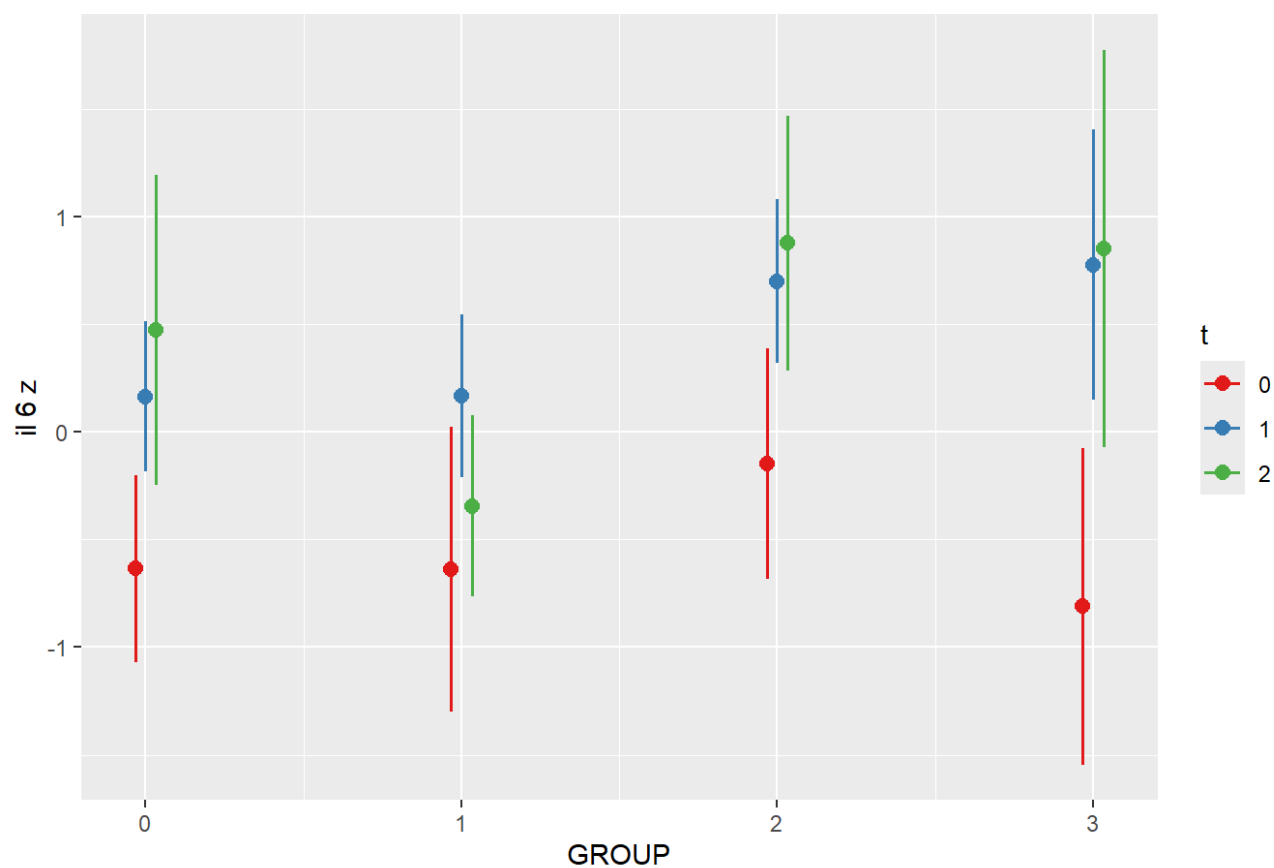

il 6 z



```
plot_model(il6Model_hetero, type = "pred",
           terms = c("GROUP", "t"),
           pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

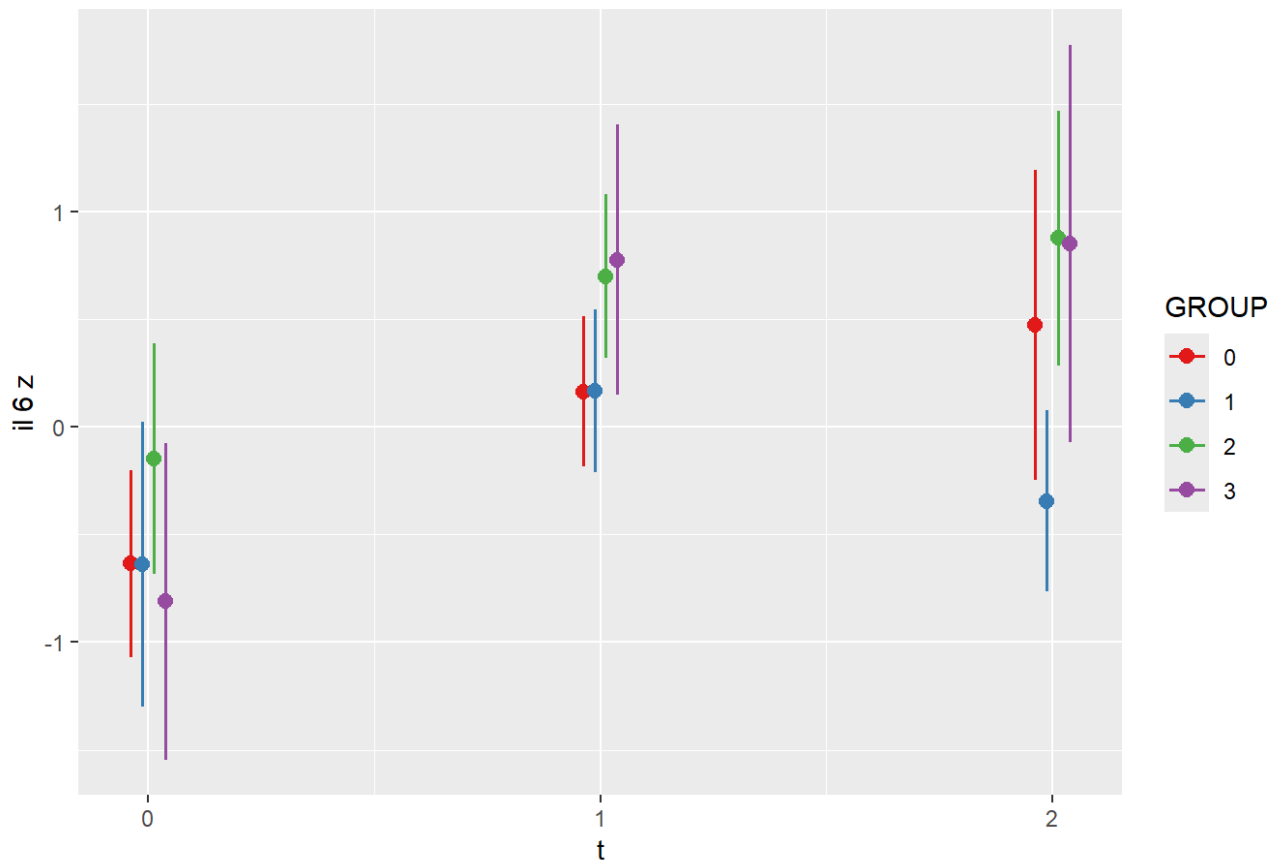
Predicted values of il 6 z



```
plot_model(il6Model_hetero, type = "pred",
           terms = c("t", "GROUP"),
           pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of il 6 z

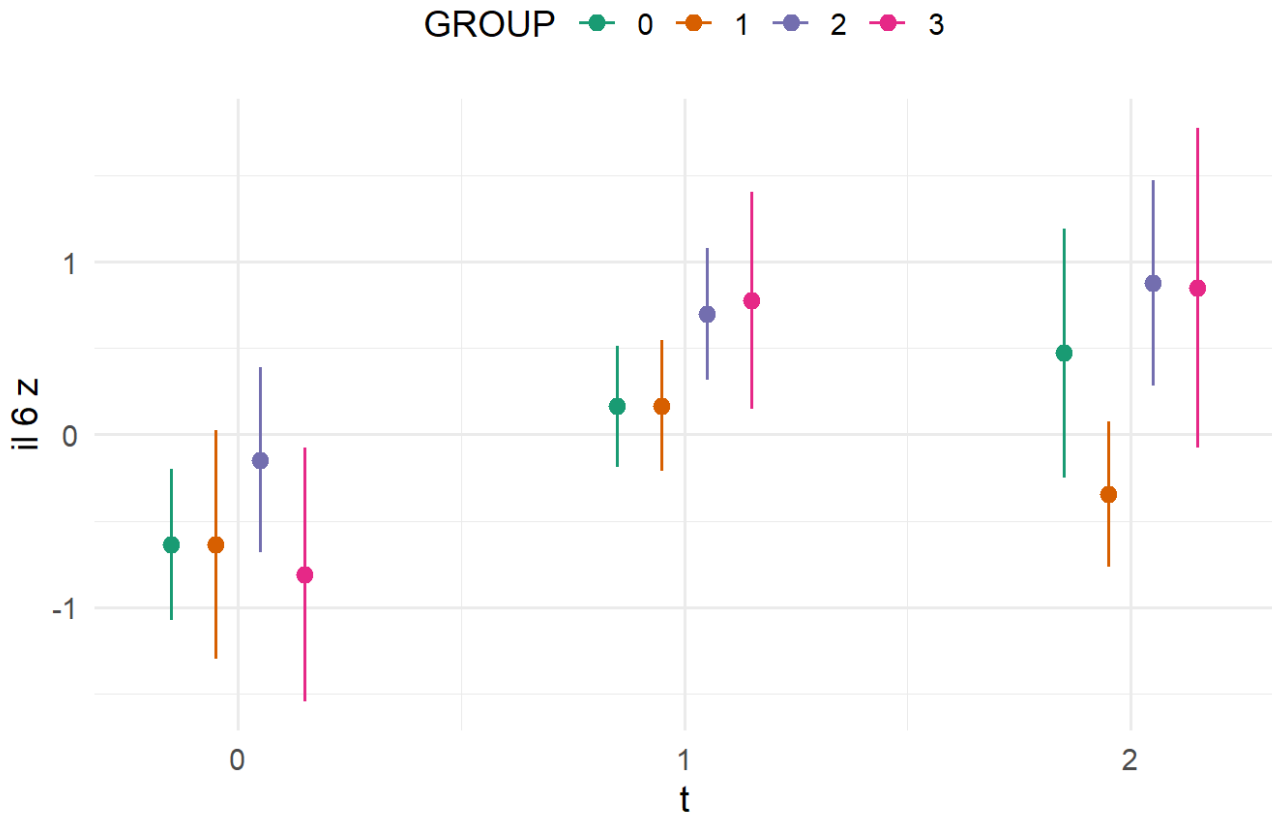


```
plot_model(il6Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

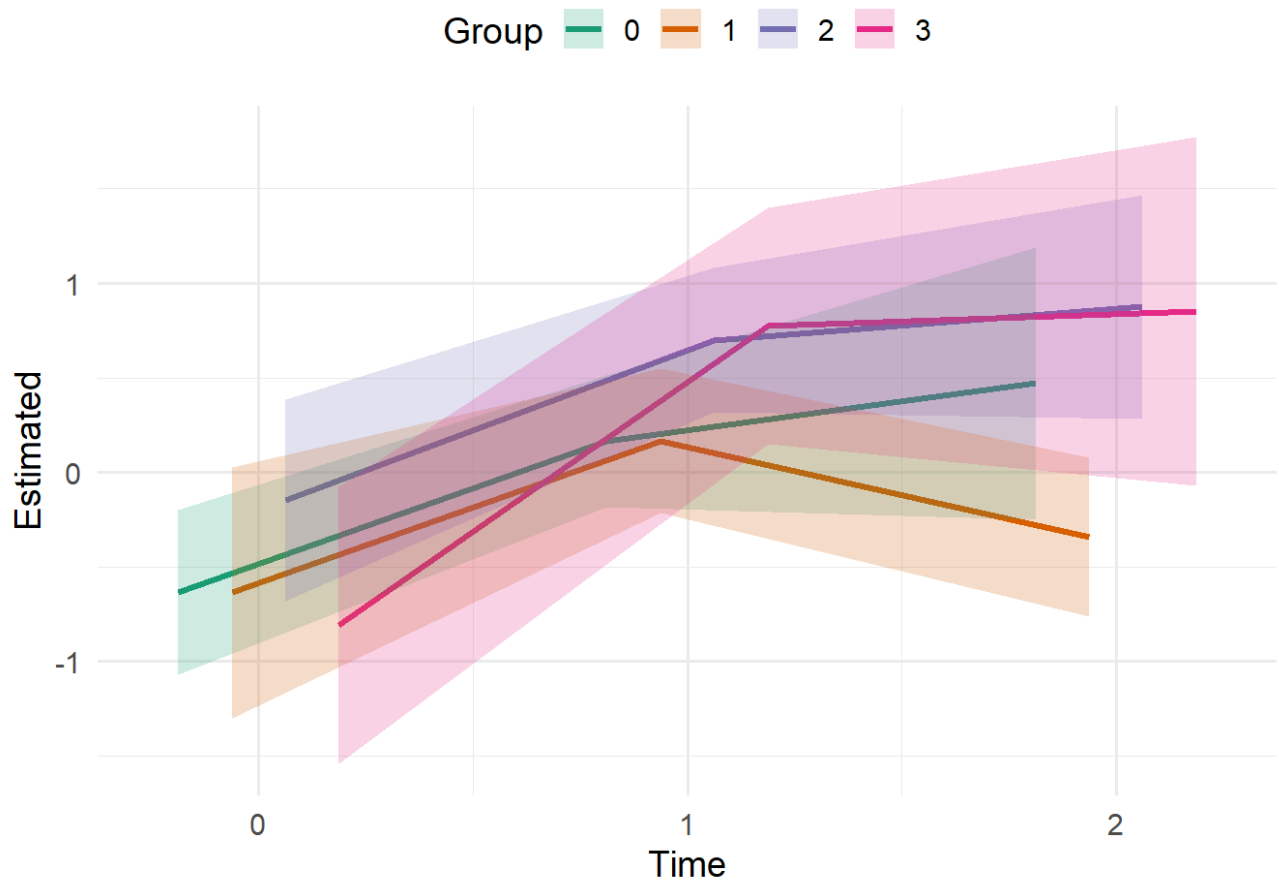
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of il 6 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(il6Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



13.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(il6Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.7148 0.208 Inf   -1.123   -0.3064
##  1      -0.7169 0.340 Inf   -1.384   -0.0497
##  2      -0.2273 0.272 Inf   -0.760    0.3055
##  3      -0.8895 0.358 Inf   -1.591   -0.1876
##
## t = 1:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.0846 0.156 Inf   -0.222    0.3911
##  1       0.0876 0.189 Inf   -0.283    0.4580
##  2       0.6204 0.192 Inf    0.245    0.9959
##  3       0.6965 0.299 Inf    0.111    1.2821
##
## t = 2:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.3919 0.357 Inf   -0.307    1.0912
##  1      -0.4234 0.210 Inf   -0.835   -0.0118
##  2       0.7968 0.295 Inf    0.219    1.3742
##  3       0.7711 0.455 Inf   -0.121    1.6631
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(il6Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  contrast          estimate    SE  df z.ratio p.value
##  GROUP0 - GROUP1      0.272 0.260 Inf   1.044  0.7237
##  GROUP0 - GROUP2     -0.476 0.234 Inf  -2.035  0.1753
##  GROUP0 - GROUP3     -0.272 0.301 Inf  -0.903  0.8033
##  GROUP1 - GROUP2     -0.748 0.251 Inf  -2.982  0.0152
##  GROUP1 - GROUP3     -0.544 0.357 Inf  -1.523  0.4238
##  GROUP2 - GROUP3      0.204 0.329 Inf   0.619  0.9260
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(il6Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.00211 0.360 Inf   0.006  1.0000
## GROUP0 - GROUP2 -0.48757 0.311 Inf  -1.569  0.3964
## GROUP0 - GROUP3  0.17466 0.402 Inf   0.434  0.9726
## GROUP1 - GROUP2 -0.48968 0.383 Inf  -1.279  0.5764
## GROUP1 - GROUP3  0.17255 0.478 Inf   0.361  0.9839
## GROUP2 - GROUP3  0.66224 0.436 Inf   1.518  0.4266
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.00298 0.263 Inf  -0.011  1.0000
## GROUP0 - GROUP2 -0.53584 0.234 Inf  -2.292  0.0998
## GROUP0 - GROUP3 -0.61195 0.319 Inf  -1.917  0.2209
## GROUP1 - GROUP2 -0.53286 0.266 Inf  -2.001  0.1874
## GROUP1 - GROUP3 -0.60897 0.379 Inf  -1.609  0.3736
## GROUP2 - GROUP3 -0.07611 0.349 Inf  -0.218  0.9963
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.81537 0.419 Inf   1.947  0.2084
## GROUP0 - GROUP2 -0.40489 0.384 Inf  -1.055  0.7170
## GROUP0 - GROUP3 -0.37918 0.442 Inf  -0.857  0.8268
## GROUP1 - GROUP2 -1.22026 0.359 Inf  -3.402  0.0037
## GROUP1 - GROUP3 -1.19455 0.507 Inf  -2.356  0.0857
## GROUP2 - GROUP3  0.02571 0.472 Inf   0.054  0.9999
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(il6Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.7994 0.227 Inf  -3.520  0.0013
## t0 - t2      -1.1068 0.431 Inf  -2.566  0.0278
## t1 - t2      -0.3074 0.356 Inf   -0.862  0.6641
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.8045 0.304 Inf  -2.646  0.0222
## t0 - t2      -0.2935 0.378 Inf  -0.777  0.7169
## t1 - t2       0.5110 0.231 Inf   2.210  0.0695
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.8477 0.266 Inf  -3.182  0.0042
## t0 - t2      -1.0241 0.399 Inf  -2.568  0.0276
## t1 - t2      -0.1764 0.298 Inf  -0.592  0.8243
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -1.5860 0.361 Inf  -4.396 <0.0001
## t0 - t2      -1.6606 0.522 Inf  -3.181  0.0042
## t1 - t2      -0.0746 0.427 Inf  -0.175  0.9833
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

14 Model il10

```
bptest(lm(il10 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(il10 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 9.3544, df = 14, p-value = 0.8077
```

```
vars <- c("il10","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

nrow(db); nrow(db_cc)
```

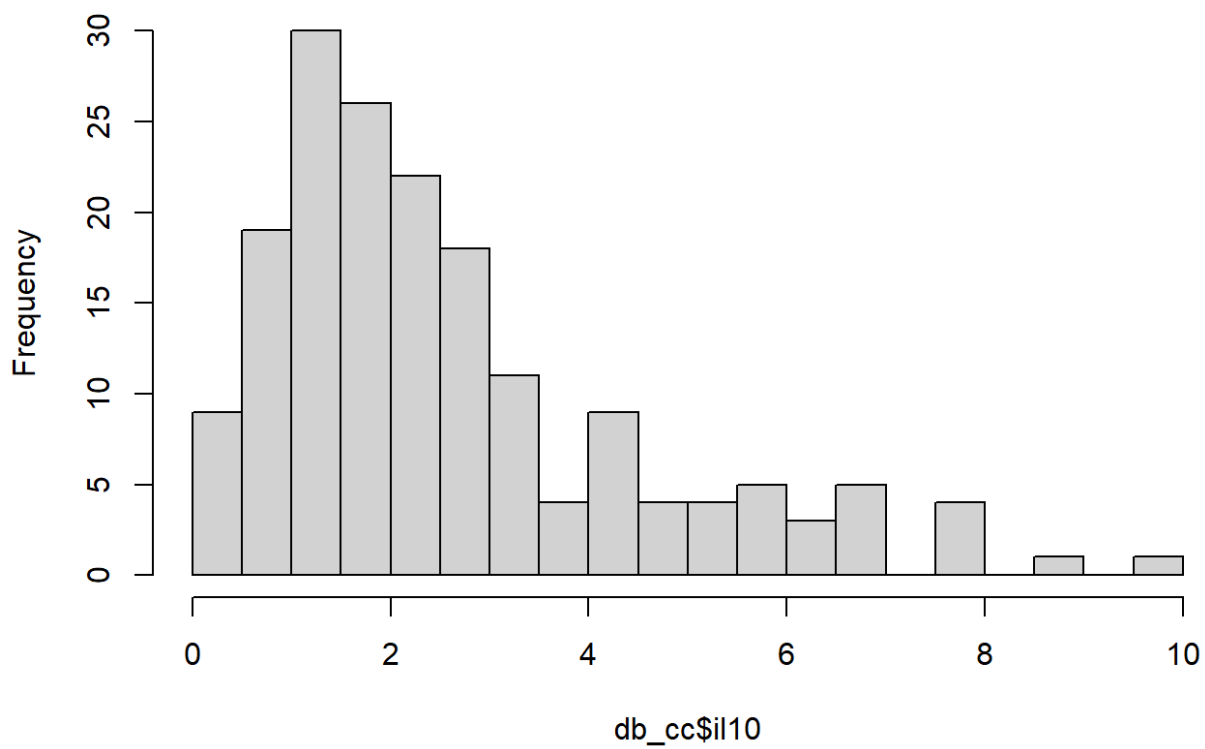


```
## [1] 243
```

```
## [1] 175
```

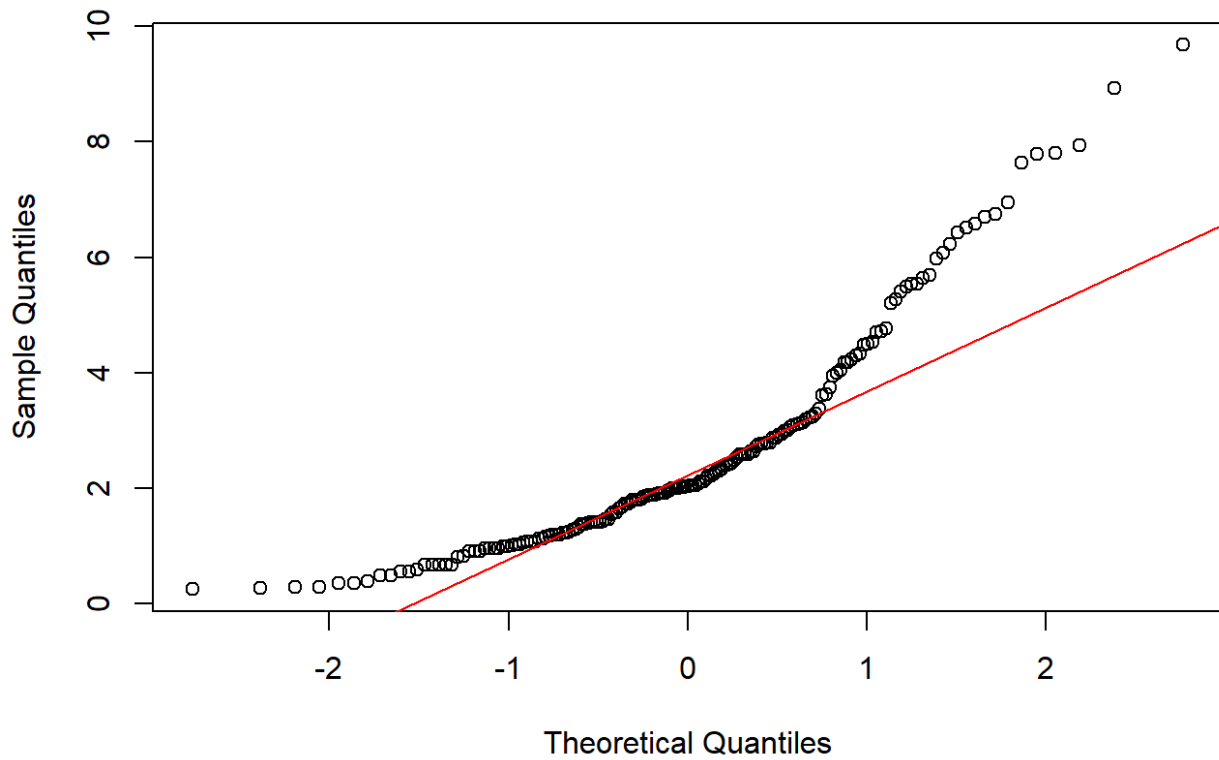
```
hist(db_cc$il10, breaks=30)
```

Histogram of db_cc\$il10



```
qqnorm(db_cc$il10); qqline(db_cc$il10, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$il10)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##  0.2515  1.2421  2.0417  2.6253  3.2041  9.6782
```

```
min(db_cc$il10, na.rm=TRUE)
```

```
## [1] 0.2514933
```

```
db$il10_z <- scale(log(db$il10))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$il10_z <- scale(log(db_cc$il10))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

il10Model_hetero <- glmmTMB(
  il10_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(il10Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## il10_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion:                ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    453.8    513.9    -207.9    415.8      156
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id       (Intercept) 0.3505   0.592
##   Residual                NA      NA
## Number of obs: 175, groups: id, 79
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -1.85015    2.59368  -0.713  0.47564
## t1              0.46268    0.23339   1.982  0.04743 *
## t2              0.16637    0.41402   0.402  0.68780
## GROUP1         -0.77243    0.40082  -1.927  0.05397 .
## GROUP2         -0.80330    0.34088  -2.357  0.01845 *
## GROUP3         -0.78655    0.43272  -1.818  0.06912 .
## Age              0.02522    0.03341   0.755  0.45033
## SexM           -0.25444    0.16861  -1.509  0.13130
## CIRS_severity_0 -0.06209    0.17071  -0.364  0.71605
## t1:GROUP1       0.48410    0.39728   1.219  0.22301
## t2:GROUP1       0.67471    0.48864   1.381  0.16735
## t1:GROUP2       0.65586    0.32319   2.029  0.04242 *
## t2:GROUP2       1.19283    0.41732   2.858  0.00426 **
## t1:GROUP3       0.32143    0.40655   0.791  0.42916
## t2:GROUP3       0.73850    0.47892   1.542  0.12307
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.1946    0.1193  -1.631  0.10293
## t1            -0.6848    0.2376  -2.882  0.00395 **
## t2            -0.1722    0.1696  -1.016  0.30987
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

il10Model_homo <- glmmTMB(
  il10_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(il10Model_homo, il10Model_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## il10Model_homo: il10_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severi
ty_0 + , zi=~0, disp=~1
## il10Model_homo:      (1 | id), zi=~0, disp=~t
## il10Model_hetero: il10_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## il10Model_hetero:      (1 | id), zi=~0, disp=~t
##
##          Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Chi
sq)
## il10Model_homo    17 460.55 514.35 -213.28   426.55
## il10Model_hetero  19 453.81 513.94 -207.90   415.81 10.747      2  0.004
638 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

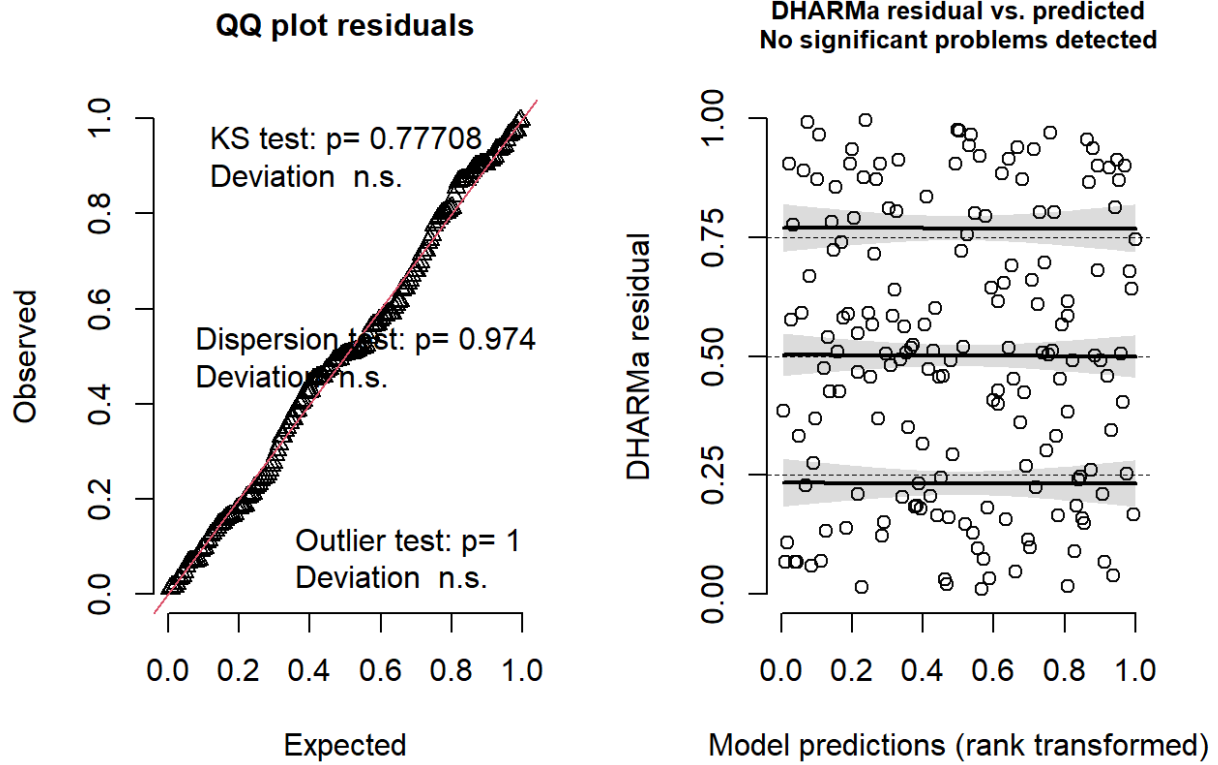
```

```

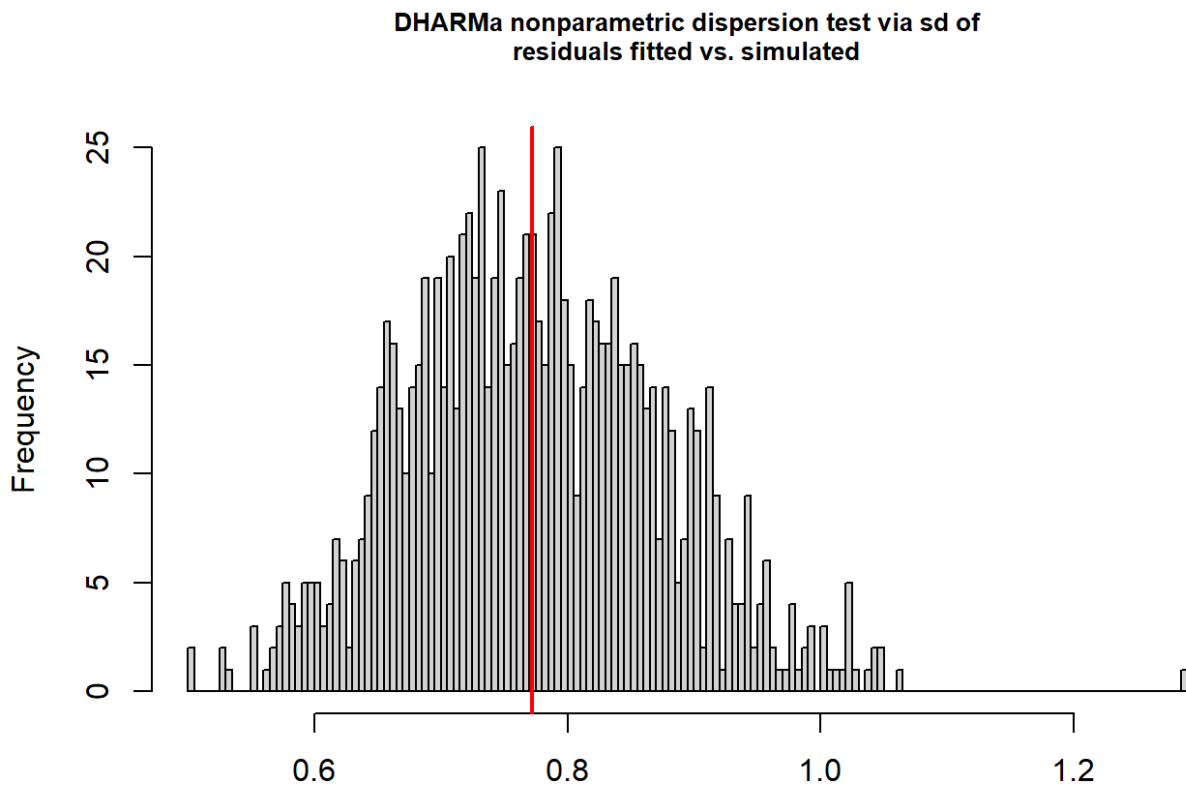
sim_res <- simulateResiduals(il10Model_hetero, n = 1000)
plot(sim_res)

```

DHARMA residual

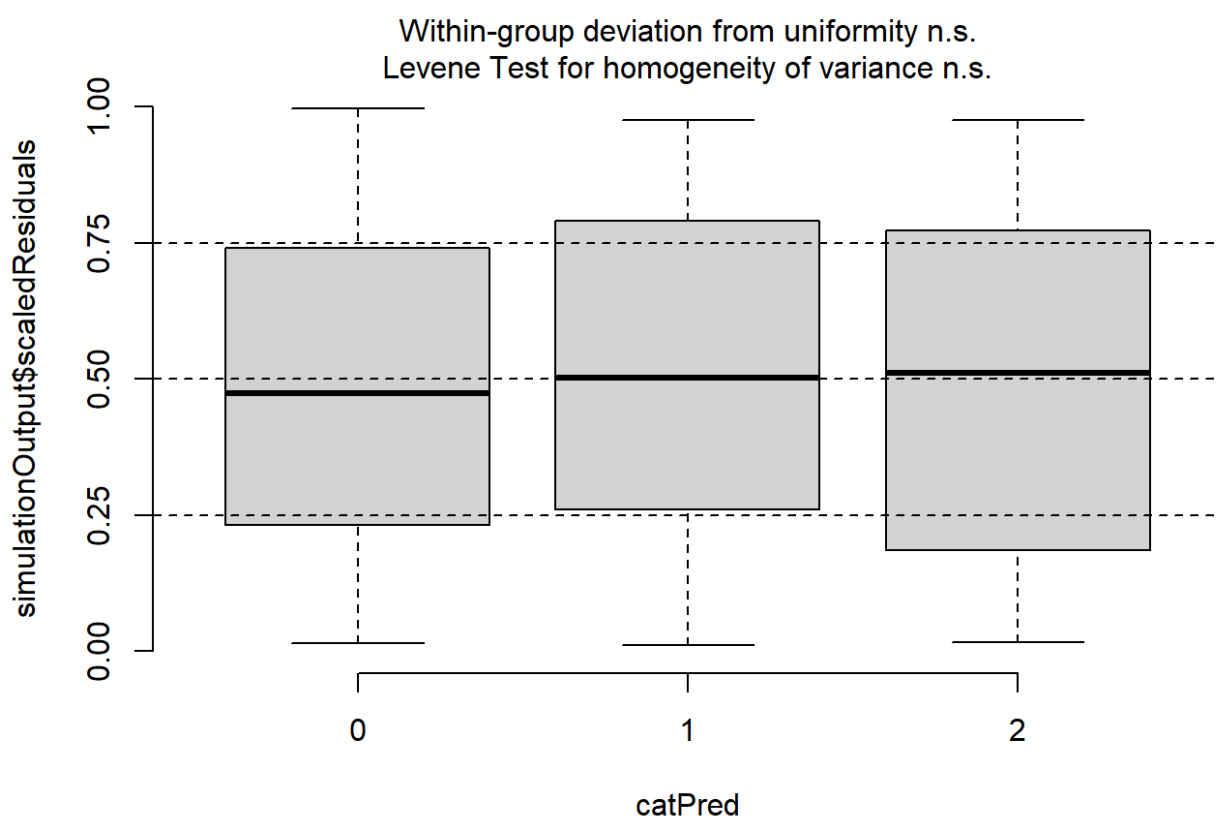


```
testDispersion(sim_res)
```



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 0.99788, p-value = 0.974
## alternative hypothesis: two.sided
```

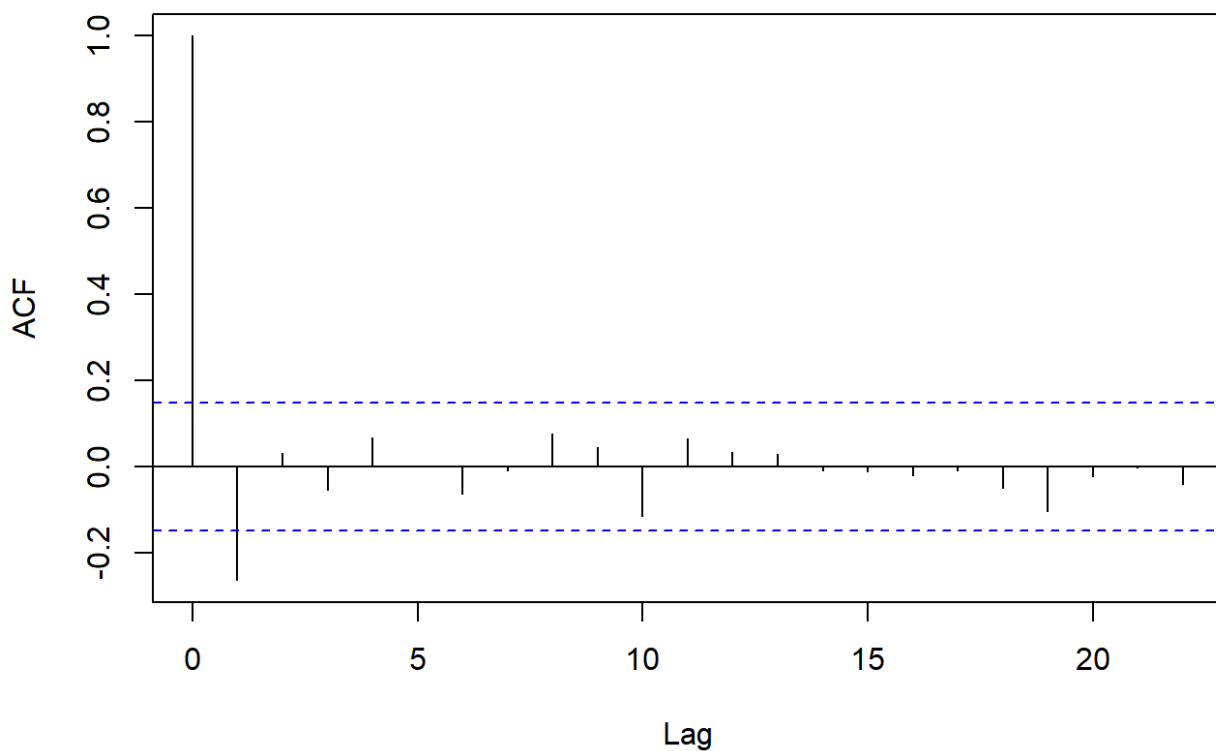
```
testQuantiles(sim_res, db_cc$t)
```



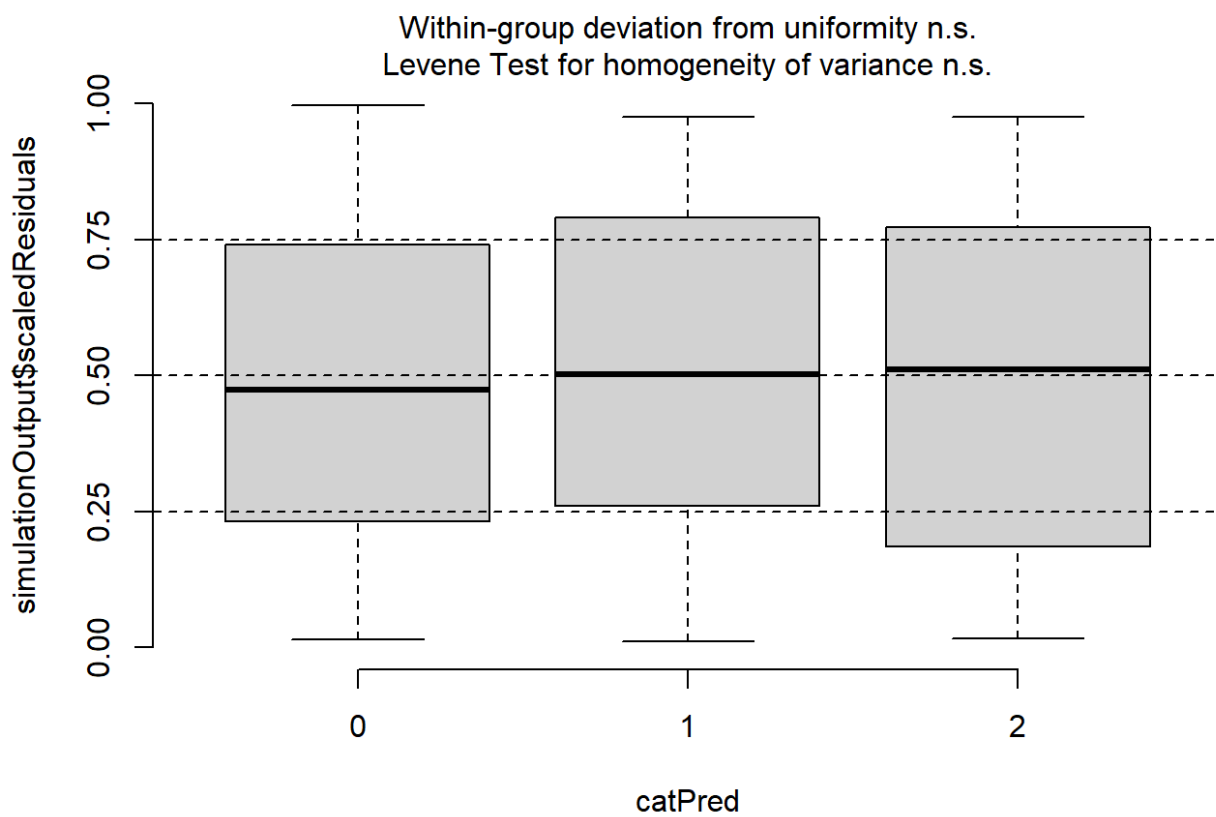
```
## NULL
```

```
acf(residuals(il10Model_hetero))
```

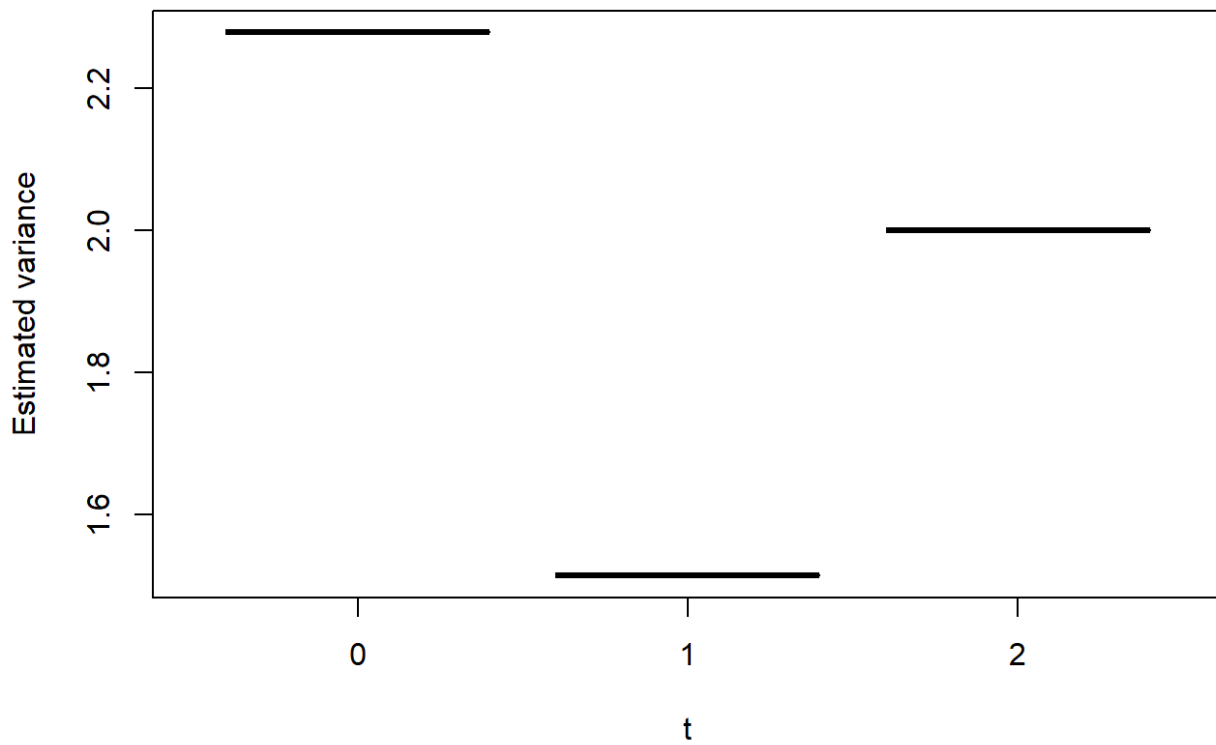
Series residuals(il10Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

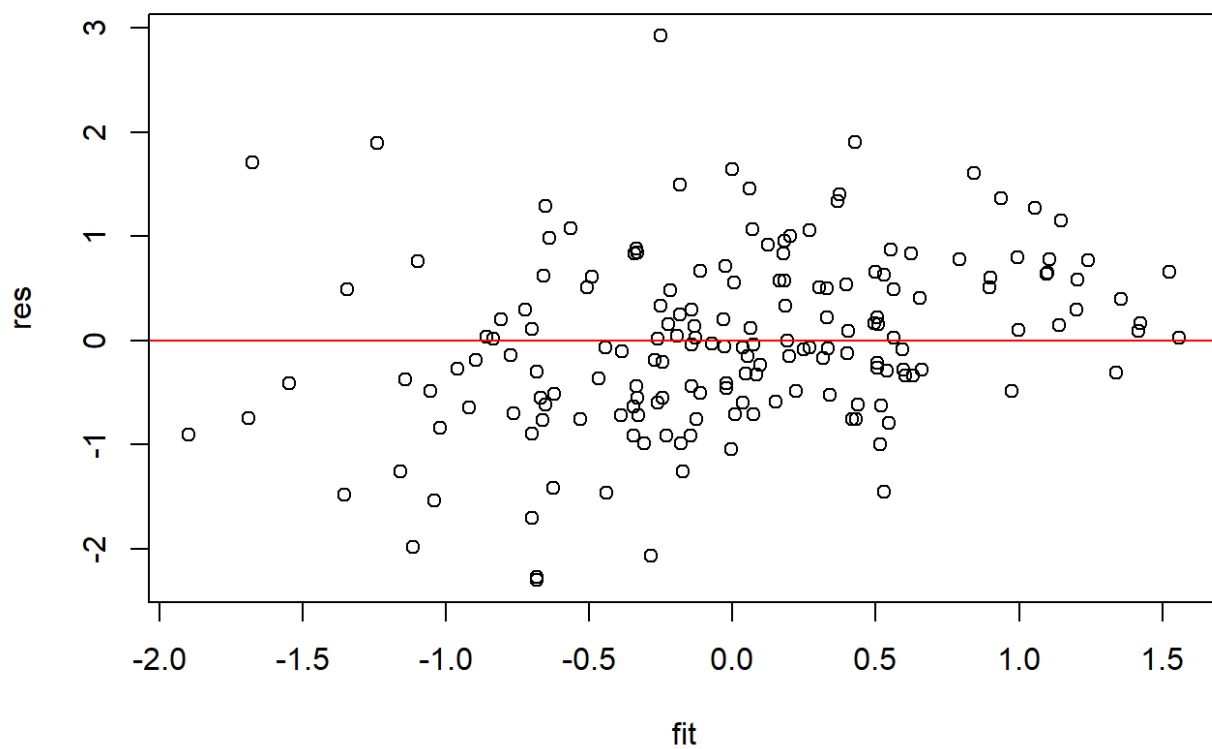



```
pred_disp <- predict(il10Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(il10Model_hetero, type="pearson")
fit <- fitted(il10Model_hetero)

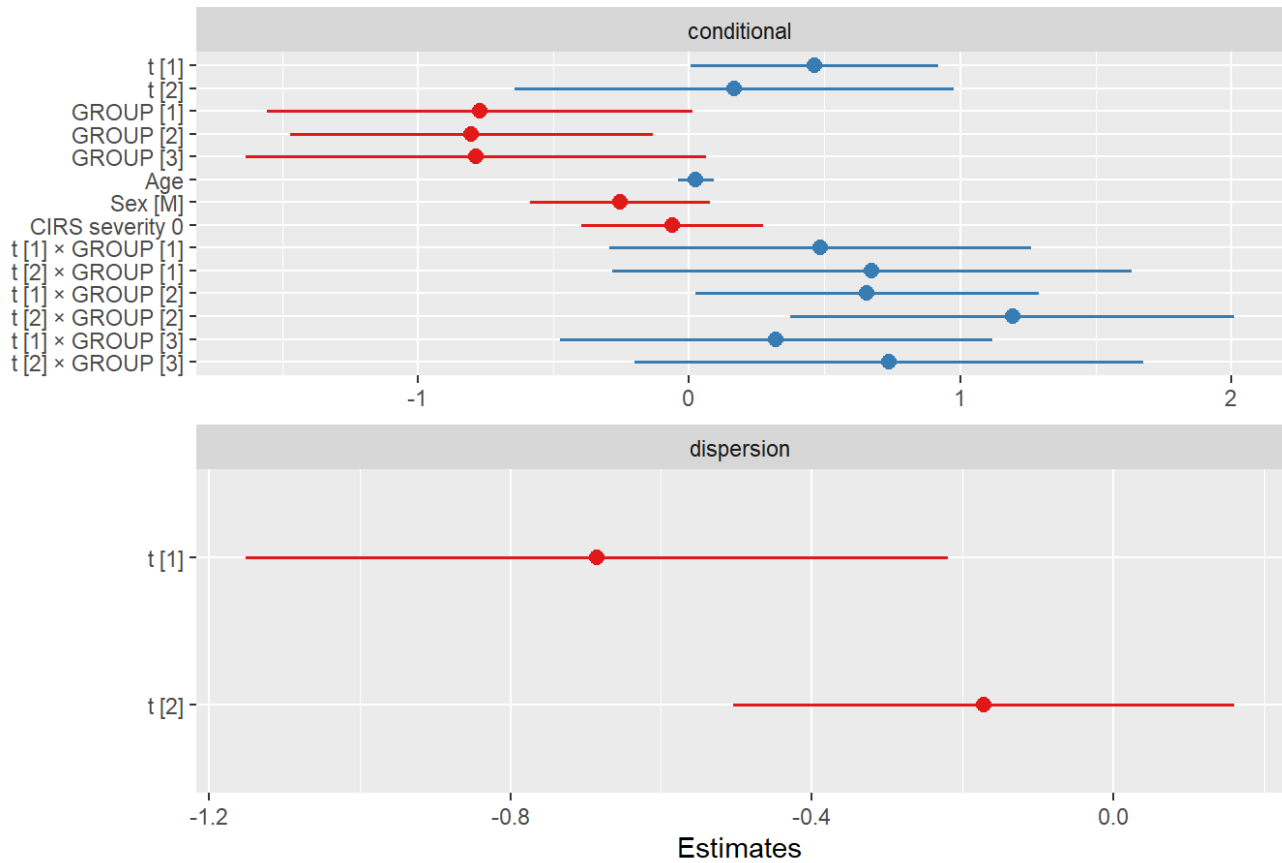
plot(fit, res)
abline(h=0, col="red")
```



14.1 Visualizzazione del modello

```
plot_model(il10Model_hetero)
```

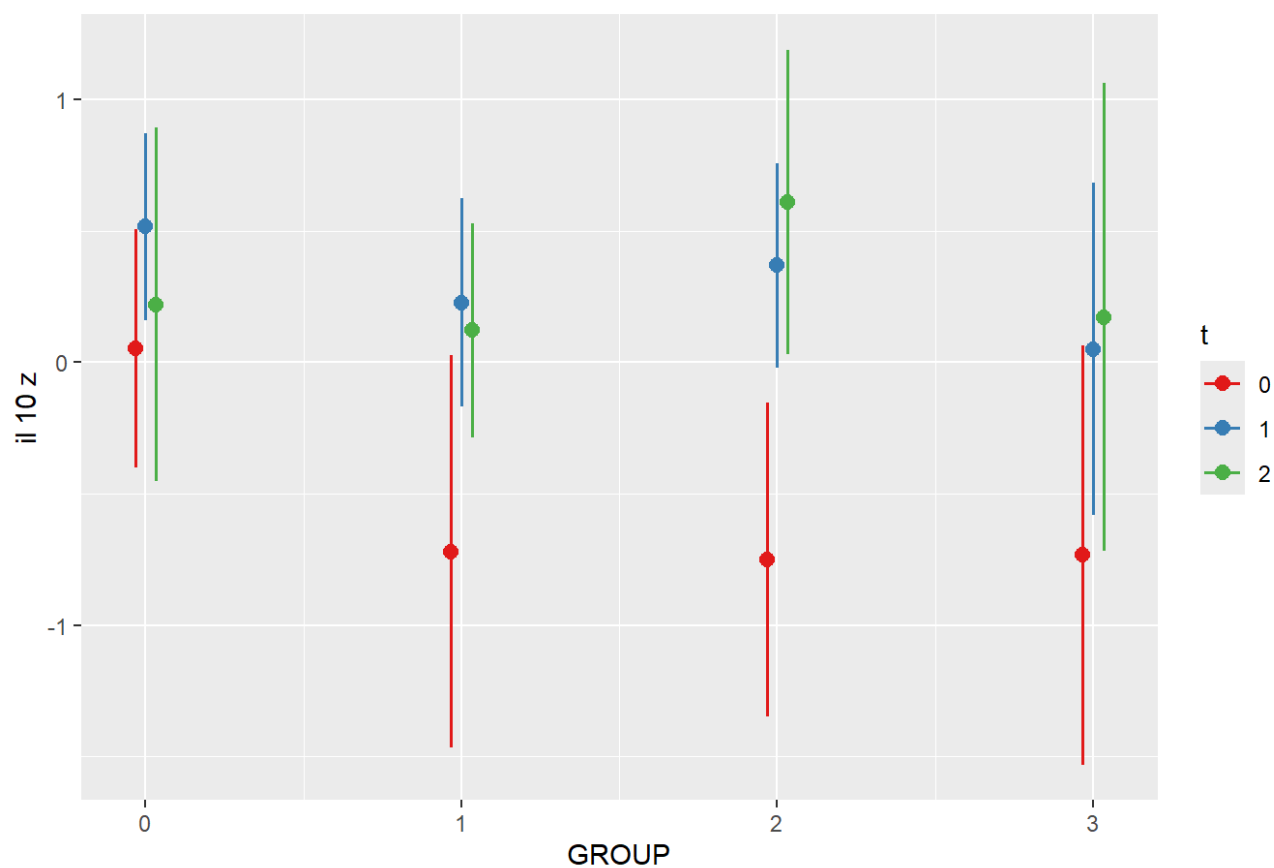
il 10 z



```
plot_model(il10Model_hetero, type = "pred",
           terms = c("GROUP", "t"),
           pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

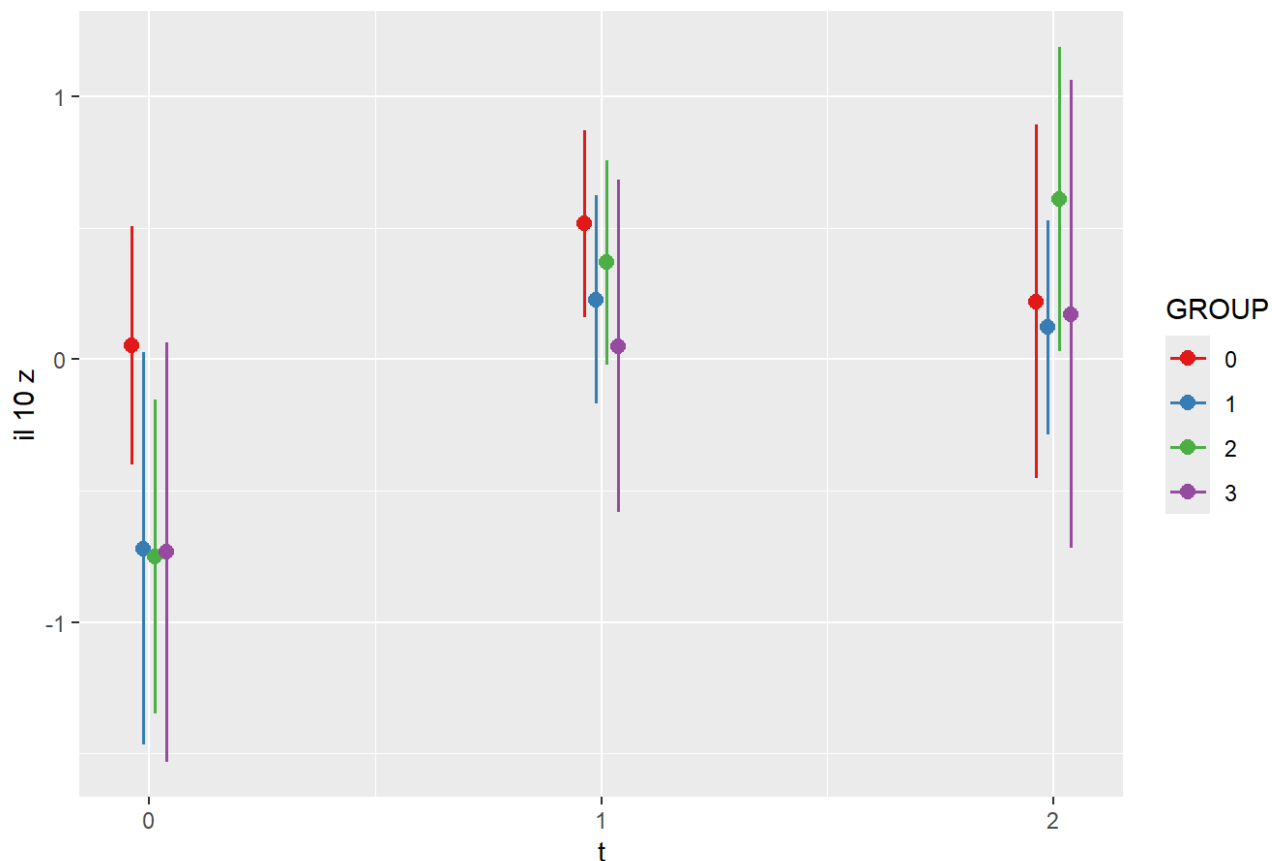
Predicted values of il 10 z



```
plot_model(il10Model_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of il 10 z

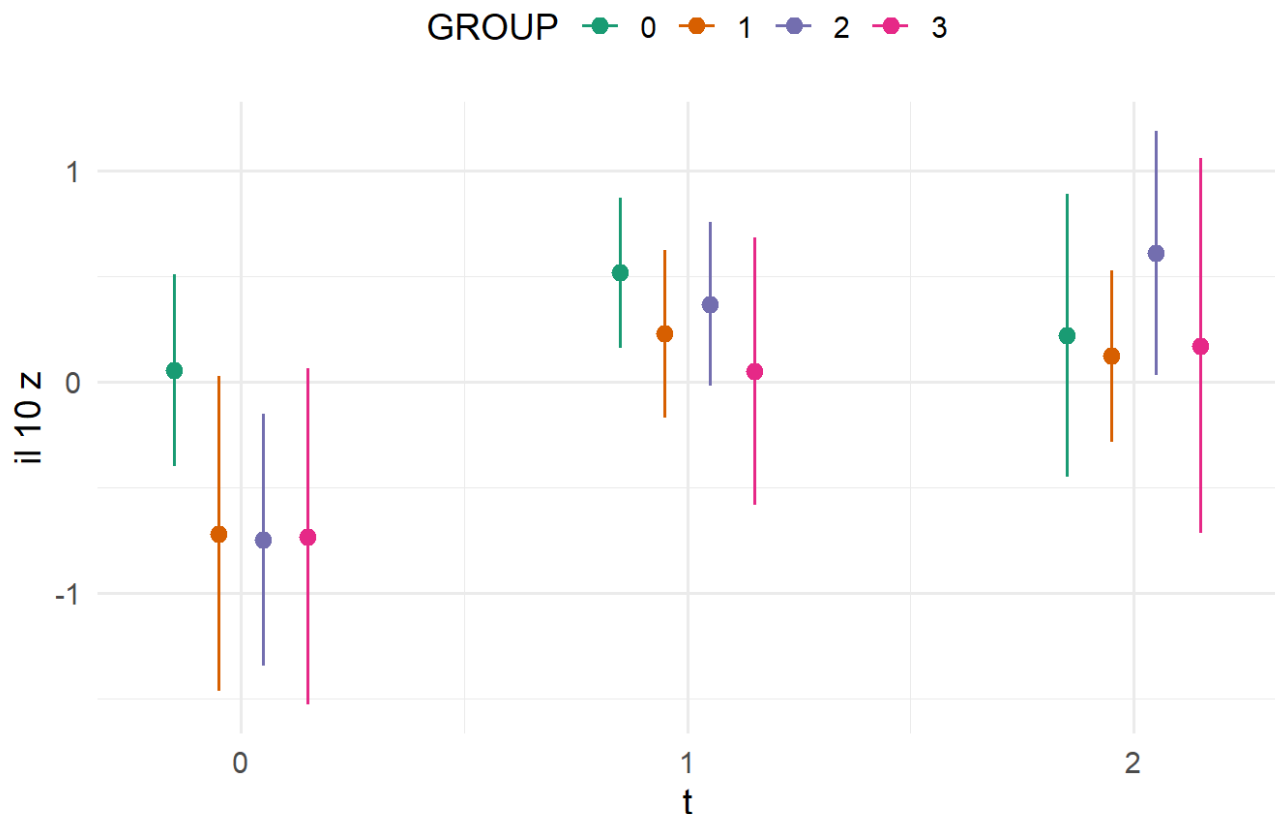


```
plot_model(il10Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

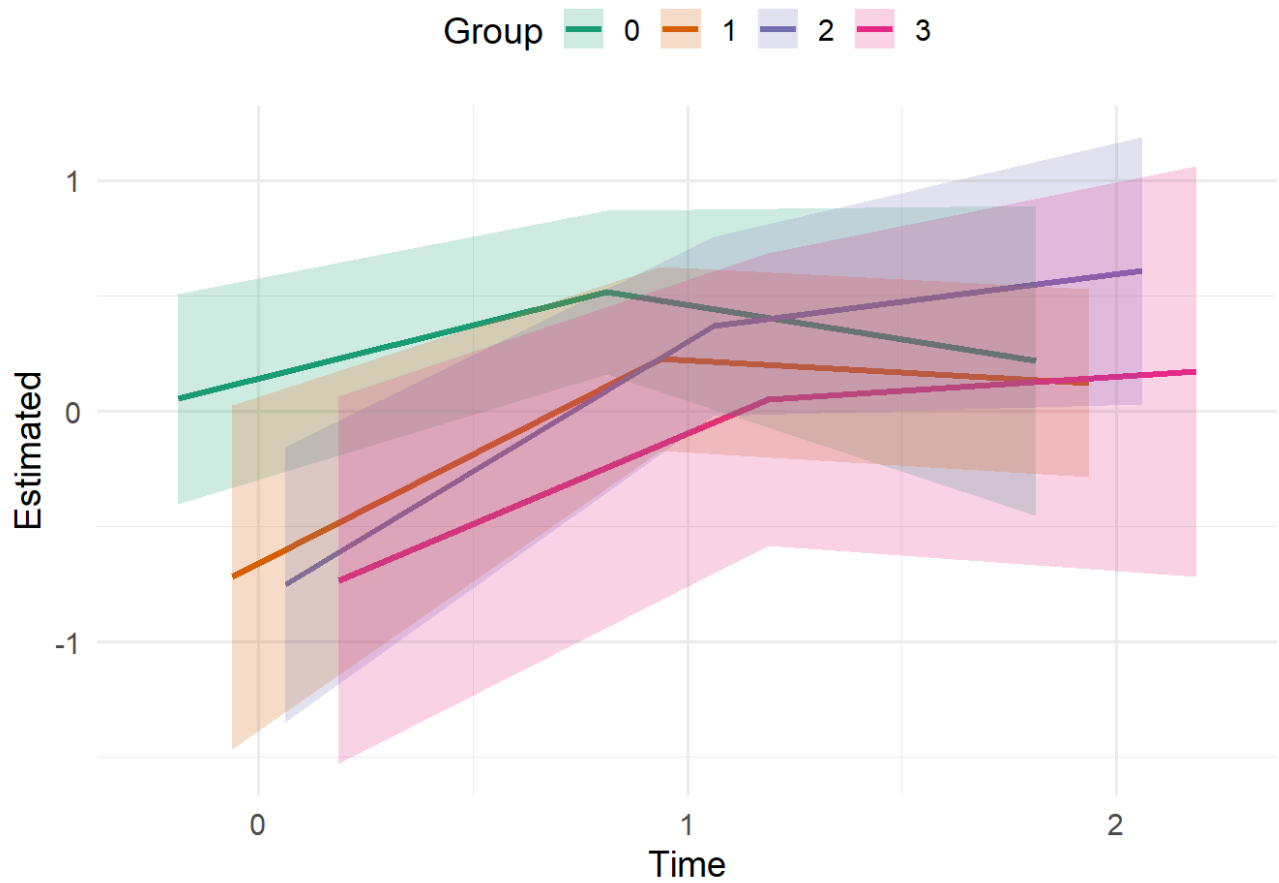
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of il 10 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(il10Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



14.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(il110Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0      -0.07419  0.216 Inf    -0.4977    0.3493
##   1      -0.84662  0.381 Inf    -1.5941   -0.0991
##   2      -0.87749  0.304 Inf    -1.4734   -0.2816
##   3      -0.86074  0.390 Inf    -1.6257   -0.0958
##
## t = 1:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0       0.38848  0.160 Inf     0.0752    0.7018
##   1       0.10016  0.197 Inf    -0.2854    0.4857
##   2       0.24104  0.195 Inf    -0.1407    0.6228
##   3      -0.07663  0.302 Inf    -0.6688    0.5155
##
## t = 2:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0       0.09217  0.330 Inf    -0.5547    0.7391
##   1      -0.00555  0.200 Inf    -0.3979    0.3868
##   2       0.48171  0.288 Inf    -0.0821    1.0455
##   3       0.04412  0.436 Inf    -0.8112    0.8995
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(il10Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##   contrast          estimate      SE   df z.ratio p.value
##   GROUP0 - GROUP1    0.3862  0.266 Inf    1.451  0.4673
##   GROUP0 - GROUP2    0.1871  0.239 Inf    0.784  0.8617
##   GROUP0 - GROUP3    0.4332  0.307 Inf    1.411  0.4925
##   GROUP1 - GROUP2   -0.1991  0.263 Inf   -0.758  0.8732
##   GROUP1 - GROUP3    0.0471  0.367 Inf    0.128  0.9992
##   GROUP2 - GROUP3    0.2462  0.340 Inf    0.725  0.8873
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(il10Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```



```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1    0.7724 0.401 Inf   1.927  0.2166
## GROUP0 - GROUP2    0.8033 0.341 Inf   2.357  0.0857
## GROUP0 - GROUP3    0.7865 0.433 Inf   1.818  0.2649
## GROUP1 - GROUP2    0.0309 0.437 Inf   0.071  0.9999
## GROUP1 - GROUP3    0.0141 0.528 Inf   0.027  1.0000
## GROUP2 - GROUP3   -0.0168 0.480 Inf  -0.035  1.0000
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1    0.2883 0.271 Inf   1.064  0.7114
## GROUP0 - GROUP2    0.1474 0.239 Inf   0.618  0.9263
## GROUP0 - GROUP3    0.4651 0.325 Inf   1.431  0.4801
## GROUP1 - GROUP2   -0.1409 0.274 Inf  -0.515  0.9556
## GROUP1 - GROUP3    0.1768 0.385 Inf   0.459  0.9679
## GROUP2 - GROUP3    0.3177 0.354 Inf   0.897  0.8065
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1    0.0977 0.389 Inf   0.251  0.9945
## GROUP0 - GROUP2   -0.3895 0.357 Inf  -1.091  0.6949
## GROUP0 - GROUP3    0.0481 0.411 Inf   0.117  0.9994
## GROUP1 - GROUP2   -0.4873 0.347 Inf  -1.404  0.4969
## GROUP1 - GROUP3   -0.0497 0.484 Inf  -0.103  0.9996
## GROUP2 - GROUP3    0.4376 0.452 Inf   0.968  0.7679
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(ill10Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.463 0.233 Inf  -1.982  0.1165
## t0 - t2      -0.166 0.414 Inf  -0.402  0.9148
## t1 - t2       0.296 0.327 Inf   0.907  0.6356
##
## GROUP = 1:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.947 0.345 Inf  -2.746  0.0166
## t0 - t2      -0.841 0.409 Inf  -2.057  0.0990
## t1 - t2       0.106 0.222 Inf   0.476  0.8828
##
## GROUP = 2:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -1.119 0.294 Inf  -3.805  0.0004
## t0 - t2      -1.359 0.415 Inf  -3.274  0.0030
## t1 - t2      -0.241 0.290 Inf  -0.829  0.6851
##
## GROUP = 3:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.784 0.385 Inf  -2.035  0.1040
## t0 - t2      -0.905 0.522 Inf  -1.732  0.1931
## t1 - t2      -0.121 0.401 Inf  -0.301  0.9512
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

15 Model tnfa

```
bptest(lm(tnfa ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(tnfa ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 26.74, df = 14, p-value = 0.02081
```

```
vars <- c("tnfa","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

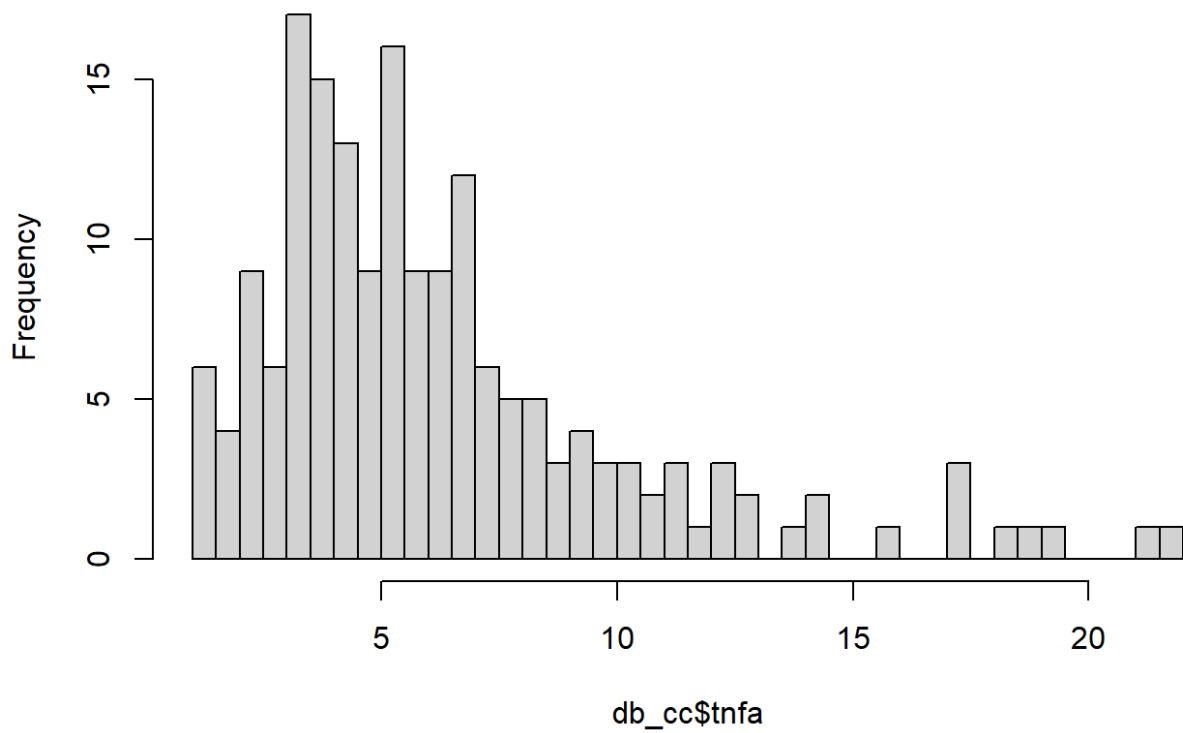
nrow(db); nrow(db_cc)
```

```
## [1] 243
```

```
## [1] 177
```

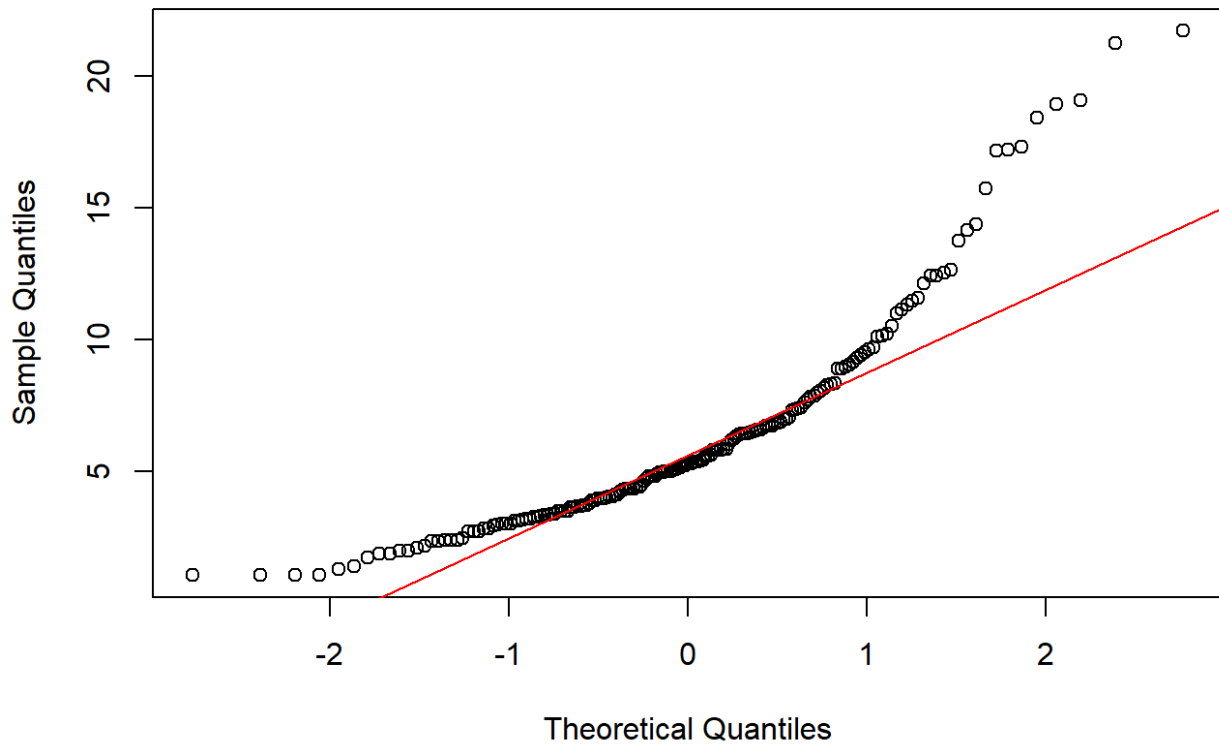
```
hist(db_cc$tnfa, breaks=30)
```

Histogram of db_cc\$tnfa



```
qqnorm(db_cc$tnfa); qqline(db_cc$tnfa, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$tnfa)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  1.084   3.511   5.310   6.332   7.744  21.702
```

```
min(db_cc$tnfa, na.rm=TRUE)
```

```
## [1] 1.083977
```

```
db$tnfa_z <- scale(log(db$tnfa))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$tnfa_z <- scale(log(db_cc$tnfa))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

tnfaModel_hetero <- glmmTMB(
  tnfa_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(tnfaModel_hetero)
```

```
## Family: gaussian ( identity )
## Formula:
## tnfa_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1 |
id)
## Dispersion: ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    412.0    472.3   -187.0    374.0     158
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id      (Intercept) 0.7145   0.8453
##   Residual              NA      NA
## Number of obs: 177, groups: id, 80
##
## Conditional model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.71211    3.11328   0.871   0.3837
## t1              0.26949    0.21282   1.266   0.2054
## t2              0.30392    0.40267   0.755   0.4504
## GROUP1         -0.72836    0.36388  -2.002   0.0453 *
## GROUP2         -0.26820    0.32044  -0.837   0.4026
## GROUP3         -1.23967    0.42709  -2.903   0.0037 **
## Age            -0.03281    0.04050  -0.810   0.4178
## SexM           -0.01822    0.20966  -0.087   0.9308
## CIRS_severity_0 -0.09546    0.12991  -0.735   0.4625
## t1:GROUP1       0.38983    0.26599   1.466   0.1428
## t2:GROUP1       0.43278    0.34630   1.250   0.2114
## t1:GROUP2       0.13450    0.21106   0.637   0.5240
## t2:GROUP2       0.44668    0.28597   1.562   0.1183
## t1:GROUP3       0.67865    0.26788   2.533   0.0113 *
## t2:GROUP3       0.78424    0.32302   2.428   0.0152 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.64539    0.13187  -4.894 9.87e-07 ***
## t1          -0.66565    0.34404  -1.935   0.053 .
## t2          -0.08323    0.18138  -0.459   0.646
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

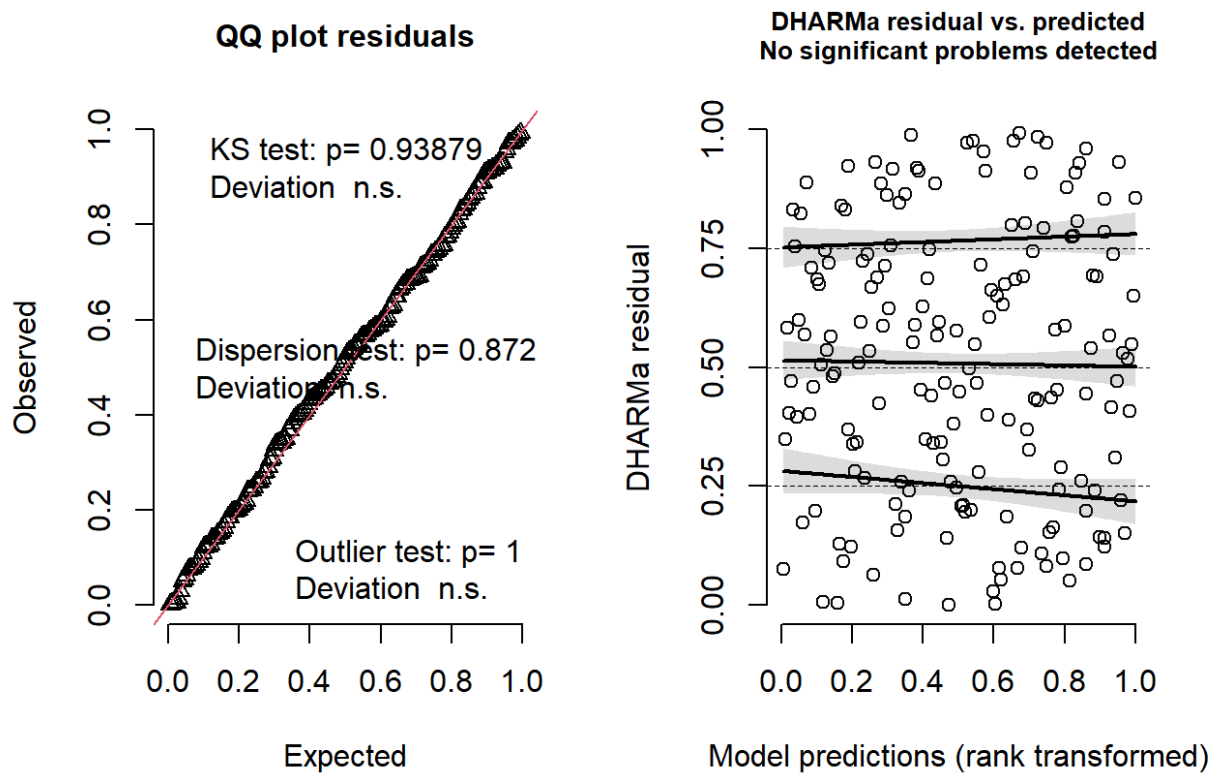
```
tnfaModel_homo <- glmmTMB(
  tnfa_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(tnfaModel_homo, tnfaModel_hetero, test="Chisq")
```

```
## Data: db_cc
## Models:
## tnfaModel_homo: tnfa_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severi
ty_0 + , zi=~0, disp=~1
## tnfaModel_homo:      (1 | id), zi=~0, disp=~t
## tnfaModel_hetero: tnfa_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## tnfaModel_hetero:      (1 | id), zi=~0, disp=~t
##
      Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chi
sq)
## tnfaModel_homo    17 414.25 468.25 -190.12   380.25
## tnfaModel_hetero  19 411.99 472.34 -187.00   373.99 6.2605      2    0.04
371 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

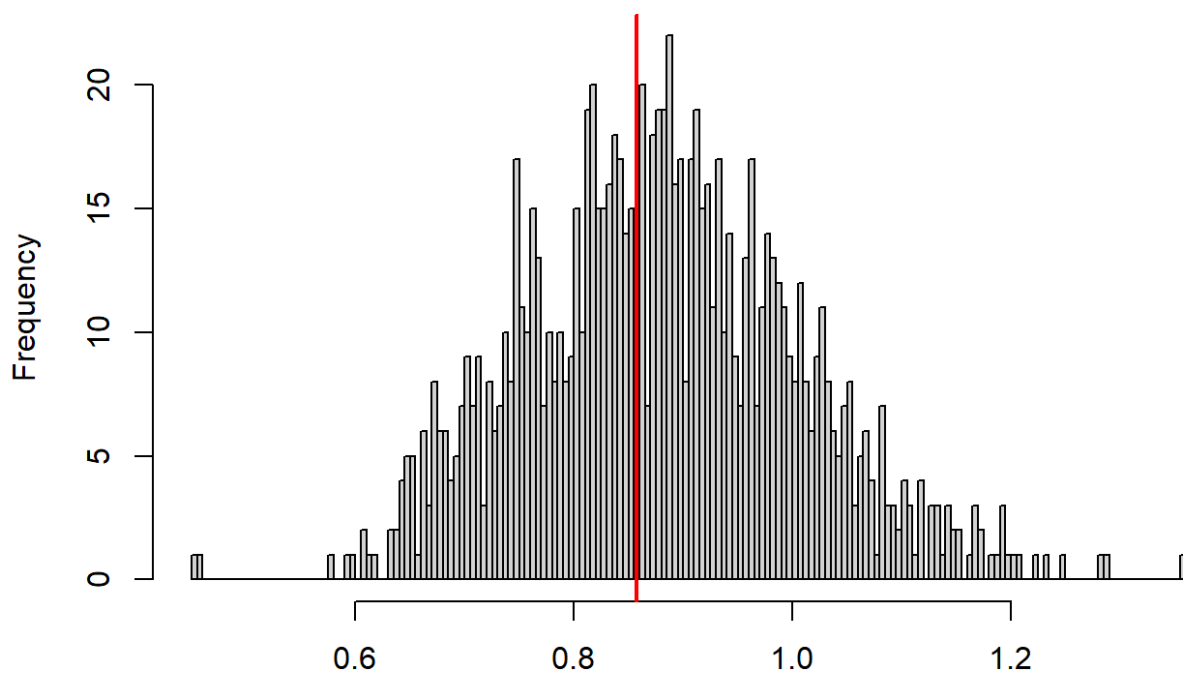
```
sim_res <- simulateResiduals(tnfaModel_hetero, n = 1000)
plot(sim_res)
```

DHARMA residual



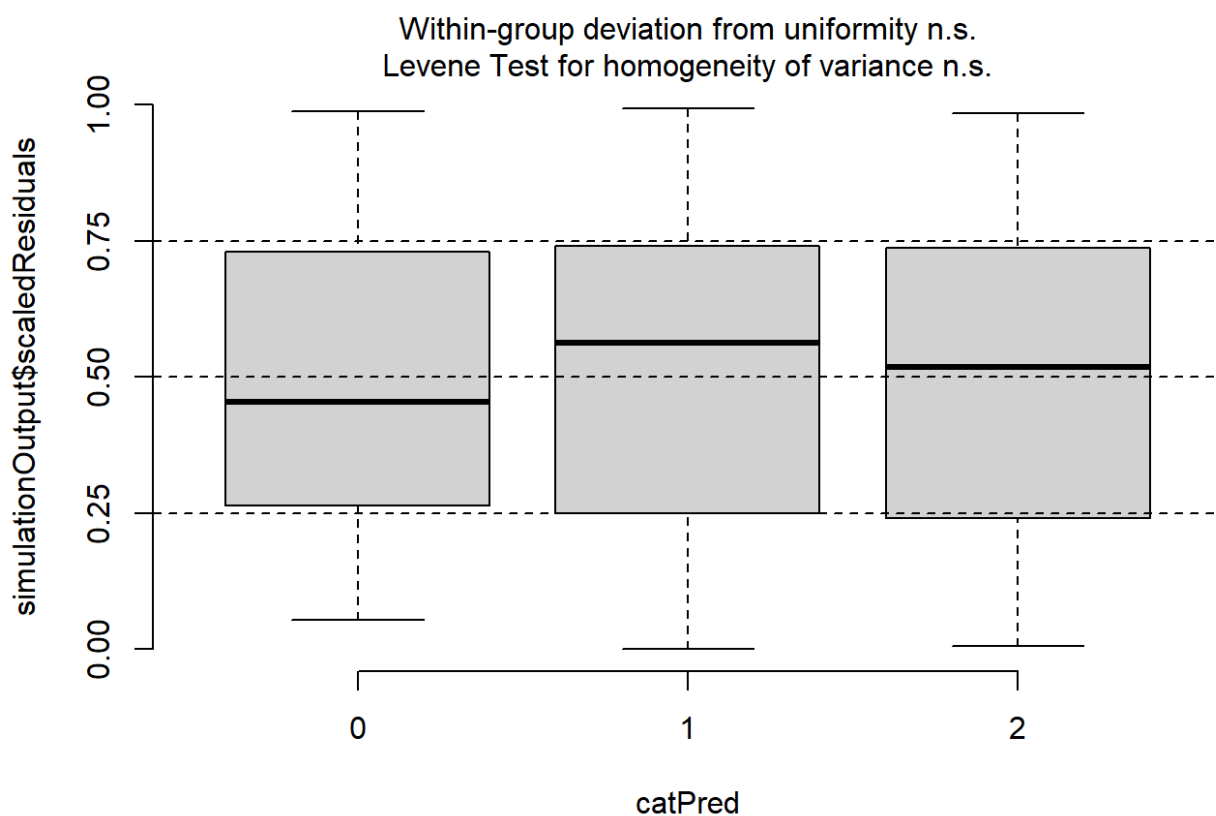
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated




```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 0.97423, p-value = 0.872
## alternative hypothesis: two.sided
```

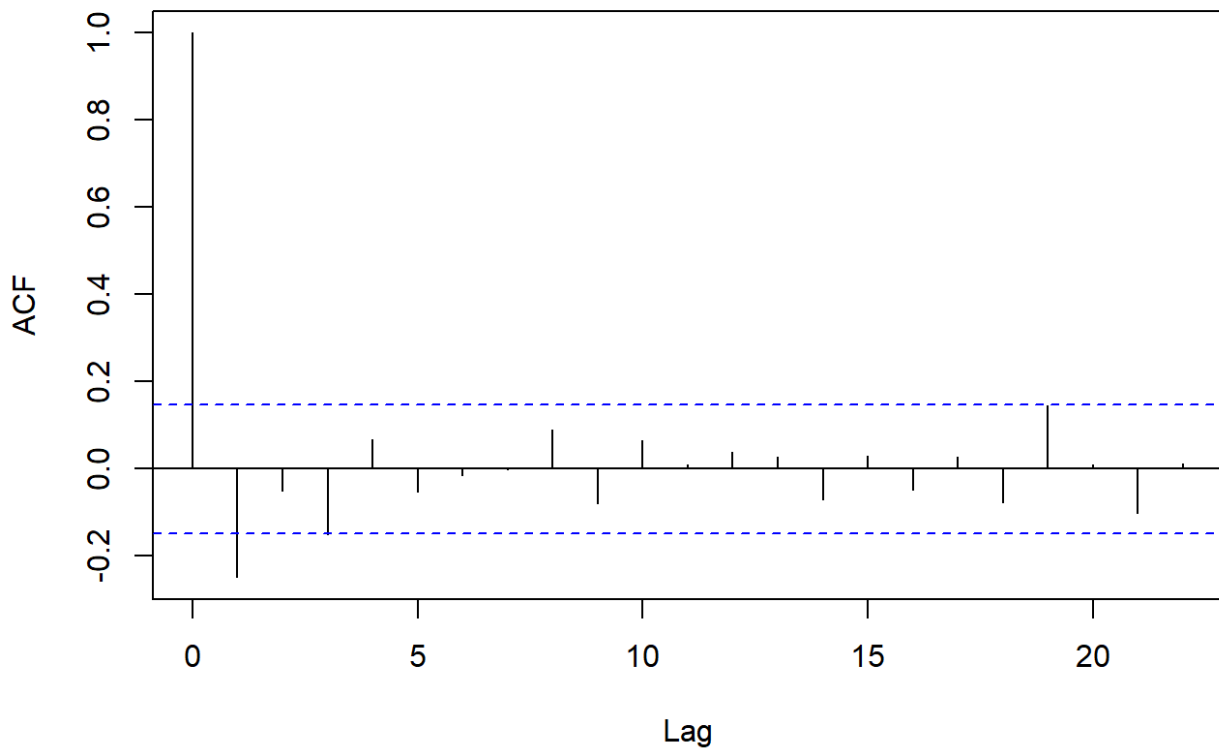
```
testQuantiles(sim_res, db_cc$t)
```



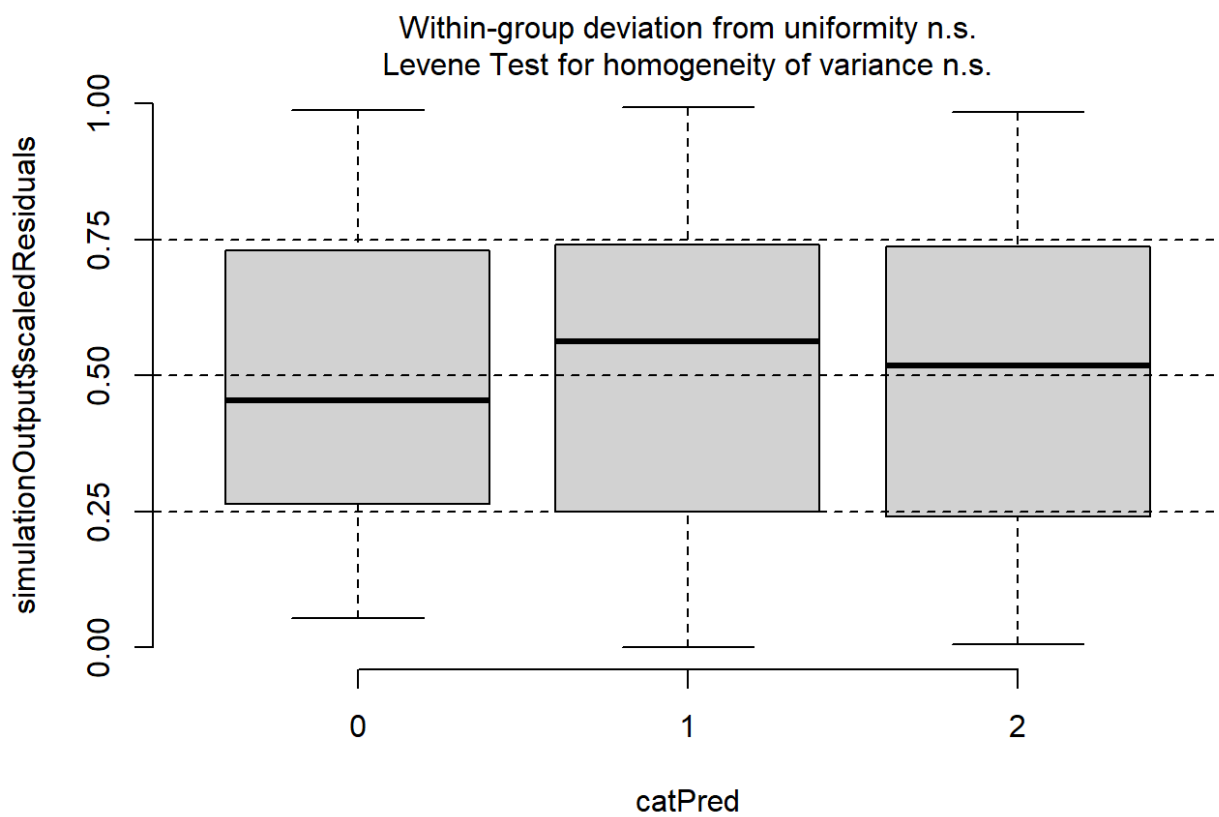
```
## NULL
```

```
acf(residuals(tnfaModel_hetero))
```

Series residuals(tnfaModel_hetero)



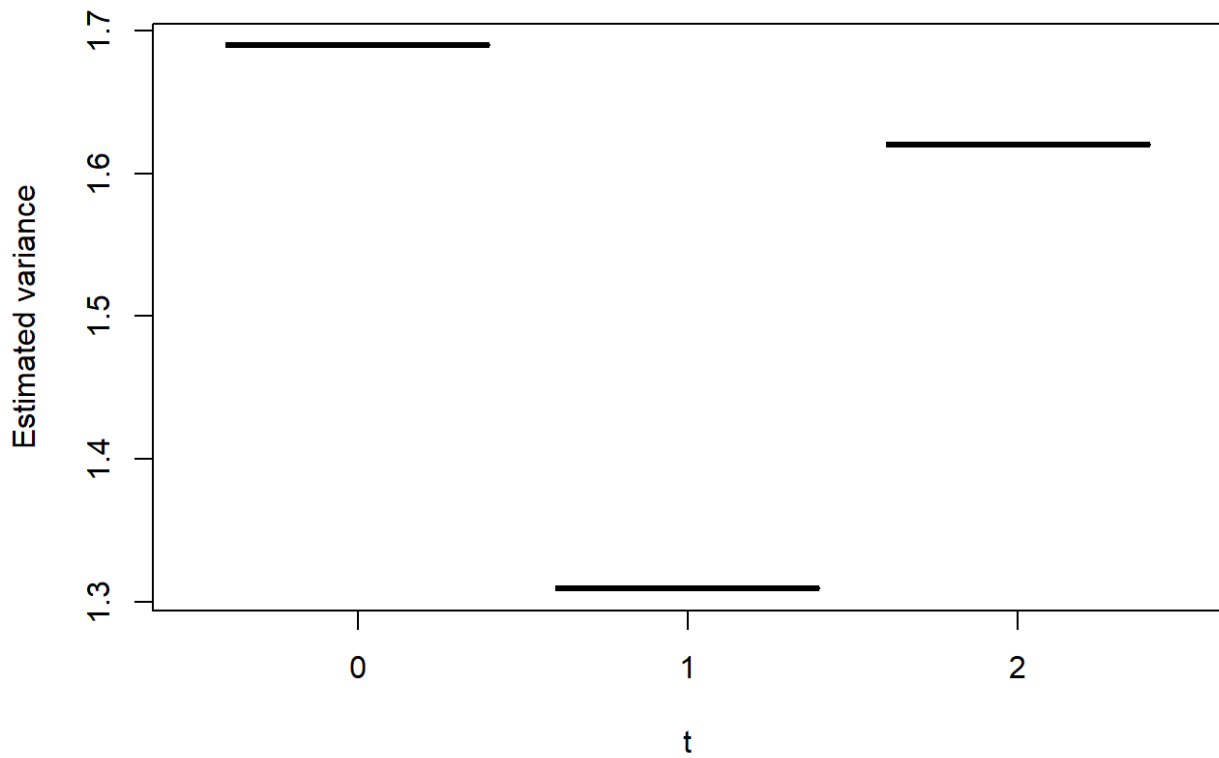
```
plotResiduals(sim_res, db_cc$t)
```



```

pred_disp <- predict(tnfaModel_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")

```

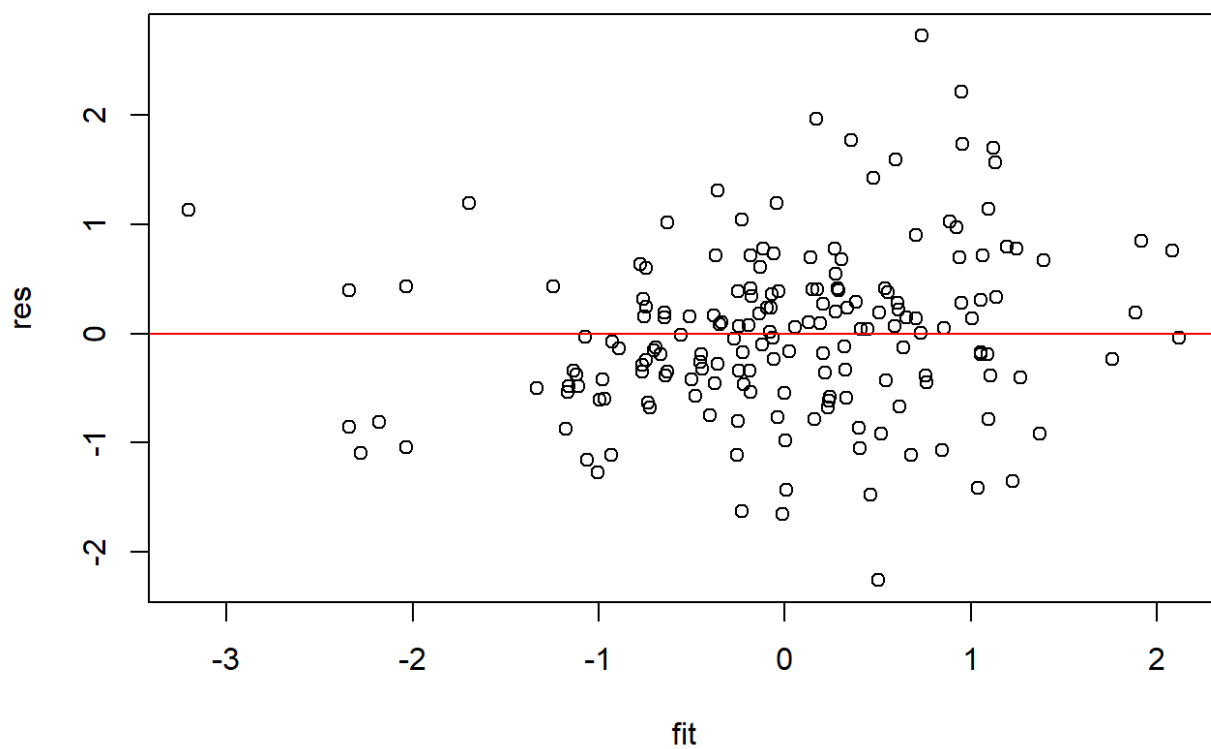


```

res <- residuals(tnfaModel_hetero, type="pearson")
fit <- fitted(tnfaModel_hetero)

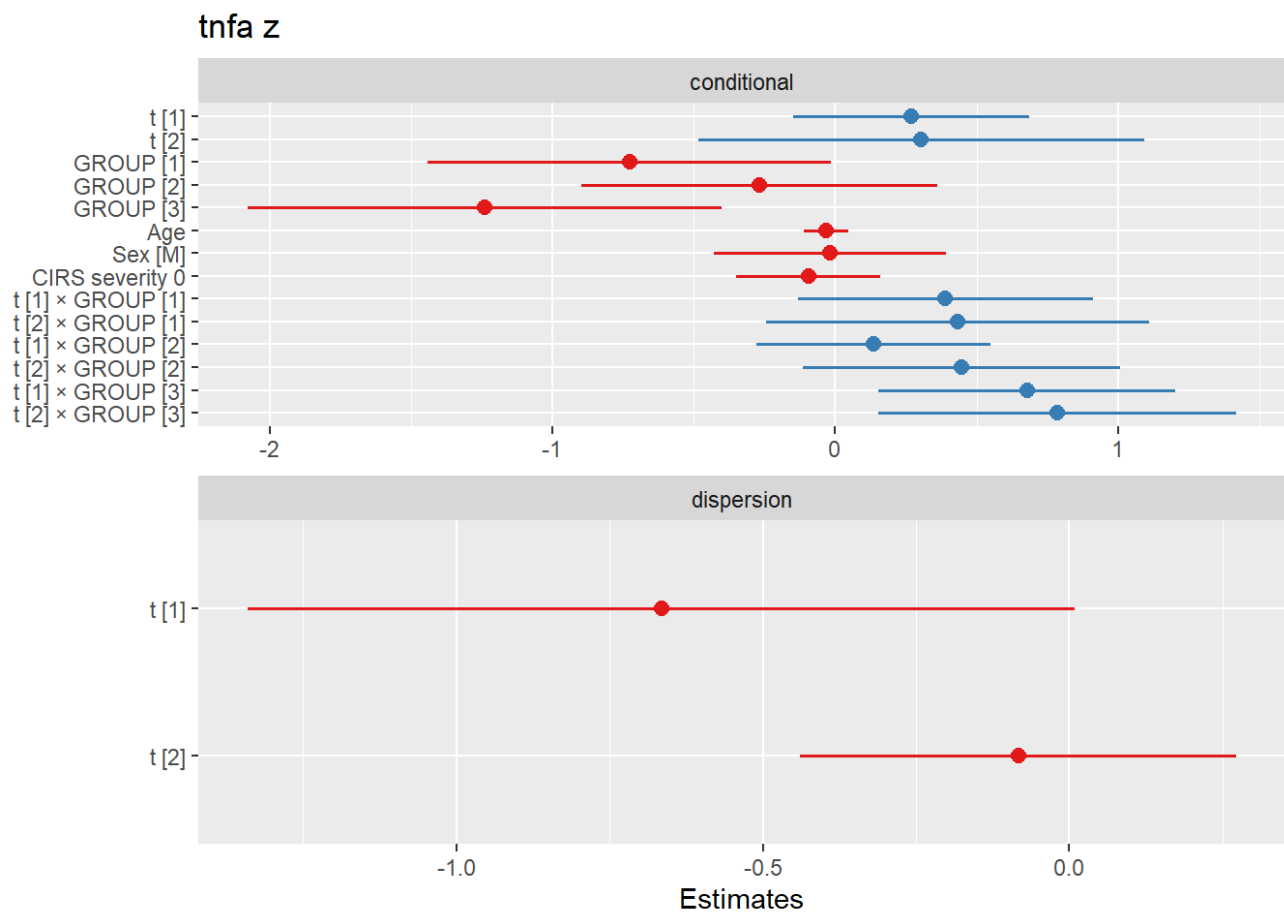
plot(fit, res)
abline(h=0, col="red")

```



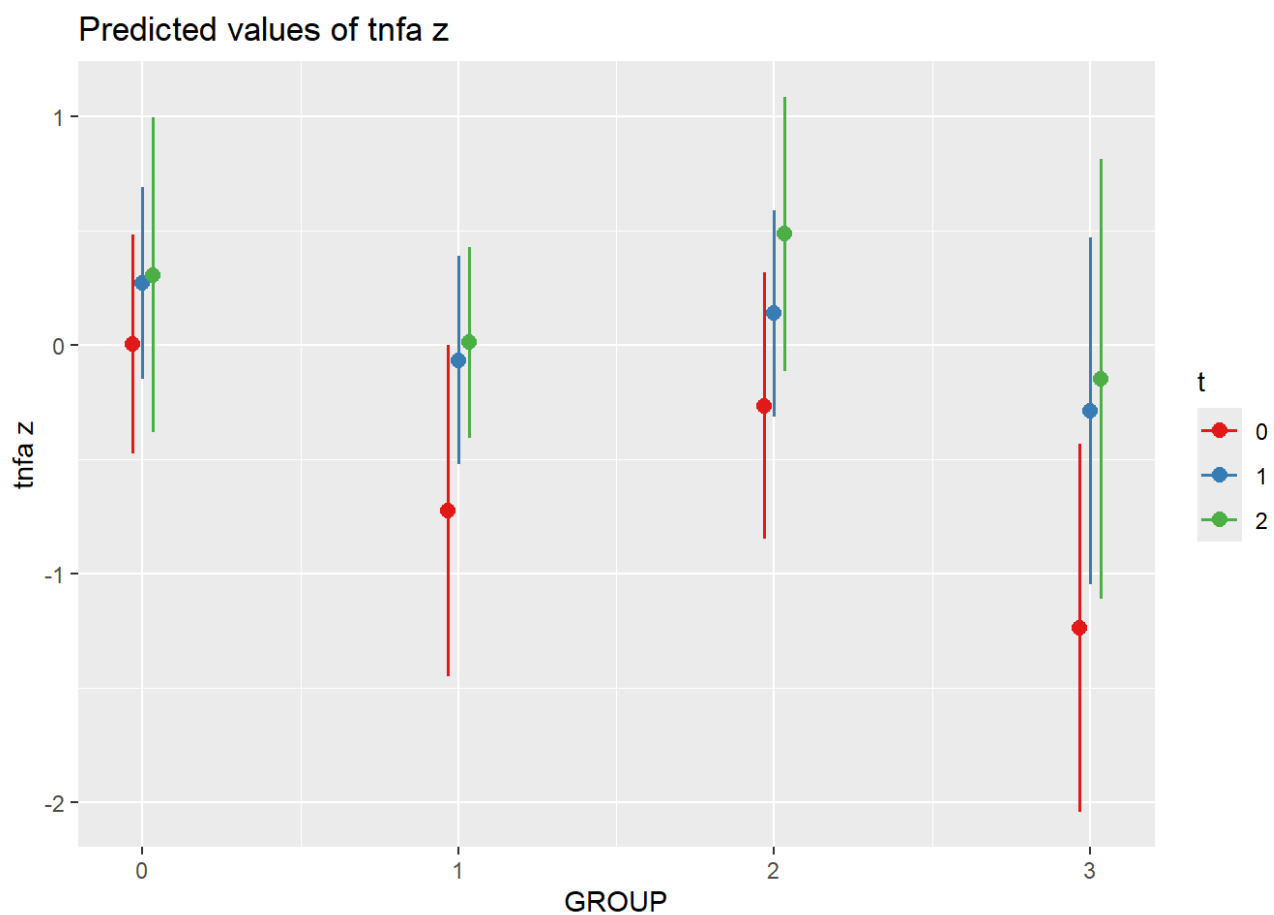
15.1 Visualizzazione del modello

```
plot_model(tnfaModel_hetero)
```



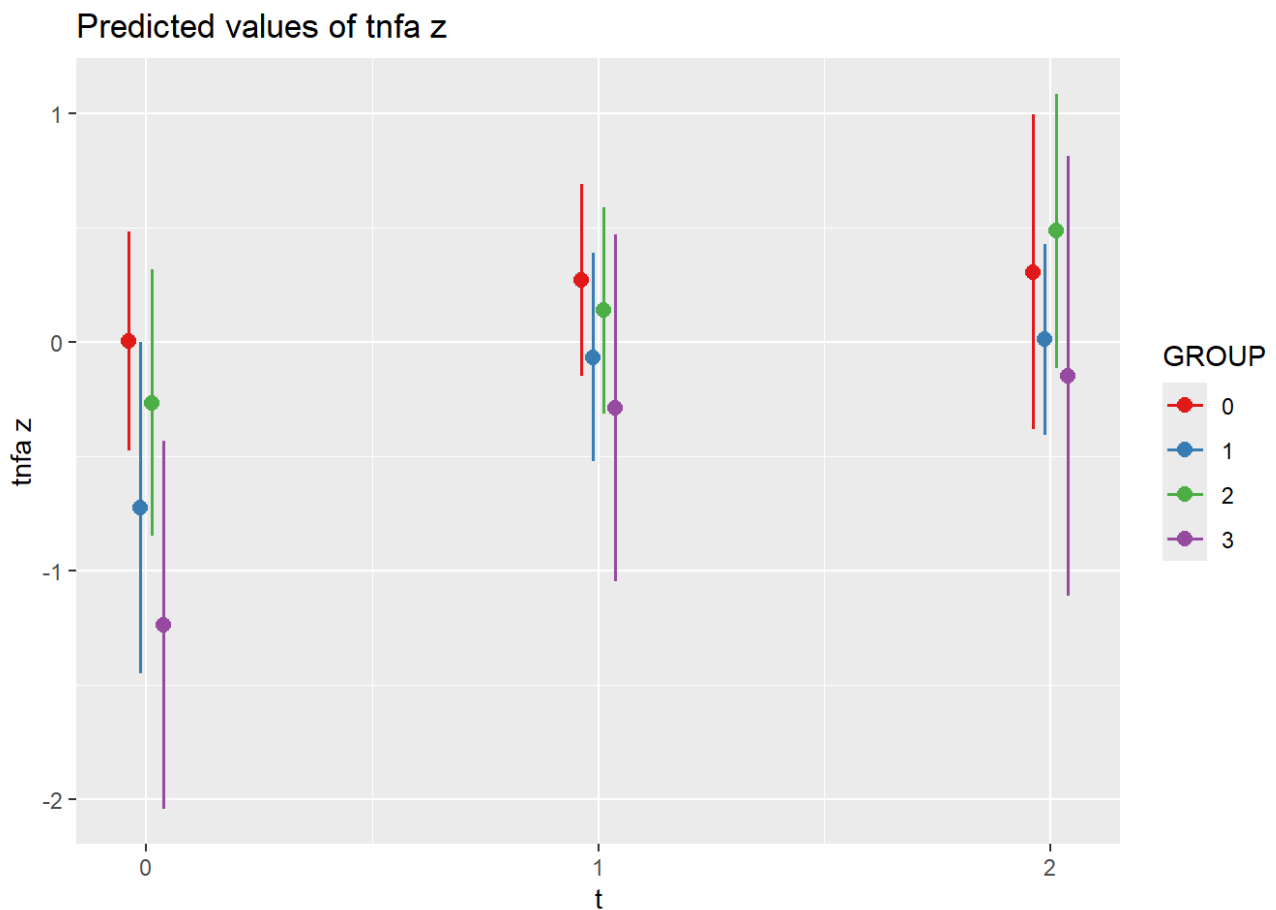
```
plot_model(tnfaModel_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```



```
plot_model(tnfaModel_hetero, type = "pred",  
            terms = c("t", "GROUP"),  
            pred.type = "fe")
```

```
## Ignoring unknown labels:  
## • linetype : "GROUP"  
## • shape : "GROUP"
```

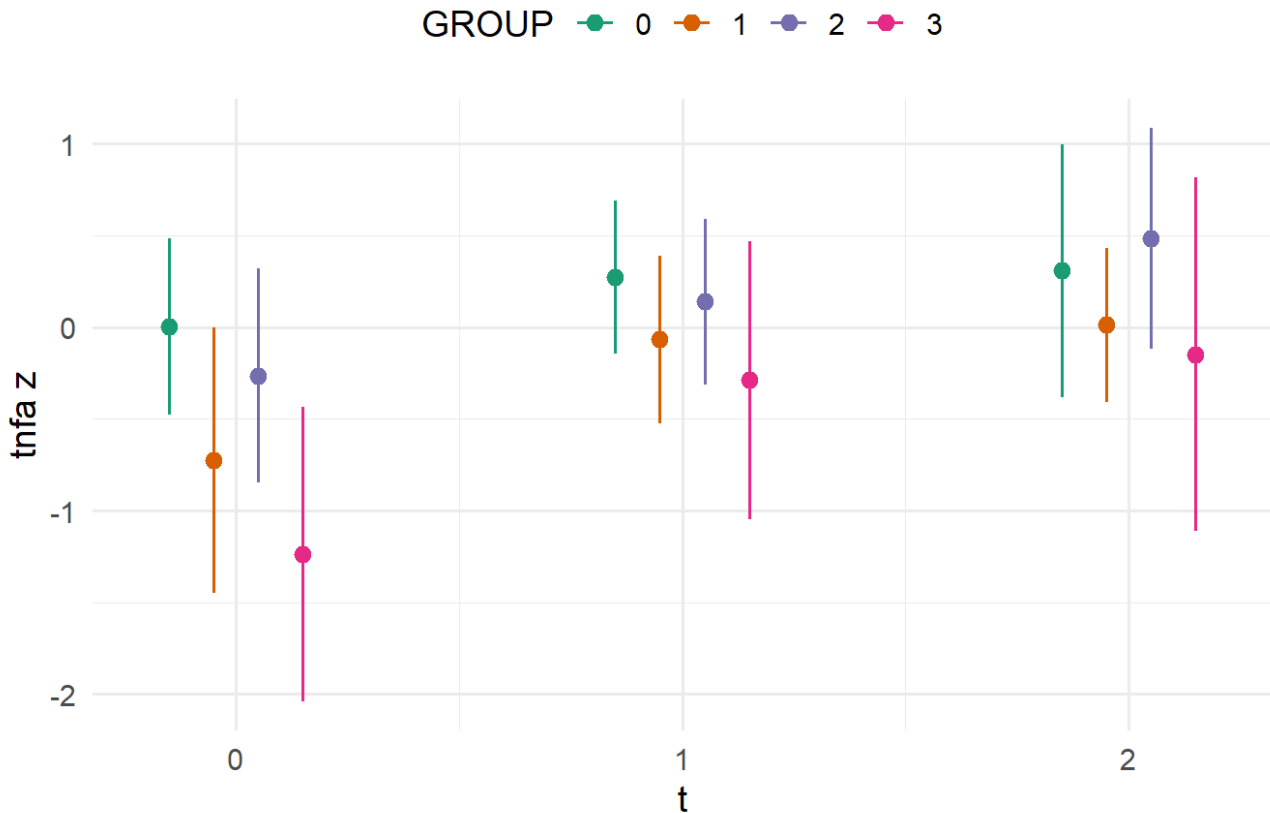


```
plot_model(tnfaModel_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

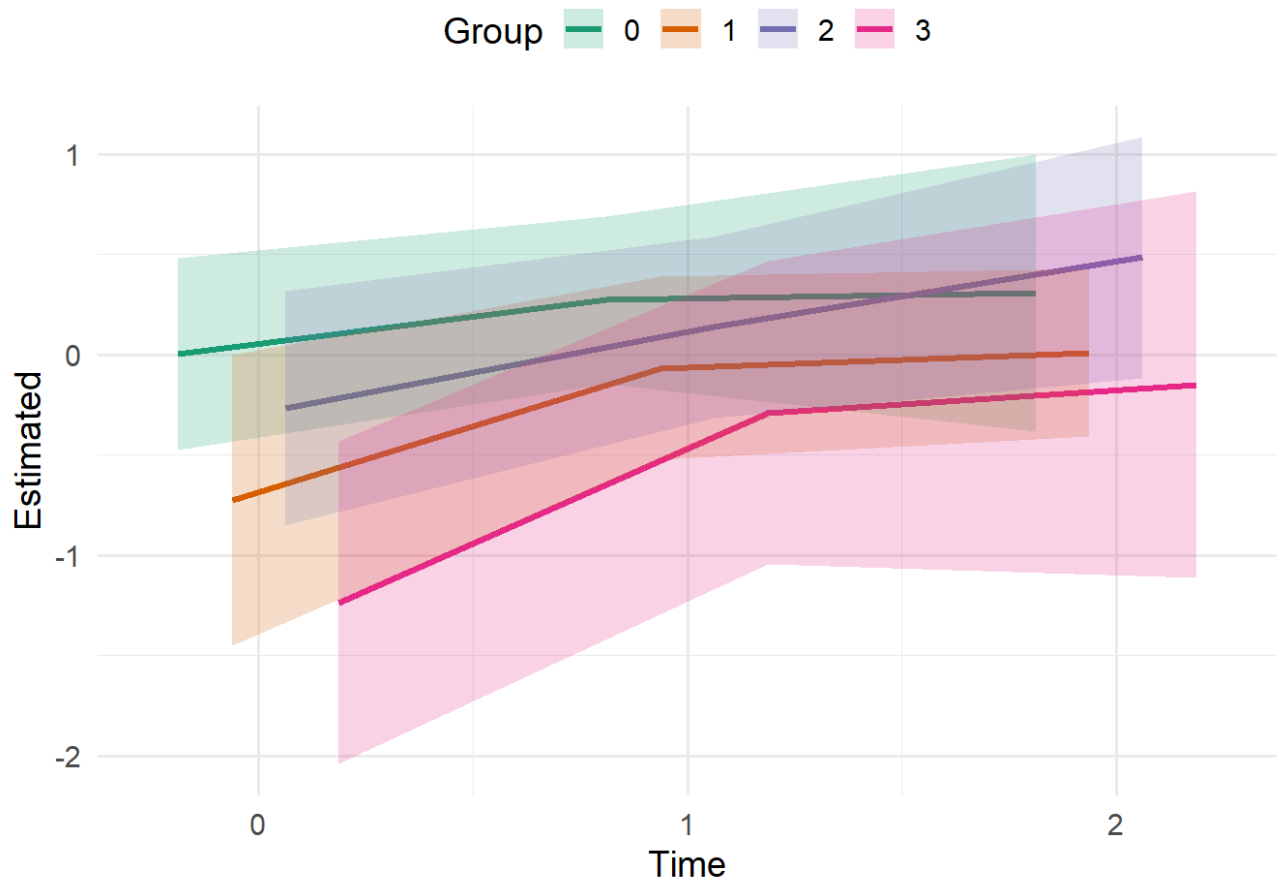
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of tnfa z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(tnfaModel_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```

15.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(tnfaModel_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0      -0.00626 0.223 Inf   -0.4440   0.43151
##   1      -0.73462 0.374 Inf   -1.4674  -0.00185
##   2      -0.27446 0.296 Inf   -0.8551   0.30621
##   3      -1.24593 0.386 Inf   -2.0029  -0.48892
##
## t = 1:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0       0.26323 0.181 Inf   -0.0924   0.61886
##   1      -0.07530 0.230 Inf   -0.5264   0.37582
##   2       0.12953 0.223 Inf   -0.3077   0.56676
##   3      -0.29779 0.357 Inf   -0.9985   0.40288
##
## t = 2:
##   GROUP    emmean      SE   df asymp.LCL asymp.UCL
##   0       0.29766 0.330 Inf   -0.3498   0.94514
##   1       0.00207 0.205 Inf   -0.3991   0.40324
##   2       0.47613 0.296 Inf   -0.1033   1.05555
##   3      -0.15778 0.465 Inf   -1.0701   0.75456
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(tnfaModel_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##   contrast      estimate      SE   df z.ratio p.value
##   GROUP0 - GROUP1    0.4542 0.313 Inf    1.452  0.4668
##   GROUP0 - GROUP2    0.0745 0.282 Inf    0.264  0.9936
##   GROUP0 - GROUP3    0.7520 0.378 Inf    1.988  0.1925
##   GROUP1 - GROUP2   -0.3797 0.306 Inf   -1.239  0.6018
##   GROUP1 - GROUP3    0.2979 0.446 Inf    0.667  0.9094
##   GROUP2 - GROUP3    0.6776 0.415 Inf    1.633  0.3600
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(tnfaModel_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1    0.728 0.364 Inf   2.002  0.1873
## GROUP0 - GROUP2    0.268 0.320 Inf   0.837  0.8369
## GROUP0 - GROUP3    1.240 0.427 Inf   2.903  0.0194
## GROUP1 - GROUP2   -0.460 0.375 Inf  -1.228  0.6091
## GROUP1 - GROUP3    0.511 0.505 Inf   1.013  0.7416
## GROUP2 - GROUP3    0.971 0.465 Inf   2.090  0.1562
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1    0.339 0.317 Inf   1.068  0.7089
## GROUP0 - GROUP2    0.134 0.282 Inf   0.475  0.9647
## GROUP0 - GROUP3    0.561 0.384 Inf   1.460  0.4621
## GROUP1 - GROUP2   -0.205 0.315 Inf  -0.650  0.9154
## GROUP1 - GROUP3    0.222 0.455 Inf   0.489  0.9616
## GROUP2 - GROUP3    0.427 0.421 Inf   1.014  0.7411
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1    0.296 0.382 Inf   0.773  0.8666
## GROUP0 - GROUP2   -0.178 0.342 Inf  -0.523  0.9537
## GROUP0 - GROUP3    0.455 0.425 Inf   1.071  0.7070
## GROUP1 - GROUP2   -0.474 0.347 Inf  -1.366  0.5211
## GROUP1 - GROUP3    0.160 0.505 Inf   0.316  0.9890
## GROUP2 - GROUP3    0.634 0.467 Inf   1.358  0.5257
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(tnfaModel_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.2695 0.213 Inf  -1.266  0.4143
## t0 - t2      -0.3039 0.403 Inf  -0.755  0.7307
## t1 - t2      -0.0344 0.276 Inf  -0.125  0.9915
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.6593 0.262 Inf  -2.516  0.0318
## t0 - t2      -0.7367 0.376 Inf  -1.960  0.1222
## t1 - t2      -0.0774 0.202 Inf  -0.384  0.9220
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.4040 0.230 Inf  -1.753  0.1857
## t0 - t2      -0.7506 0.384 Inf  -1.956  0.1234
## t1 - t2      -0.3466 0.239 Inf  -1.448  0.3162
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.9481 0.284 Inf  -3.335  0.0025
## t0 - t2      -1.0882 0.447 Inf  -2.435  0.0396
## t1 - t2      -0.1400 0.316 Inf  -0.443  0.8973
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

16 Model FABP3

```
bptest(lm(FABP3 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(FABP3 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id), data = db)
## BP = 30.974, df = 14, p-value = 0.005591
```

```
vars <- c("FABP3", "t", "GROUP", "Age", "Sex", "CIRS_severity_0", "id")
db_cc <- db[complete.cases(db[, vars]), vars]

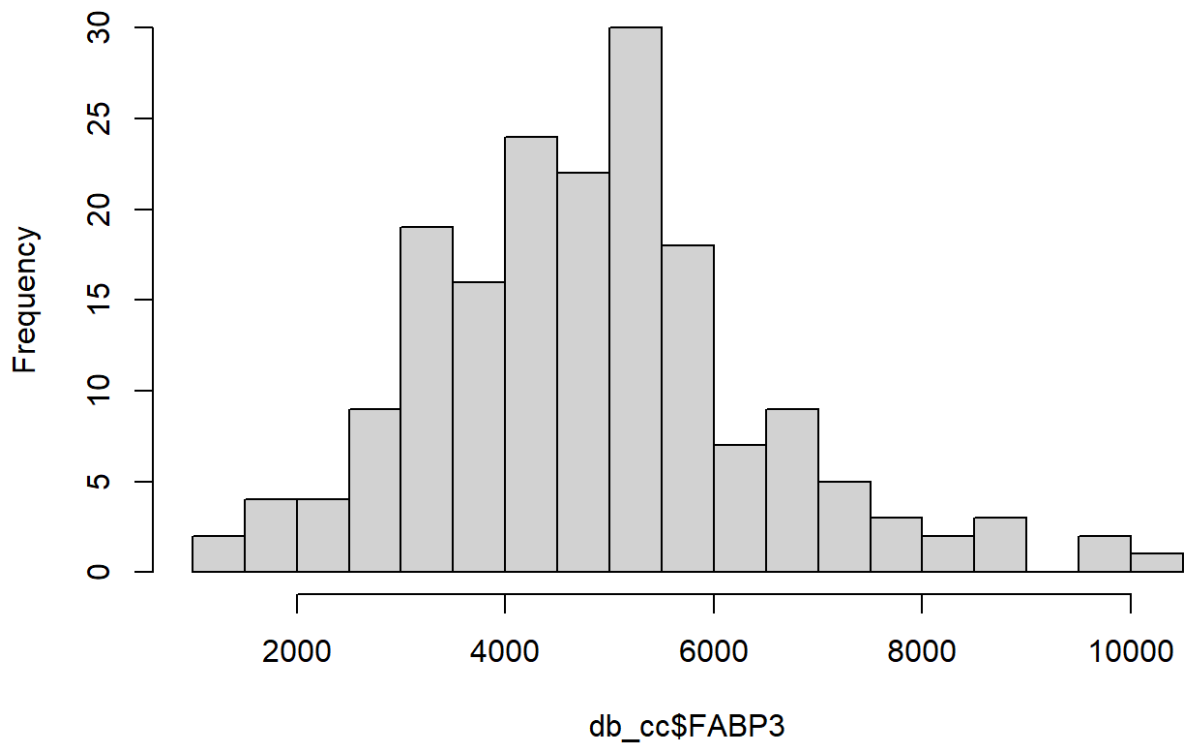
nrow(db); nrow(db_cc)
```

```
## [1] 243
```

```
## [1] 180
```

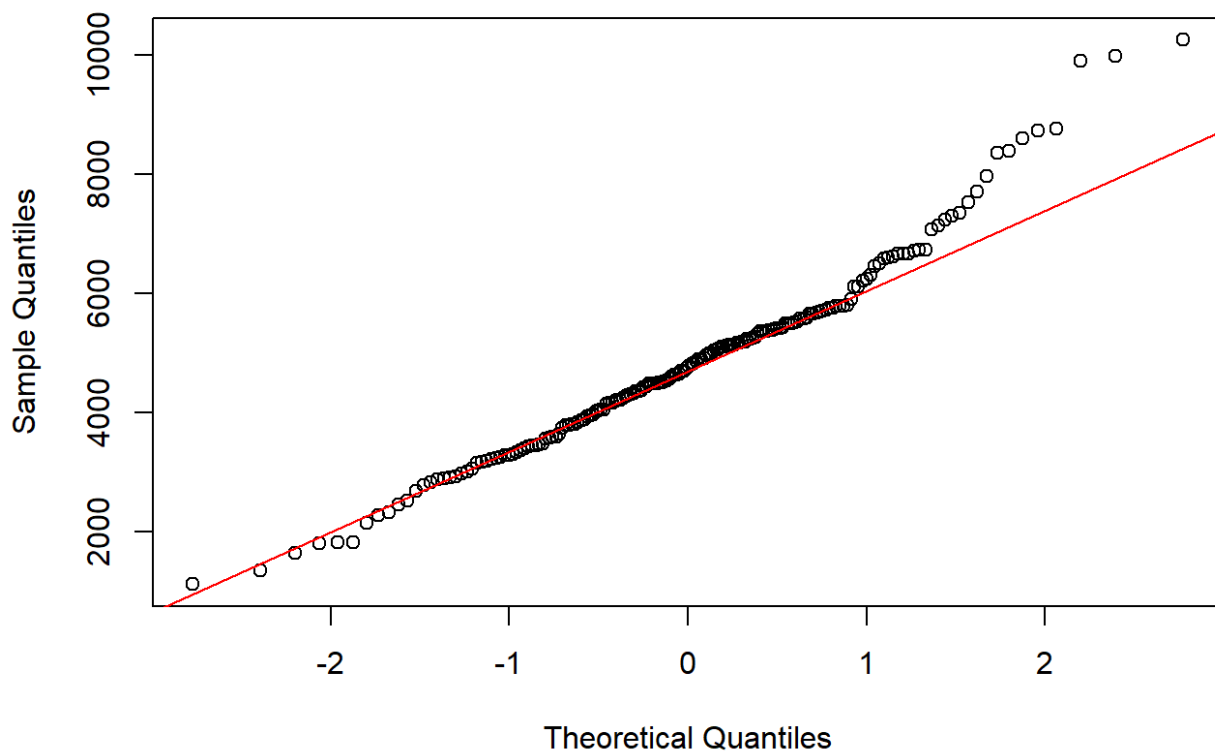
```
hist(db_cc$FABP3, breaks=30)
```

Histogram of db_cc\$FABP3



```
qqnorm(db_cc$FABP3); qqline(db_cc$FABP3, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$FABP3)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	1123	3786	4768	4836	5603	10247

```
min(db_cc$FABP3, na.rm=TRUE)
```

```
## [1] 1122.556
```

```
db$FABP3_z <- scale(log(db$FABP3))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$FABP3_z <- scale(log(db_cc$FABP3))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

FABP3Model_hetero <- glmmTMB(
  FABP3_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(FABP3Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## FABP3_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1
| id)
## Dispersion:                ~t
## Data: db_cc
##
##           AIC           BIC      logLik -2*log(L)  df.resid
##          438.8          499.5      -200.4      400.8       161
##
## Random effects:
##
## Conditional model:
##   Groups   Name              Variance Std.Dev.
##   id       (Intercept) 0.5976   0.7731
##   Residual                NA         NA
## Number of obs: 180, groups: id, 79
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -7.92641    3.05325  -2.596  0.00943 **
## t1              -0.41943    0.21554  -1.946  0.05166 .
## t2              -0.35662    0.41855  -0.852  0.39420
## GROUP1          0.74529    0.38485   1.937  0.05280 .
## GROUP2          0.91606    0.32394   2.828  0.00469 **
## GROUP3          0.02523    0.42456   0.059  0.95262
## Age             0.09806    0.03968   2.471  0.01346 *
## SexM            0.06784    0.19525   0.347  0.72827
## CIRS_severity_0 0.05678    0.14928   0.380  0.70368
## t1:GROUP1       0.09119    0.31326   0.291  0.77098
## t2:GROUP1      -0.24575    0.40992  -0.600  0.54884
## t1:GROUP2       0.01238    0.24341   0.051  0.95943
## t2:GROUP2      -0.39399    0.33707  -1.169  0.24245
## t1:GROUP3       0.31982    0.32499   0.984  0.32507
## t2:GROUP3      -0.04079    0.39879  -0.102  0.91853
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.46103    0.12560  -3.671 0.000242 ***
## t1           -0.74857    0.34403  -2.176 0.029563 *
## t2           -0.08894    0.16360  -0.544 0.586694
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```



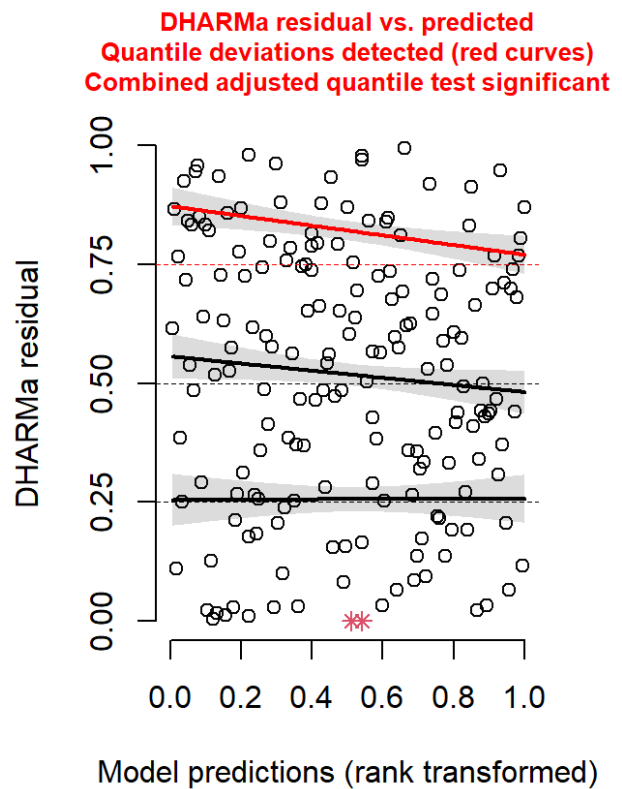
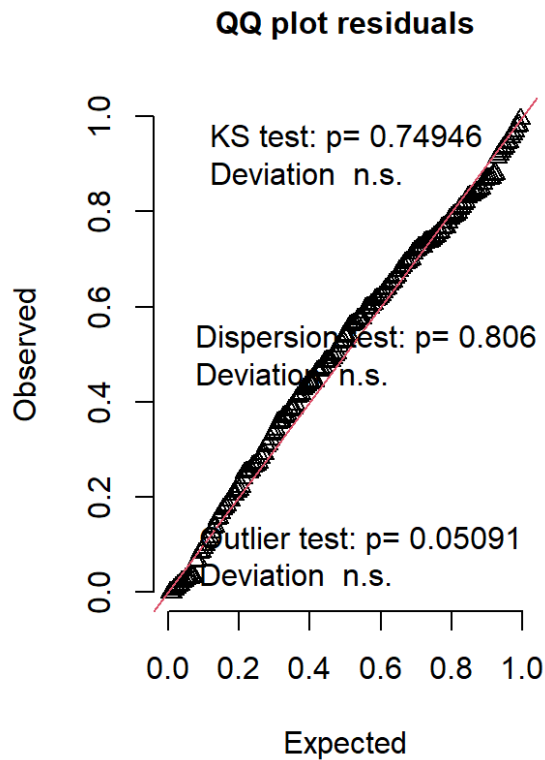
```
FABP3Model_homo <- glmmTMB(
  FABP3_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(FABP3Model_homo, FABP3Model_hetero, test="Chisq")
```

```
## Data: db_cc
## Models:
## FABP3Model_homo: FABP3_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## FABP3Model_homo:      (1 | id), zi=~0, disp=~t
## FABP3Model_hetero: FABP3_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_se
verity_0 + , zi=~0, disp=~1
## FABP3Model_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Ch
isq)
## FABP3Model_homo    17 443.41 497.70 -204.71   409.41
## FABP3Model_hetero  19 438.79 499.46 -200.40   400.79 8.6222      2    0.0
1342 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

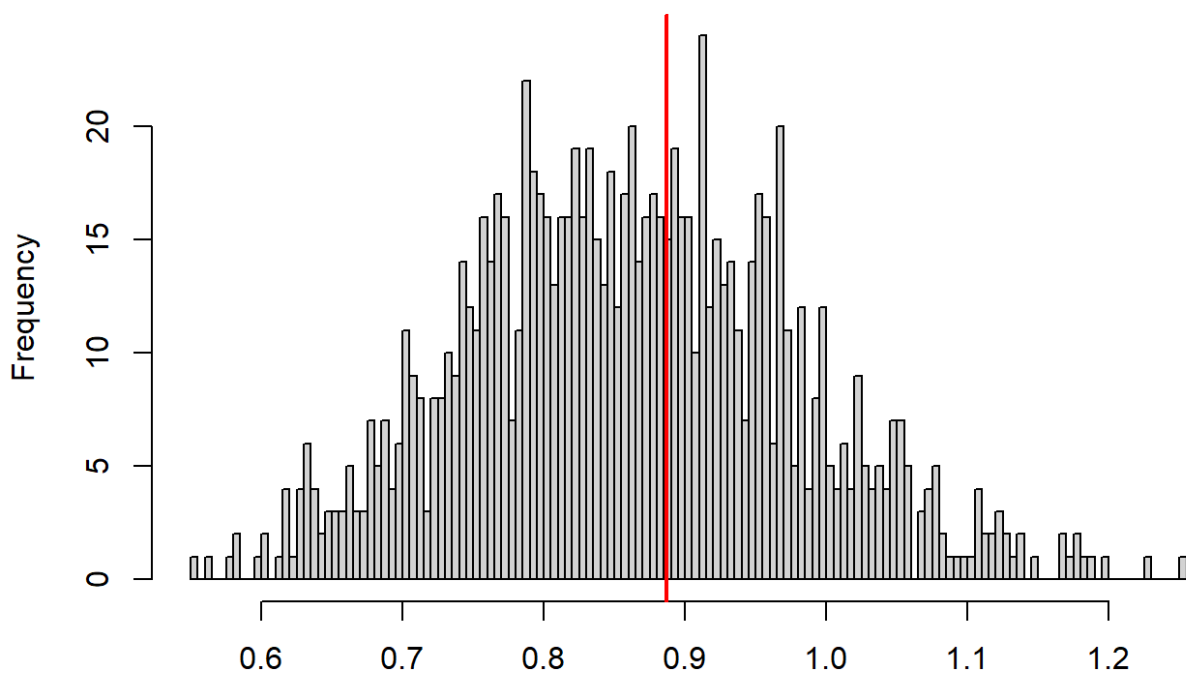
```
sim_res <- simulateResiduals(FABP3Model_hetero, n = 1000)
plot(sim_res)
```

DHARMA residual



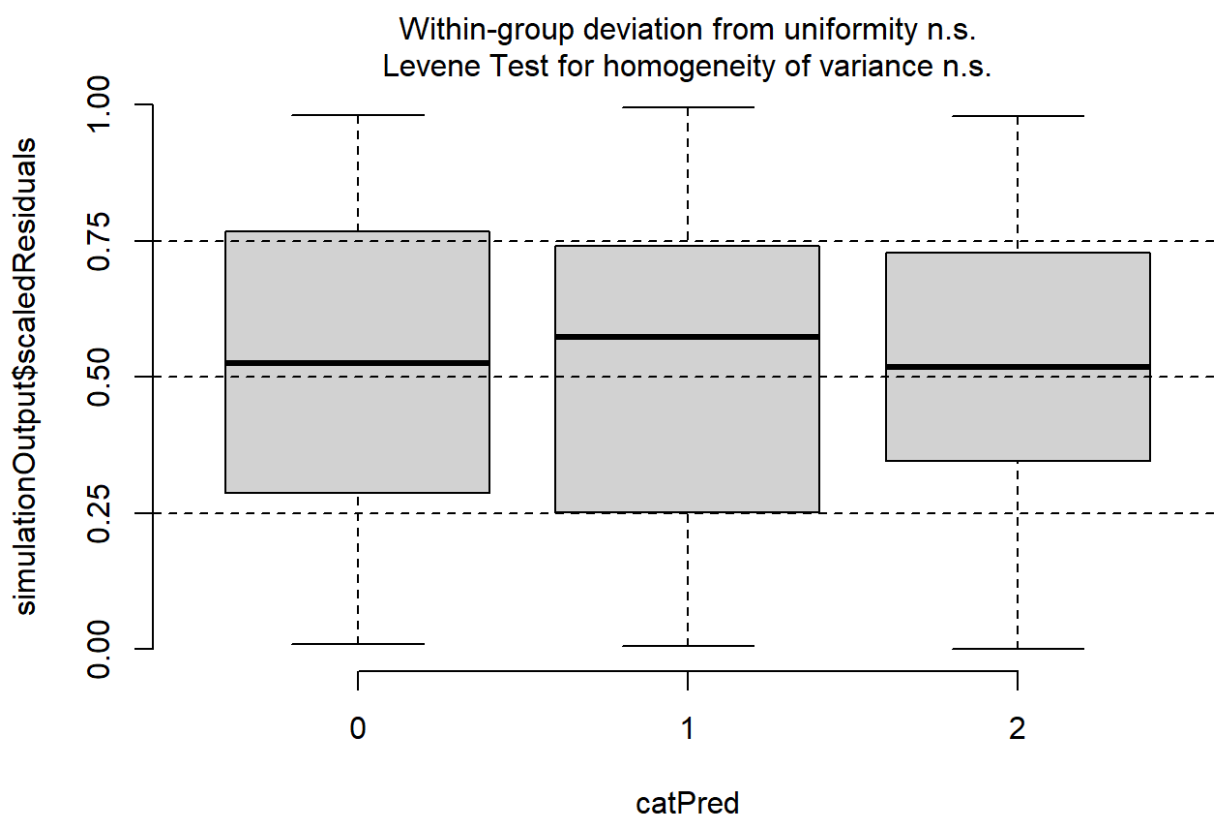
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data:  simulationOutput
## dispersion = 1.0314, p-value = 0.806
## alternative hypothesis: two.sided
```

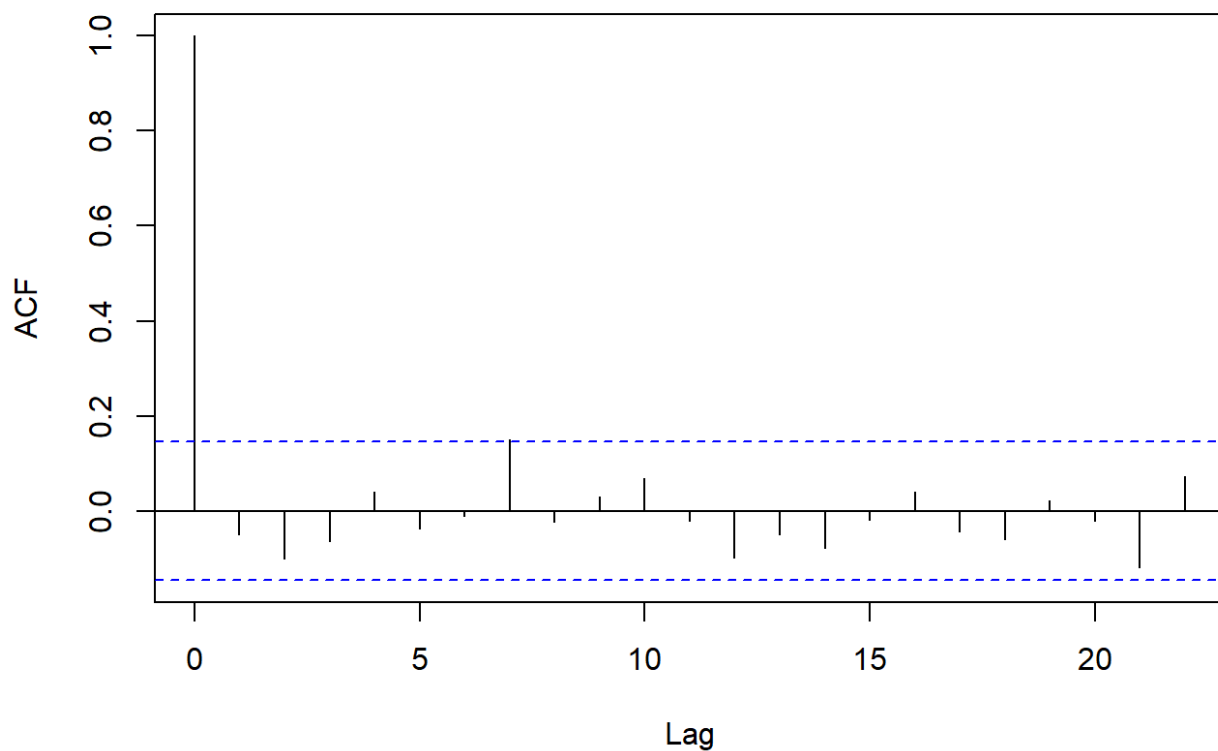
```
testQuantiles(sim_res, db_cc$t)
```



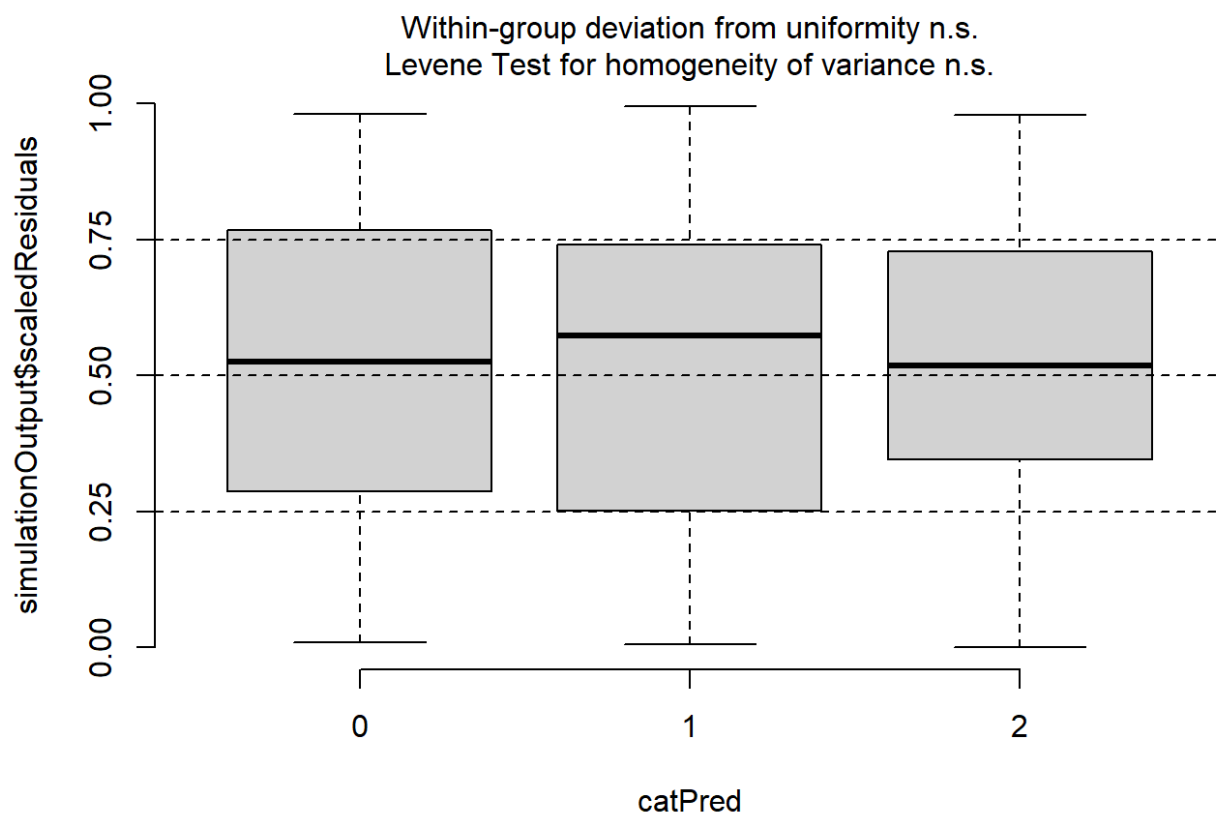
```
## NULL
```

```
acf(residuals(FABP3Model_hetero))
```

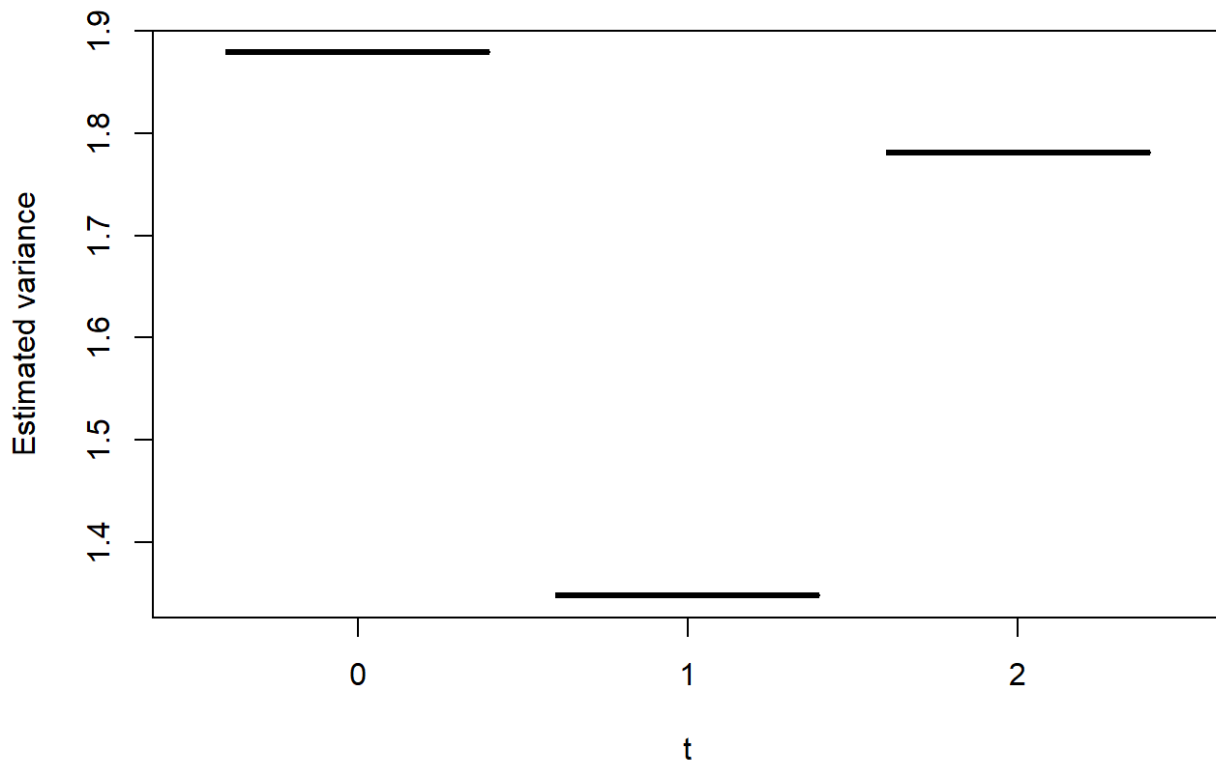
Series residuals(FABP3Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

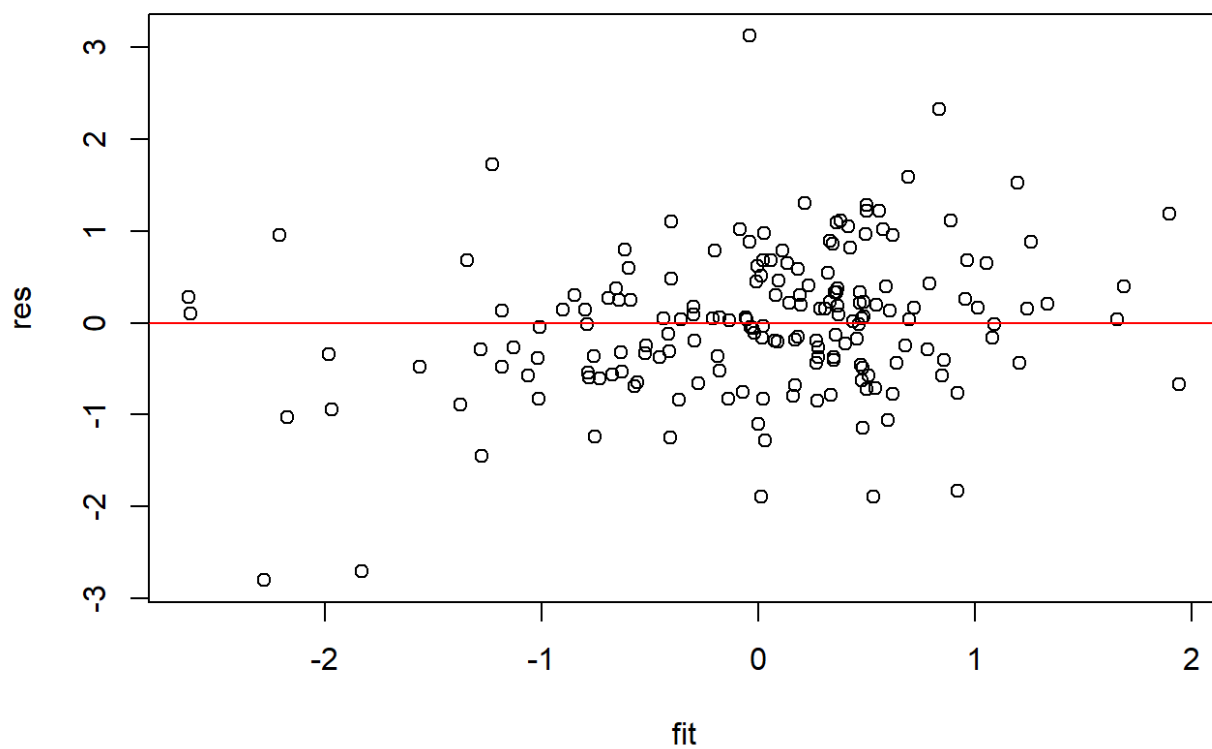


```
pred_disp <- predict(FABP3Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(FABP3Model_hetero, type="pearson")
fit <- fitted(FABP3Model_hetero)

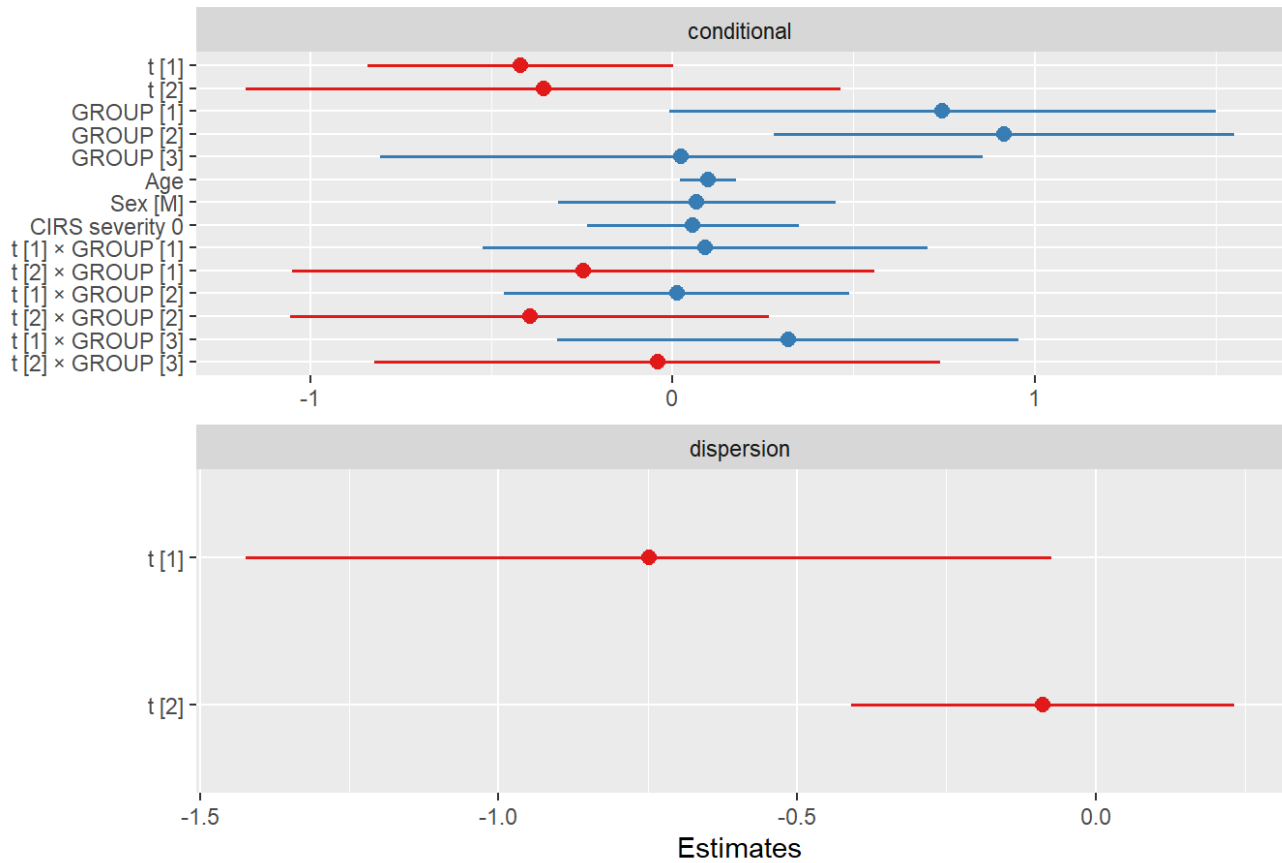
plot(fit, res)
abline(h=0, col="red")
```



16.1 Visualizzazione del modello

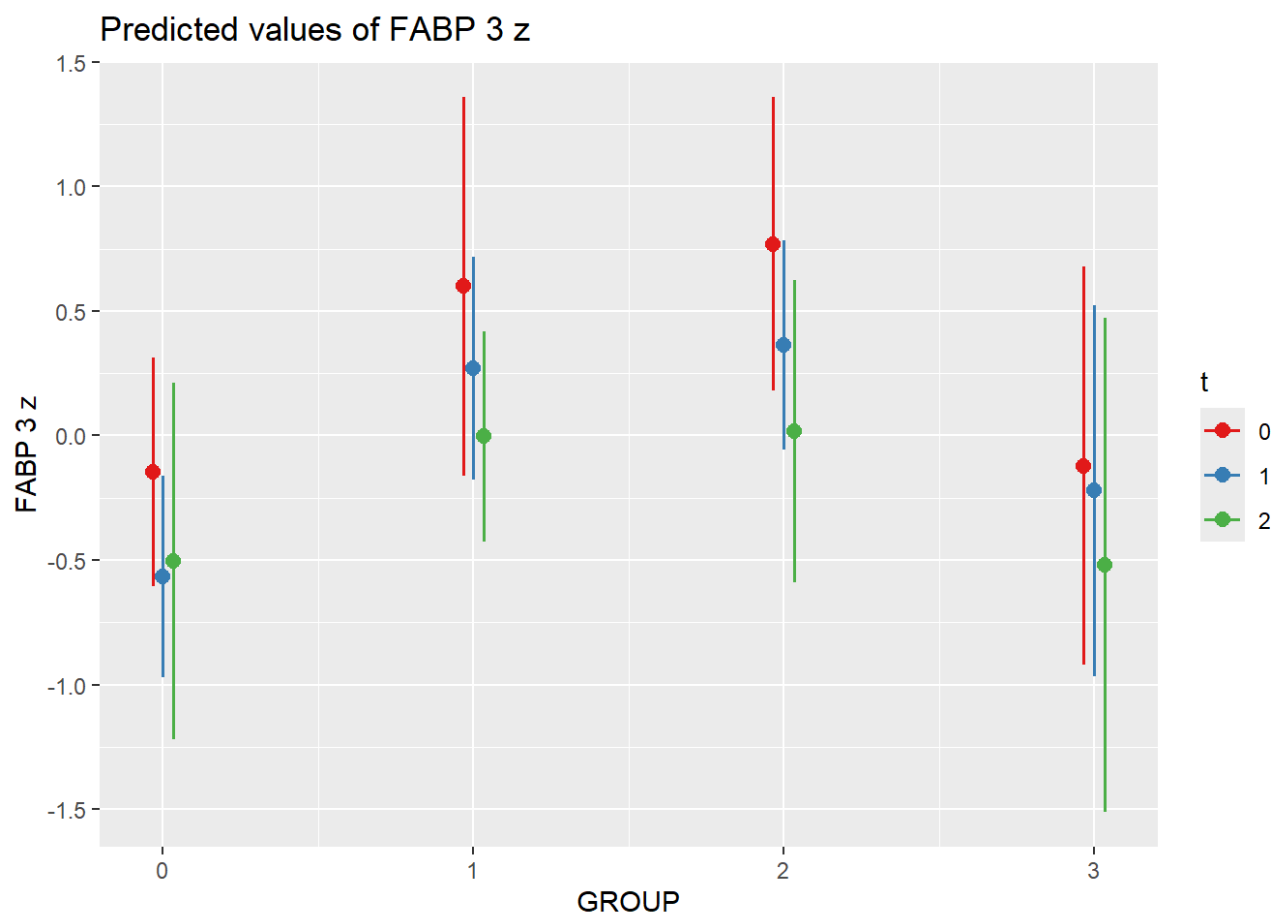
```
plot_model(FABP3Model_hetero)
```

FABP 3 z



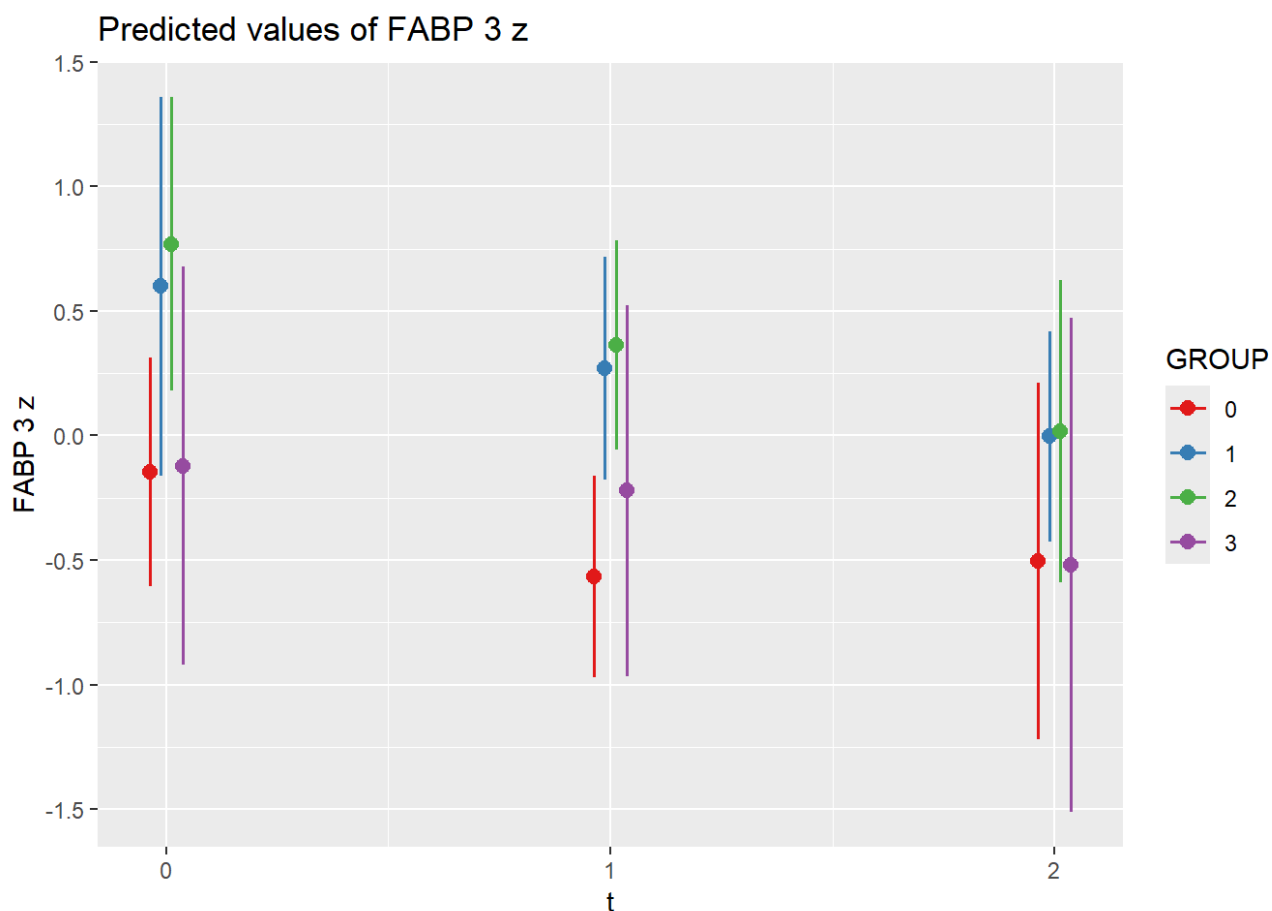
```
plot_model(FABP3Model_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```



```
plot_model(FABP3Model_hetero, type = "pred",  
            terms = c("t", "GROUP"),  
            pred.type = "fe")
```

```
## Ignoring unknown labels:  
## • linetype : "GROUP"  
## • shape : "GROUP"
```

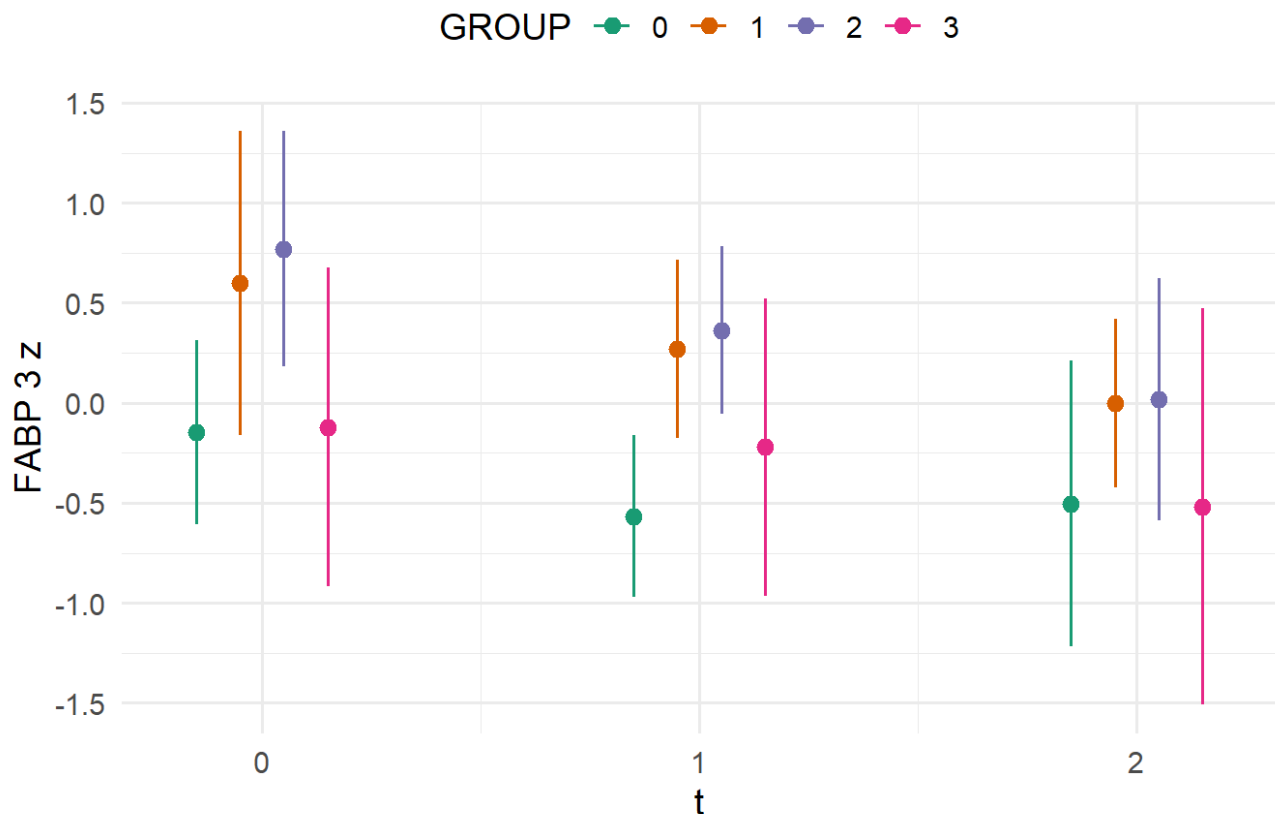



```
plot_model(FABP3Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

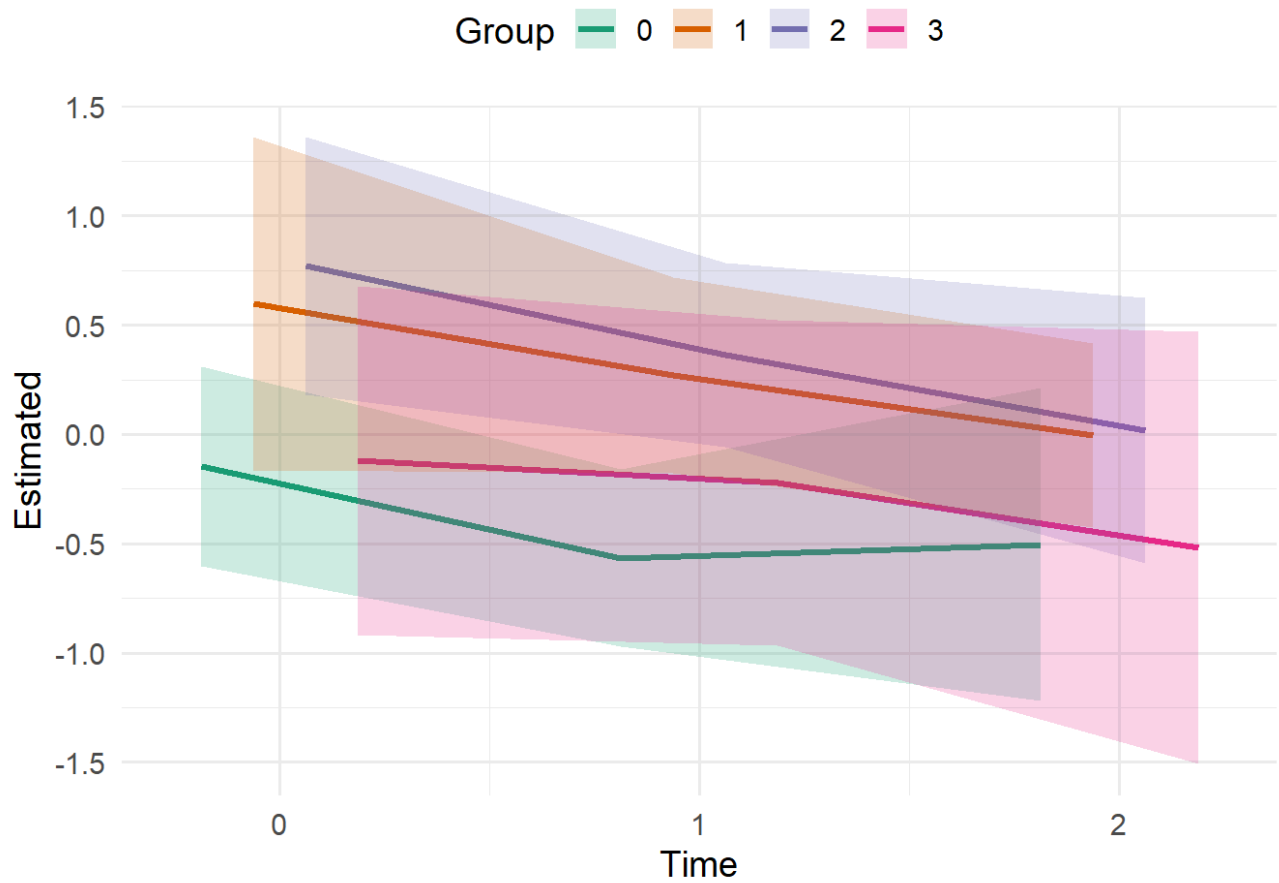
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of FABP 3 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(FABP3Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



16.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(FABP3Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.1125 0.213 Inf  -0.53083    0.306
##  1       0.6328 0.390 Inf  -0.13125    1.397
##  2       0.8036 0.299 Inf   0.21823    1.389
##  3      -0.0873 0.386 Inf  -0.84307    0.669
##
## t = 1:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.5319 0.177 Inf  -0.87872   -0.185
##  1       0.3045 0.223 Inf  -0.13292    0.742
##  2       0.3965 0.206 Inf  -0.00703    0.800
##  3      -0.1869 0.354 Inf  -0.88032    0.507
##
## t = 2:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.4691 0.346 Inf  -1.14818    0.210
##  1       0.0304 0.206 Inf  -0.37417    0.435
##  2       0.0529 0.301 Inf  -0.53635    0.642
##  3      -0.4847 0.485 Inf  -1.43569    0.466
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(FABP3Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.6938 0.310 Inf  -2.237  0.1134
## GROUP0 - GROUP2 -0.7889 0.269 Inf  -2.929  0.0179
## GROUP0 - GROUP3 -0.1182 0.362 Inf  -0.327  0.9880
## GROUP1 - GROUP2 -0.0951 0.300 Inf  -0.317  0.9890
## GROUP1 - GROUP3  0.5755 0.437 Inf   1.317  0.5522
## GROUP2 - GROUP3  0.6706 0.398 Inf   1.684  0.3321
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(FABP3Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.7453 0.385 Inf  -1.937  0.2127
## GROUP0 - GROUP2 -0.9161 0.324 Inf  -2.828  0.0242
## GROUP0 - GROUP3 -0.0252 0.425 Inf  -0.059  0.9999
## GROUP1 - GROUP2 -0.1708 0.404 Inf  -0.422  0.9747
## GROUP1 - GROUP3  0.7201 0.520 Inf   1.384  0.5094
## GROUP2 - GROUP3  0.8908 0.469 Inf   1.898  0.2288
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.8365 0.313 Inf  -2.675  0.0375
## GROUP0 - GROUP2 -0.9284 0.263 Inf  -3.530  0.0024
## GROUP0 - GROUP3 -0.3450 0.377 Inf  -0.915  0.7966
## GROUP1 - GROUP2 -0.0920 0.302 Inf  -0.304  0.9902
## GROUP1 - GROUP3  0.4914 0.451 Inf   1.089  0.6963
## GROUP2 - GROUP3  0.5834 0.408 Inf   1.429  0.4810
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.4995 0.402 Inf  -1.244  0.5987
## GROUP0 - GROUP2 -0.5221 0.354 Inf  -1.475  0.4526
## GROUP0 - GROUP3  0.0156 0.441 Inf   0.035  1.0000
## GROUP1 - GROUP2 -0.0225 0.355 Inf  -0.063  0.9999
## GROUP1 - GROUP3  0.5151 0.526 Inf   0.978  0.7618
## GROUP2 - GROUP3  0.5376 0.482 Inf   1.115  0.6802
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(FABP3Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      0.4194 0.216 Inf   1.946  0.1259
## t0 - t2      0.3566 0.419 Inf   0.852  0.6706
## t1 - t2     -0.0628 0.303 Inf  -0.207  0.9767
##
## GROUP = 1:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      0.3282 0.298 Inf   1.100  0.5138
## t0 - t2      0.6024 0.402 Inf   1.497  0.2922
## t1 - t2      0.2741 0.211 Inf   1.301  0.3945
##
## GROUP = 2:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      0.4071 0.252 Inf   1.614  0.2395
## t0 - t2      0.7506 0.403 Inf   1.861  0.1502
## t1 - t2      0.3436 0.252 Inf   1.363  0.3606
##
## GROUP = 3:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      0.0996 0.333 Inf   0.299  0.9520
## t0 - t2      0.3974 0.500 Inf   0.795  0.7063
## t1 - t2      0.2978 0.364 Inf   0.819  0.6911
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

17 Model Ferritin

```
bptest(lm(Ferritin ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1
| id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(Ferritin ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0
+      (1 | id), data = db)
## BP = 10.096, df = 14, p-value = 0.7552
```

```
vars <- c("Ferritin","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

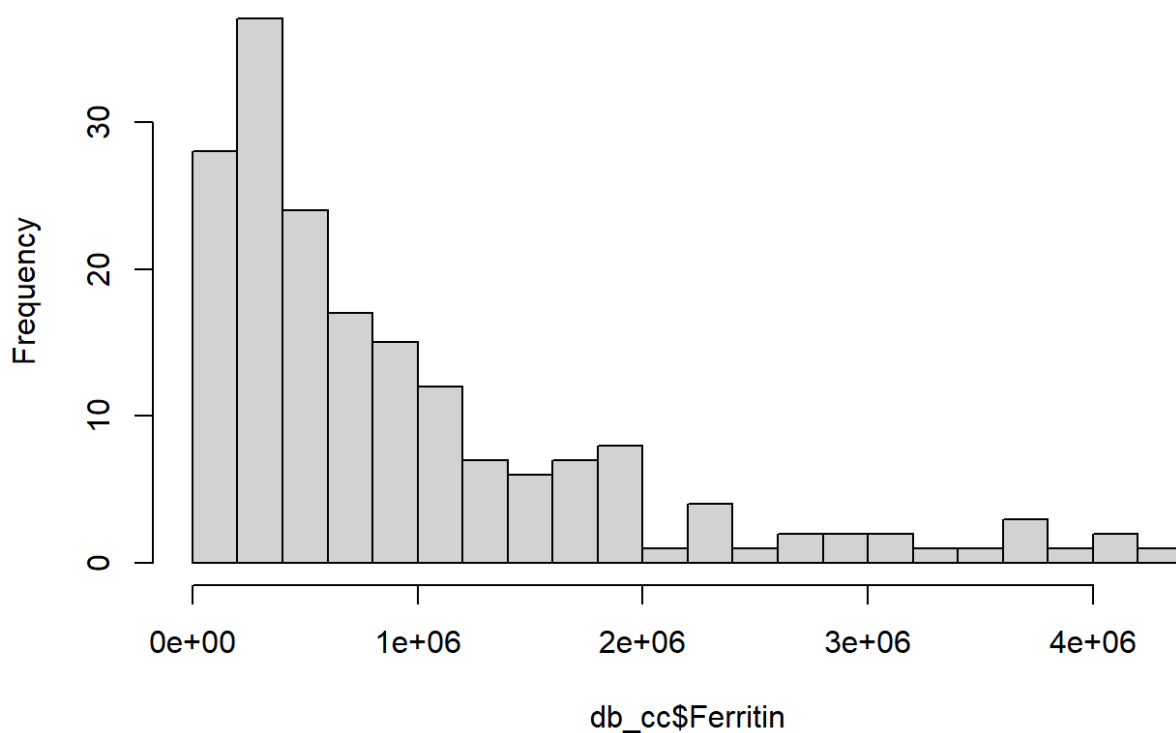
nrow(db); nrow(db_cc)
```

```
## [1] 243
```

```
## [1] 182
```

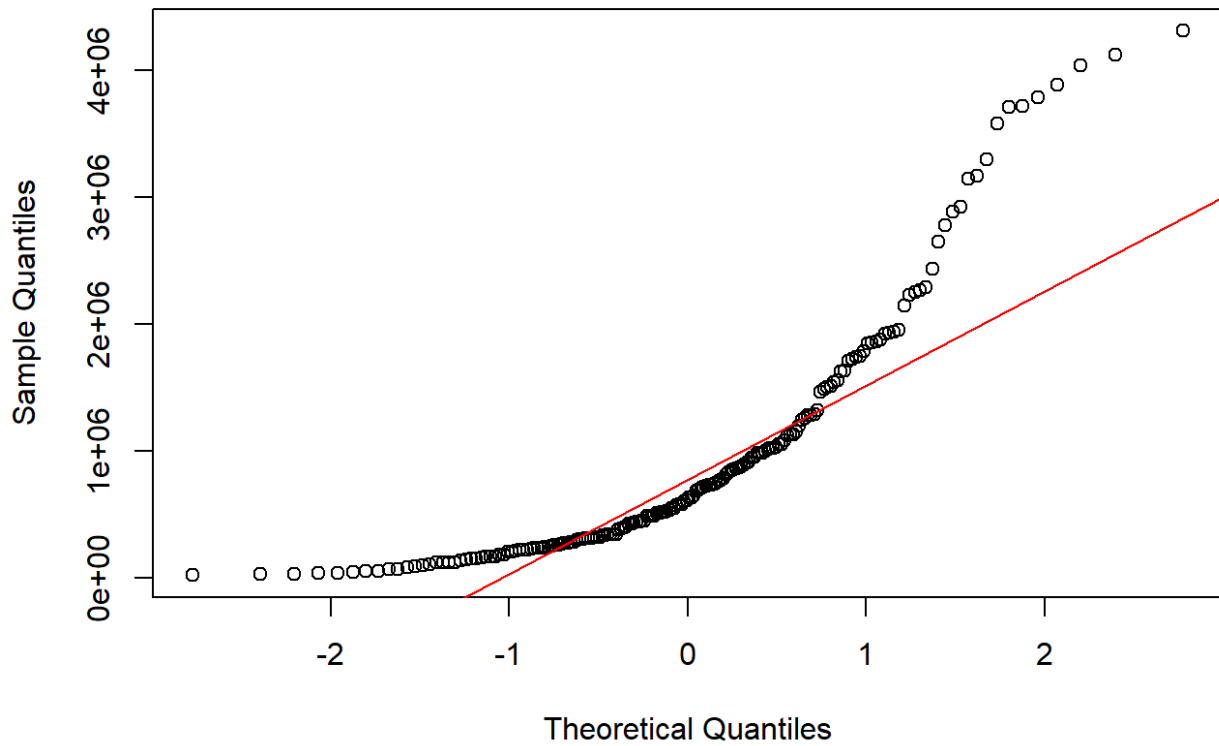
```
hist(db_cc$Ferritin, breaks=30)
```

Histogram of db_cc\$Ferritin



```
qqnorm(db_cc$Ferritin); qqline(db_cc$Ferritin, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$Ferritin)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##  25133  278635  626645  960365 1279607 4309418
```

```
min(db_cc$Ferritin, na.rm=TRUE)
```

```
## [1] 25132.47
```



```
db$Ferritin_z <- scale(log(db$Ferritin))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$Ferritin_z <- scale(log(db_cc$Ferritin))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

FerritinModel_hetero <- glmmTMB(
  Ferritin_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(FerritinModel_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## Ferritin_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id)
## Dispersion: ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    427.6    488.5    -194.8    389.6      163
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id       (Intercept) 0.6475   0.8047
##   Residual              NA       NA
## Number of obs: 182, groups: id, 79
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    7.574581   3.099323   2.444   0.0145 *
## t1              0.366266   0.203826   1.797   0.0723 .
## t2              0.916886   0.403603   2.272   0.0231 *
## GROUP1          0.093536   0.334719   0.279   0.7799
## GROUP2          0.227149   0.299128   0.759   0.4476
## GROUP3          0.362388   0.398785   0.909   0.3635
## Age            -0.101602   0.040214  -2.527   0.0115 *
## SexM            0.437159   0.204579   2.137   0.0326 *
## CIRS_severity_0 -0.002284   0.141171  -0.016   0.9871
## t1:GROUP1       -0.681327   0.251750  -2.706   0.0068 **
## t2:GROUP1       -0.638373   0.347994  -1.834   0.0666 .
## t1:GROUP2       -0.446713   0.198827  -2.247   0.0247 *
## t2:GROUP2       -0.687153   0.293495  -2.341   0.0192 *
## t1:GROUP3       -0.627697   0.263946  -2.378   0.0174 *
## t2:GROUP3       -0.816573   0.345562  -2.363   0.0181 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.7475     0.1473  -5.073 3.91e-07 ***
## t1            -0.3340     0.2912  -1.147   0.251
## t2             0.1553     0.1918   0.810   0.418
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

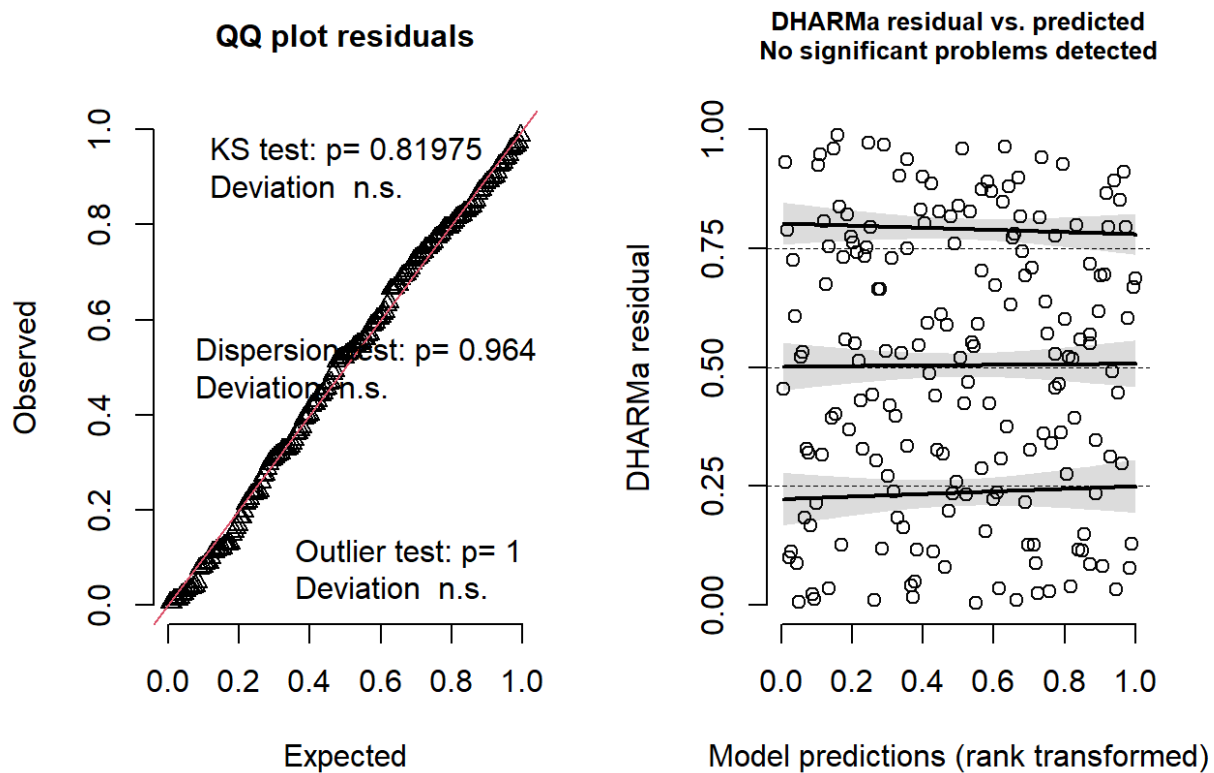
```
FerritinModel_homo <- glmmTMB(
  Ferritin_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | i
d),
  family = gaussian(),
  data = db_cc
)

anova(FerritinModel_homo, FerritinModel_hetero, test="Chisq")
```

```
## Data: db_cc
## Models:
## FerritinModel_homo: Ferritin_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## FerritinModel_homo:      (1 | id), zi=~0, disp=~t
## FerritinModel_hetero: Ferritin_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 + , zi=~0, disp=~1
## FerritinModel_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr
(>Chisq)
## FerritinModel_homo    17 427.94 482.41 -196.97   393.94
## FerritinModel_hetero  19 427.59 488.47 -194.80   389.59 4.3479      2
0.1137
```

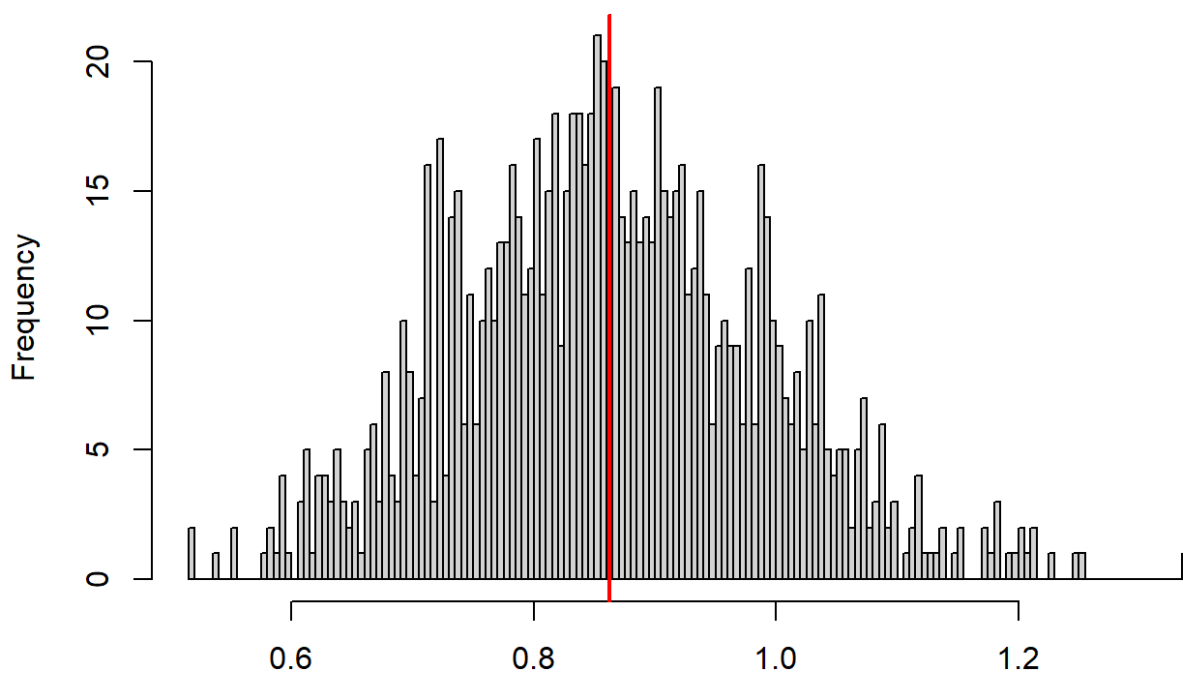
```
sim_res <- simulateResiduals(FerritinModel_hetero, n = 1000)
plot(sim_res)
```

DHARMA residual



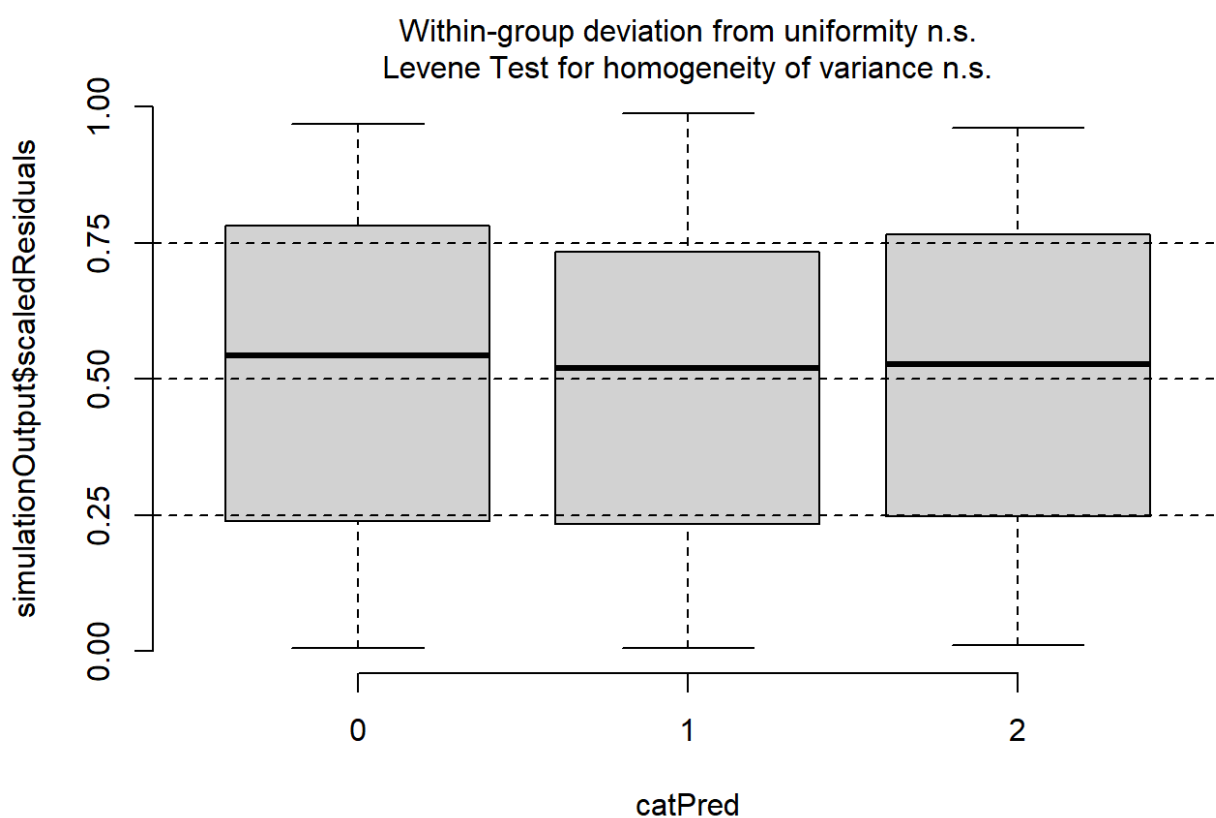
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 0.99841, p-value = 0.964
## alternative hypothesis: two.sided
```

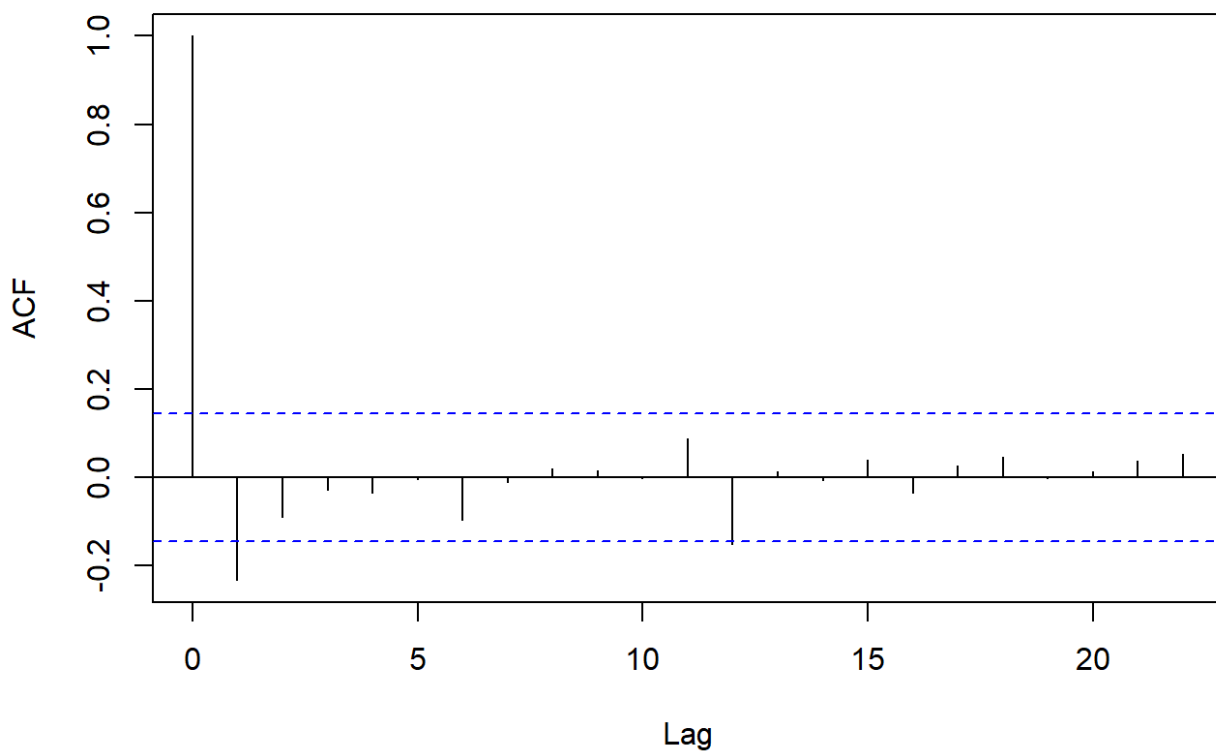
```
testQuantiles(sim_res, db_cc$t)
```



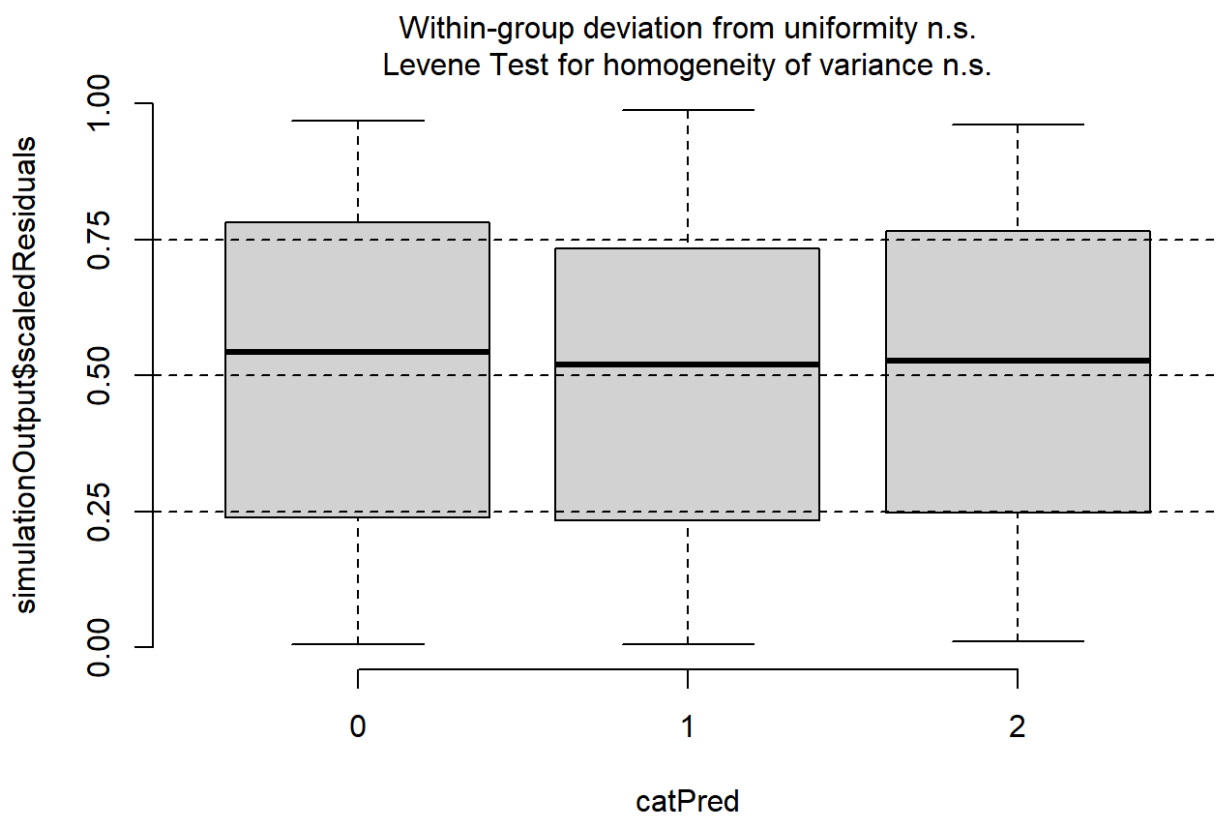
```
## NULL
```

```
acf(residuals(FerritinModel_hetero))
```

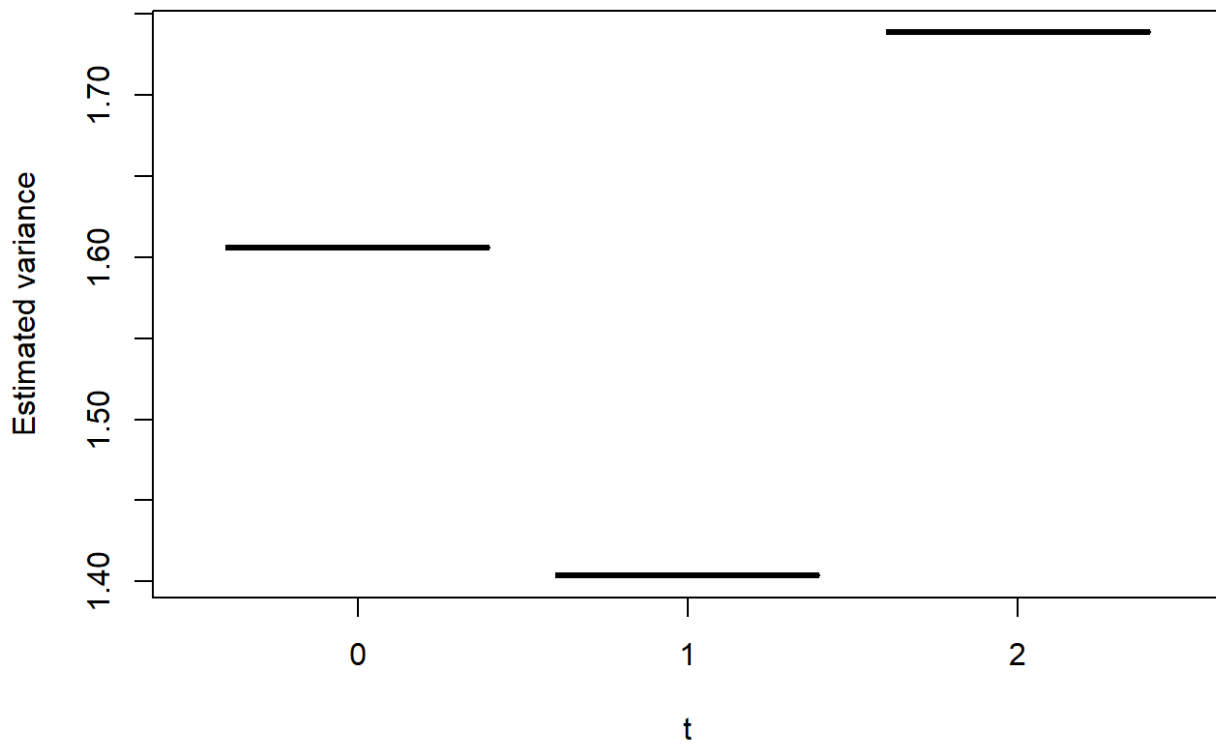
Series residuals(FerritinModel_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

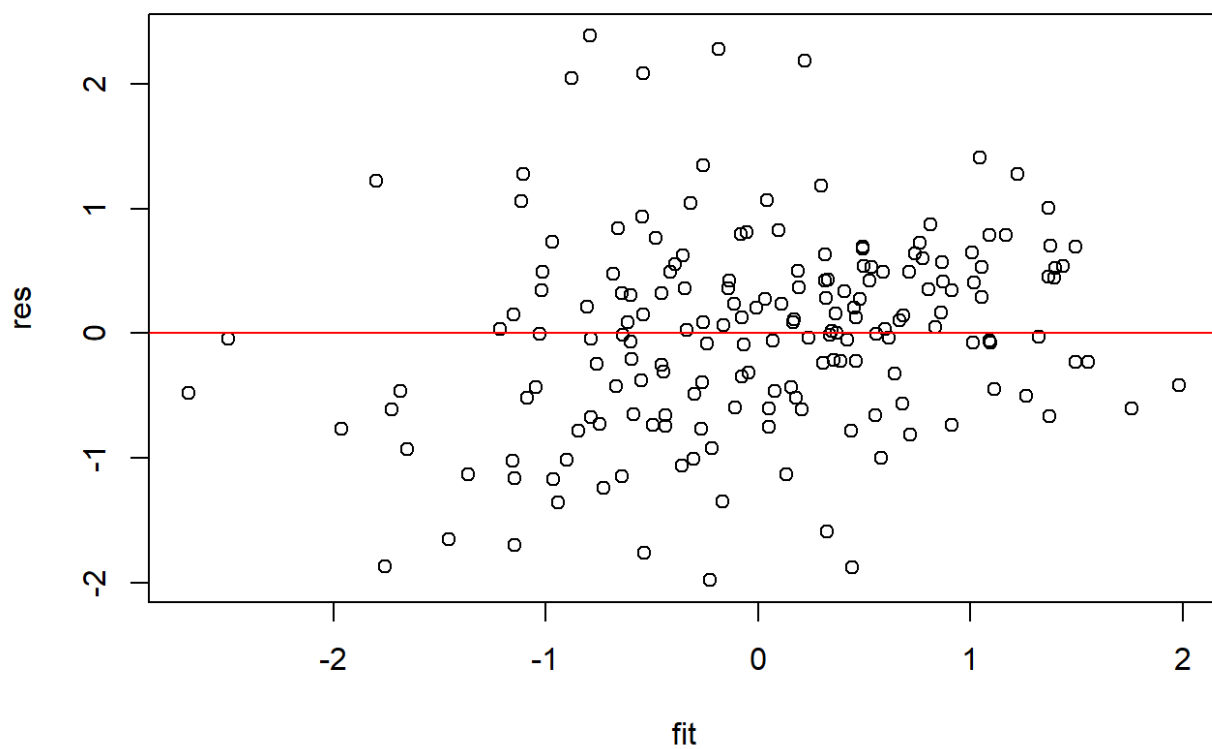


```
pred_disp <- predict(FerritinModel_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(FerritinModel_hetero, type="pearson")
fit <- fitted(FerritinModel_hetero)

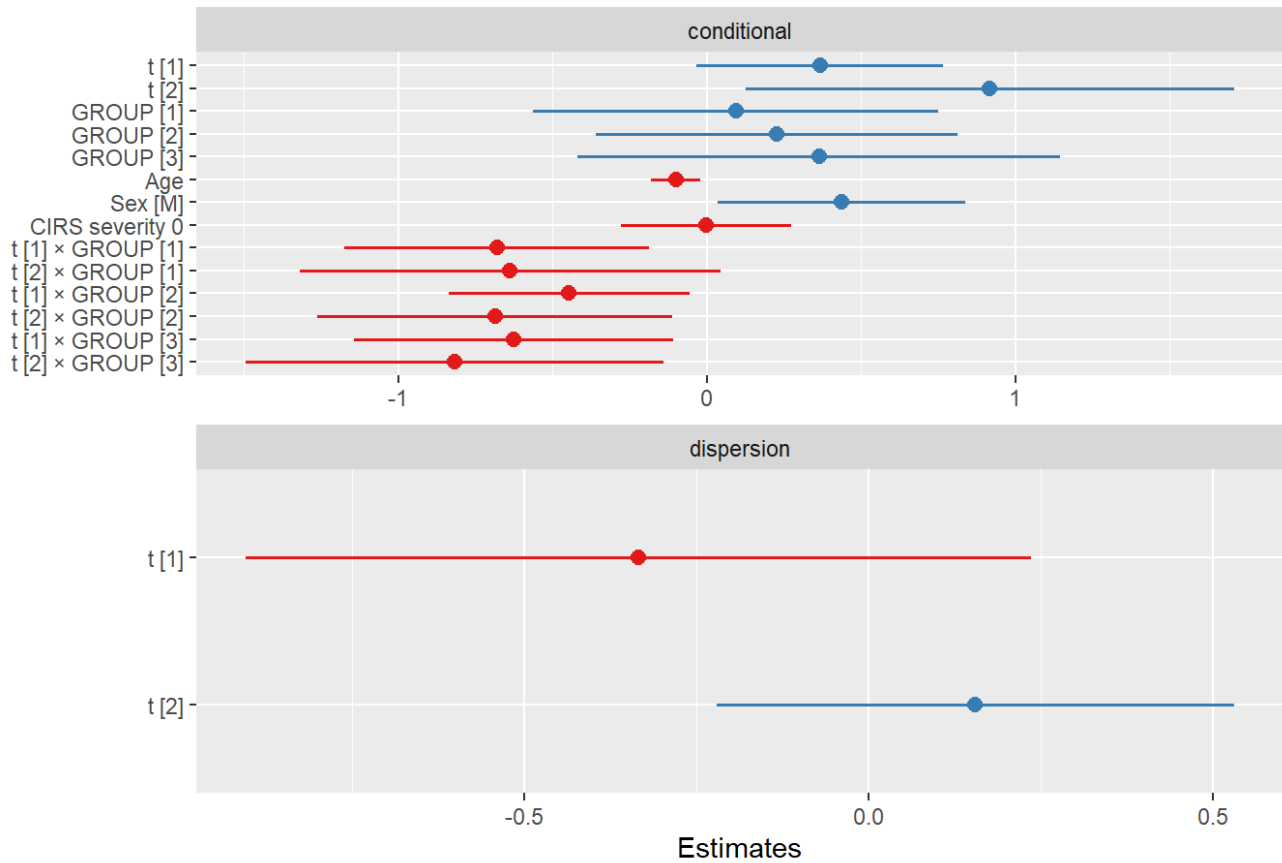
plot(fit, res)
abline(h=0, col="red")
```



17.1 Visualizzazione del modello

```
plot_model(FerritinModel_hetero)
```

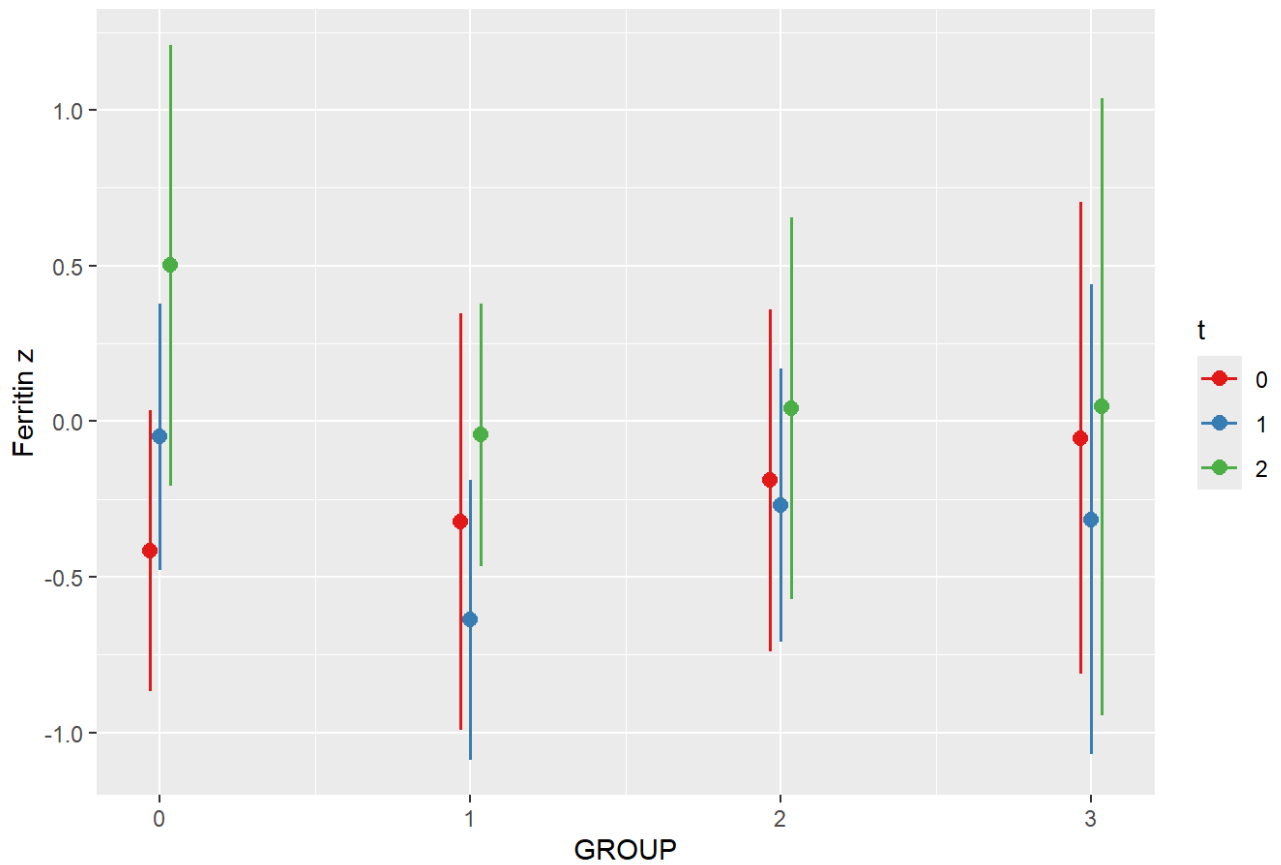

Ferritin z



```
plot_model(FerritinModel_hetero, type = "pred",
  terms = c("GROUP", "t"),
  pred.type = "fe")
```

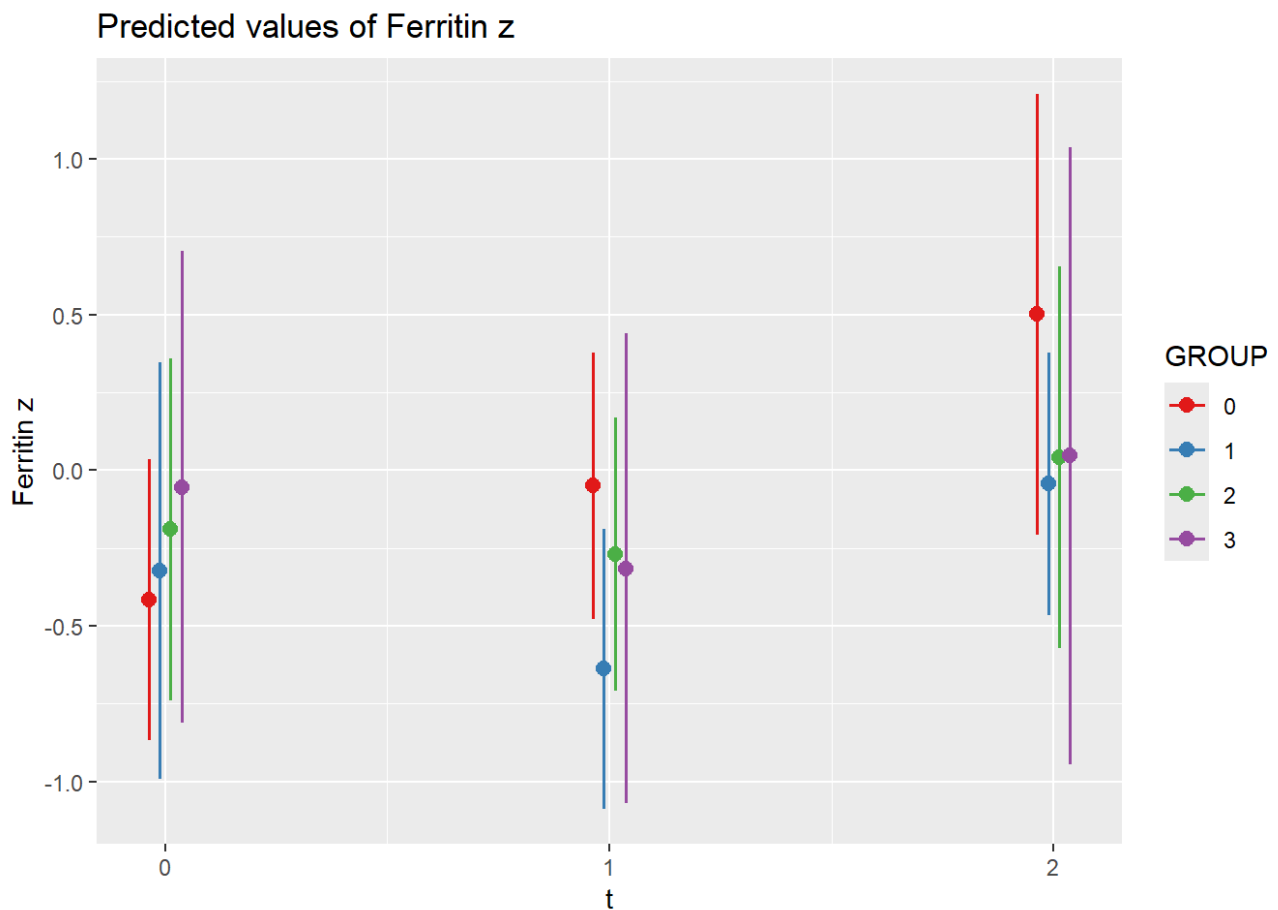
```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

Predicted values of Ferritin z



```
plot_model(FerritinModel_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

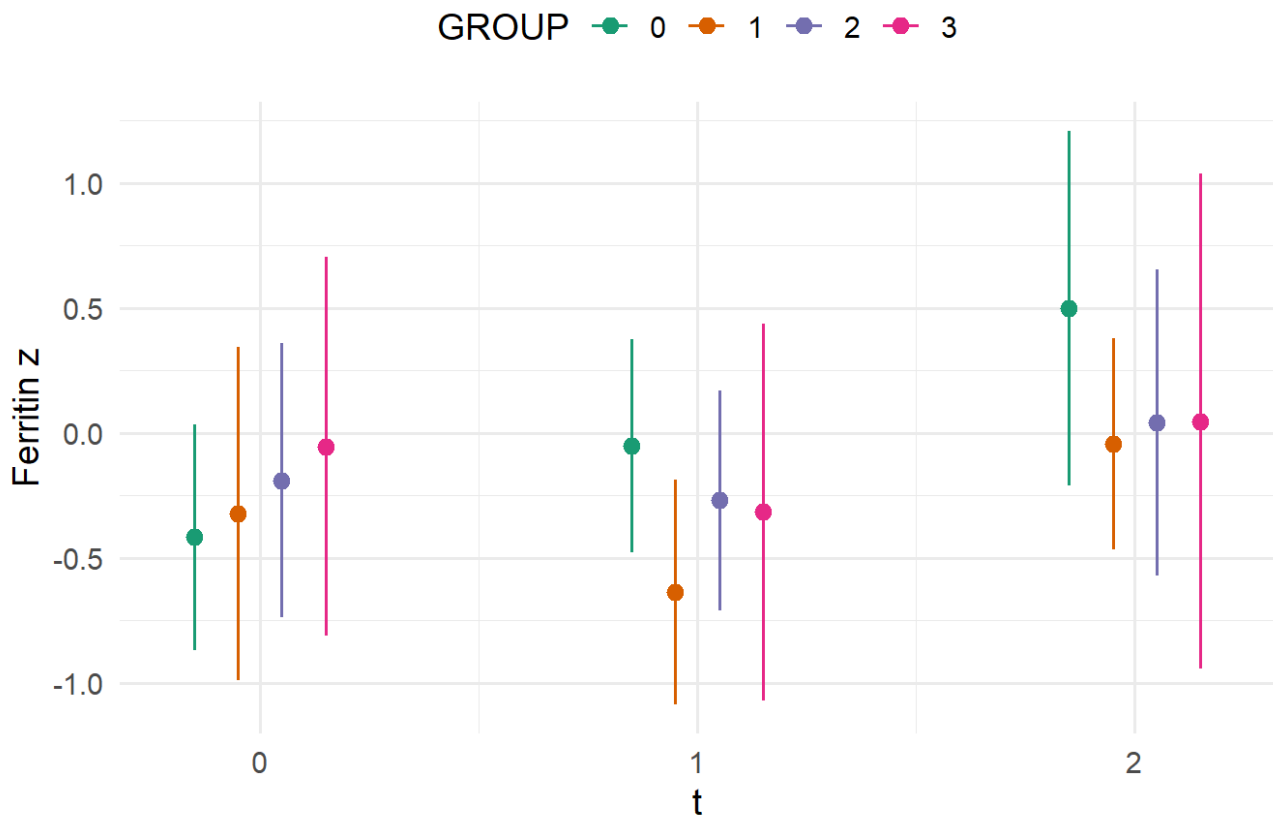


```
plot_model(FerritinModel_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

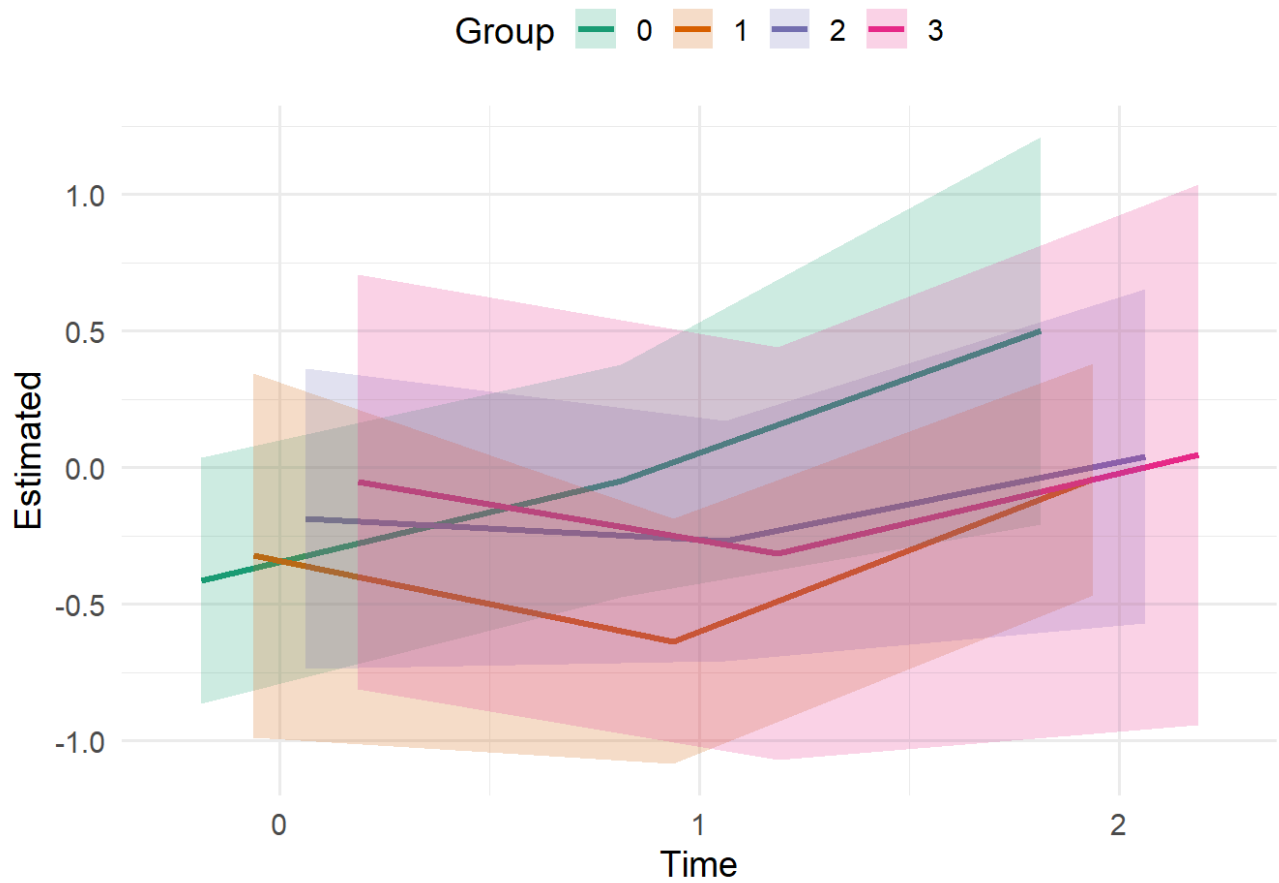
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of Ferritin z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(FerritinModel_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



17.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(FerritinModel_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.1974 0.206 Inf   -0.6020    0.2073
##  1      -0.1038 0.344 Inf   -0.7787    0.5710
##  2       0.0298 0.279 Inf   -0.5167    0.5763
##  3       0.1650 0.362 Inf   -0.5452    0.8752
##
## t = 1:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.1689 0.185 Inf   -0.1927    0.5305
##  1      -0.4189 0.229 Inf   -0.8669    0.0291
##  2      -0.0507 0.215 Inf   -0.4720    0.3707
##  3      -0.0964 0.357 Inf   -0.7967    0.6039
##
## t = 2:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.7195 0.340 Inf    0.0524    1.3866
##  1       0.1747 0.207 Inf   -0.2314    0.5807
##  2       0.2595 0.302 Inf   -0.3318    0.8508
##  3       0.2653 0.482 Inf   -0.6801    1.2107
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(FerritinModel_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  contrast          estimate    SE  df z.ratio p.value
##  GROUP0 - GROUP1    0.3464 0.305 Inf    1.135  0.6680
##  GROUP0 - GROUP2    0.1508 0.273 Inf    0.552  0.9461
##  GROUP0 - GROUP3    0.1190 0.368 Inf    0.324  0.9883
##  GROUP1 - GROUP2   -0.1956 0.295 Inf   -0.662  0.9112
##  GROUP1 - GROUP3   -0.2273 0.435 Inf   -0.523  0.9536
##  GROUP2 - GROUP3   -0.0318 0.402 Inf   -0.079  0.9998
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(FerritinModel_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.09354 0.335 Inf  -0.279  0.9924
## GROUP0 - GROUP2 -0.22715 0.299 Inf  -0.759  0.8727
## GROUP0 - GROUP3 -0.36239 0.399 Inf  -0.909  0.8002
## GROUP1 - GROUP2 -0.13361 0.345 Inf  -0.387  0.9803
## GROUP1 - GROUP3 -0.26885 0.470 Inf  -0.572  0.9404
## GROUP2 - GROUP3 -0.13524 0.437 Inf  -0.309  0.9897
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.58779 0.321 Inf   1.832  0.2583
## GROUP0 - GROUP2  0.21956 0.277 Inf   0.794  0.8573
## GROUP0 - GROUP3  0.26531 0.383 Inf   0.693  0.8996
## GROUP1 - GROUP2 -0.36823 0.312 Inf  -1.180  0.6394
## GROUP1 - GROUP3 -0.32248 0.456 Inf  -0.707  0.8944
## GROUP2 - GROUP3  0.04574 0.415 Inf   0.110  0.9995
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.54484 0.393 Inf   1.386  0.5081
## GROUP0 - GROUP2  0.46000 0.351 Inf   1.312  0.5555
## GROUP0 - GROUP3  0.45418 0.443 Inf   1.026  0.7340
## GROUP1 - GROUP2 -0.08483 0.354 Inf  -0.239  0.9952
## GROUP1 - GROUP3 -0.09065 0.522 Inf  -0.174  0.9981
## GROUP2 - GROUP3 -0.00582 0.482 Inf  -0.012  1.0000
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(FerritinModel_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1    -0.3663 0.204 Inf  -1.797  0.1706
## t0 - t2    -0.9169 0.404 Inf  -2.272  0.0598
## t1 - t2    -0.5506 0.294 Inf  -1.876  0.1457
##
## GROUP = 1:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1     0.3151 0.241 Inf   1.310  0.3895
## t0 - t2    -0.2785 0.356 Inf  -0.781  0.7145
## t1 - t2    -0.5936 0.213 Inf  -2.783  0.0149
##
## GROUP = 2:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1     0.0804 0.224 Inf   0.359  0.9316
## t0 - t2    -0.2297 0.384 Inf  -0.599  0.8209
## t1 - t2    -0.3102 0.251 Inf  -1.236  0.4317
##
## GROUP = 3:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1     0.2614 0.286 Inf   0.914  0.6314
## t0 - t2    -0.1003 0.464 Inf  -0.216  0.9745
## t1 - t2    -0.3617 0.352 Inf  -1.026  0.5602
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

18 Model Neurogranin

```
bptest(lm(Neurogranin ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 +
(1 | id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(Neurogranin ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severit
y_0 +      (1 | id), data = db)
## BP = 28.196, df = 14, p-value = 0.0134
```

```
vars <- c("Neurogranin","t","GROUP","Age","Sex","CIRS_severity_0","id")
db_cc <- db[complete.cases(db[, vars]), vars]

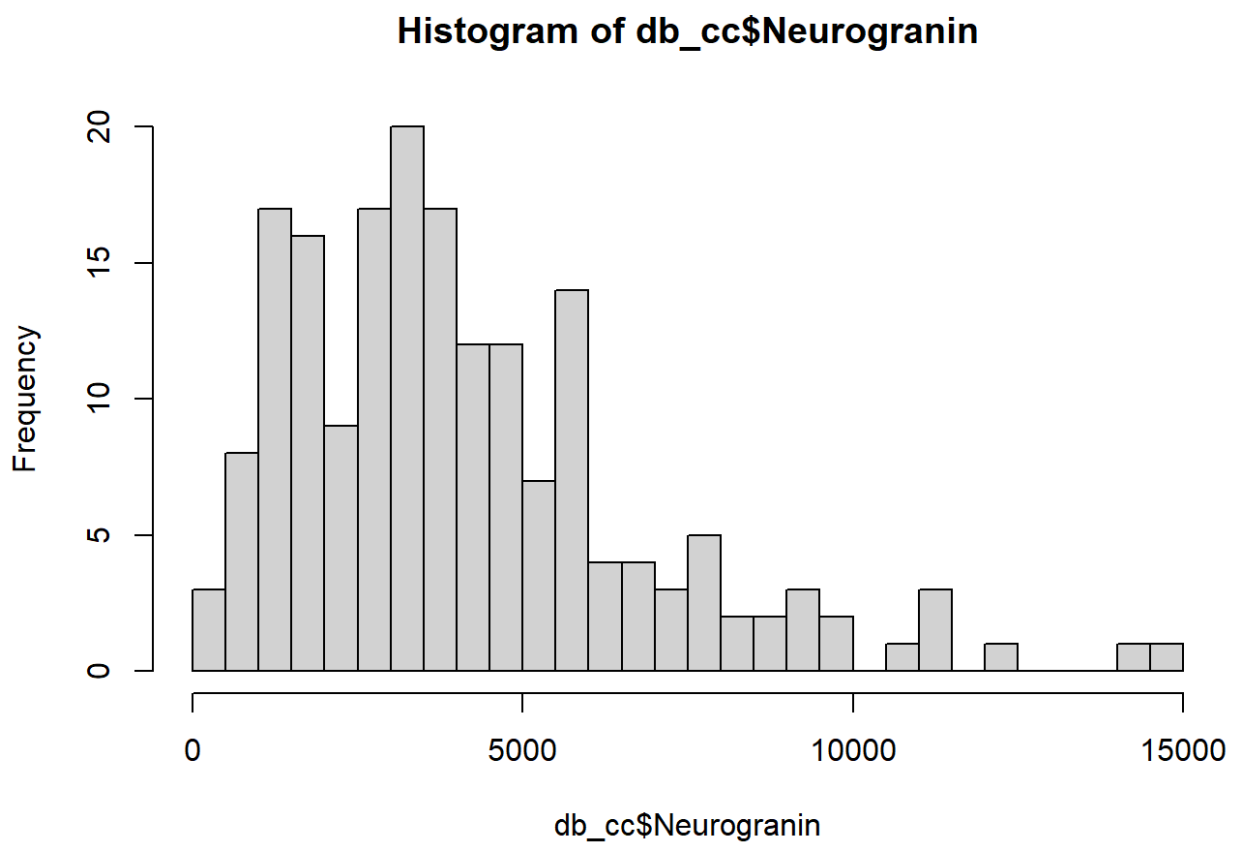
nrow(db); nrow(db_cc)
```



```
## [1] 243
```

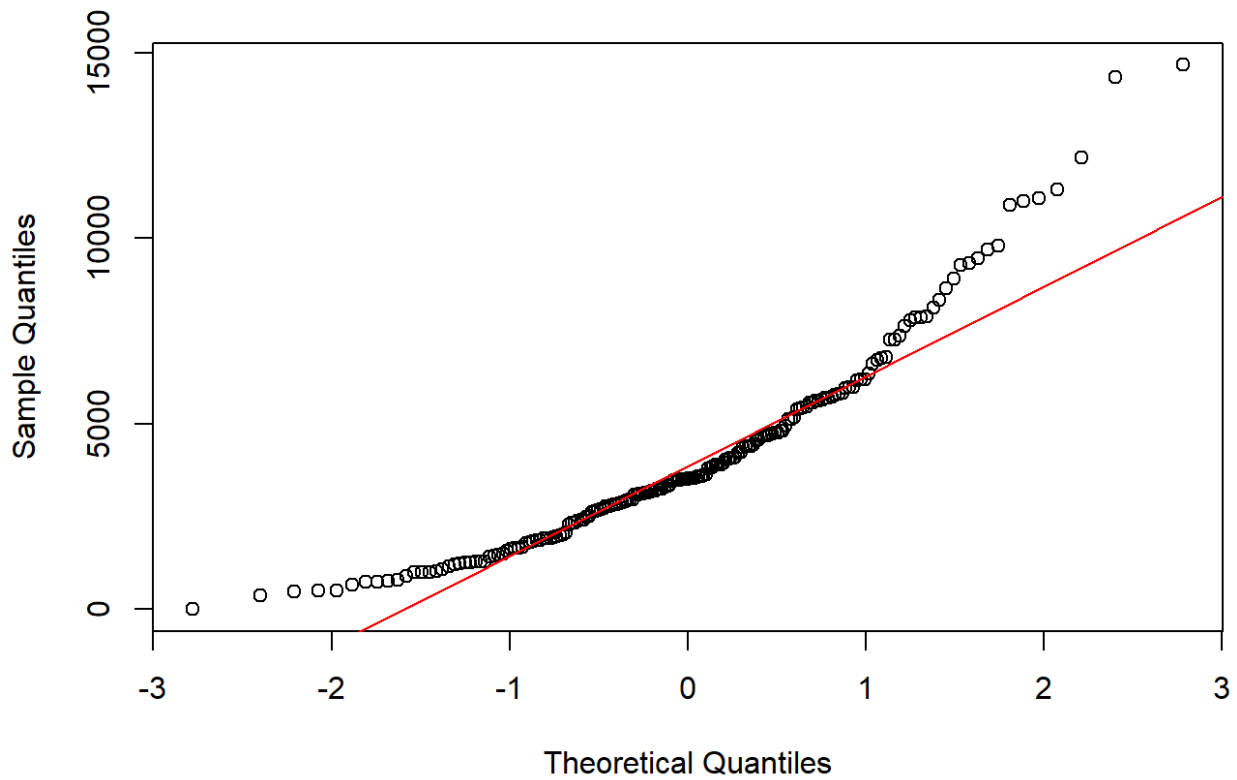
```
## [1] 184
```

```
hist(db_cc$Neurogranin, breaks=30)
```



```
qqnorm(db_cc$Neurogranin); qqline(db_cc$Neurogranin, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$Neurogranin)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
##    4.883  2225.177  3513.528  4116.916  5496.101 14668.996
```

```
min(db_cc$Neurogranin, na.rm=TRUE)
```

```
## [1] 4.882812
```

```
db$Neurogranin_z <- scale(log(db$Neurogranin))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$Neurogranin_z <- scale(log(db_cc$Neurogranin))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

NeurograninModel_hetero <- glmmTMB(
  Neurogranin_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | i
d),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(NeurograninModel_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## Neurogranin_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
## (1 | id)
## Dispersion: ~t
## Data: db_cc
##
##      AIC      BIC    logLik -2*log(L)  df.resid
##    502.7    563.8    -232.3    464.7      165
##
## Random effects:
##
## Conditional model:
##   Groups   Name      Variance Std.Dev.
##   id      (Intercept) 0.04262  0.2065
##   Residual              NA      NA
## Number of obs: 184, groups: id, 80
##
## Conditional model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    9.16284    2.28038   4.018 5.87e-05 ***
## t1              0.12020    0.28336   0.424  0.6714
## t2              0.14909    0.39699   0.376  0.7073
## GROUP1         -0.51553    0.40797  -1.264  0.2063
## GROUP2          0.24227    0.36214   0.669  0.5035
## GROUP3         -0.01000    0.46447  -0.022  0.9828
## Age            -0.11861    0.02892  -4.101 4.11e-05 ***
## SexM            0.11519    0.13336   0.864  0.3877
## CIRS_severity_0 -0.09480    0.16553  -0.573  0.5668
## t1:GROUP1       0.86449    0.47295   1.828  0.0676 .
## t2:GROUP1       0.89085    0.51606   1.726  0.0843 .
## t1:GROUP2      -0.14558    0.42695  -0.341  0.7331
## t2:GROUP2       0.08577    0.46363   0.185  0.8532
## t1:GROUP3       0.55213    0.57036   0.968  0.3330
## t2:GROUP3       0.56700    0.57532   0.986  0.3244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.06978    0.09721   0.718  0.47286
## t1          -0.35574    0.13714  -2.594  0.00949 **
## t2          -0.39704    0.14597  -2.720  0.00653 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

NeurograninModel_homo <- glmmTMB(
  Neurogranin_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 |
id),
  family = gaussian(),
  data = db_cc
)

anova(NeurograninModel_homo, NeurograninModel_hetero, test="Chisq")

```

```

## Data: db_cc
## Models:
## NeurograninModel_homo: Neurogranin_z ~ t + GROUP + GROUP * t + Age + Sex
+ CIRS_severity_0 + , zi=~0, disp=~1
## NeurograninModel_homo:      (1 | id), zi=~0, disp=~t
## NeurograninModel_hetero: Neurogranin_z ~ t + GROUP + GROUP * t + Age + S
ex + CIRS_severity_0 + , zi=~0, disp=~1
## NeurograninModel_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance Chisq Chi Df
## NeurograninModel_homo    17 509.29 563.94 -237.64   475.29
## NeurograninModel_hetero   19 502.70 563.78 -232.35   464.70 10.59      2
##
##           Pr(>Chisq)
## NeurograninModel_homo
## NeurograninModel_hetero  0.005018 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

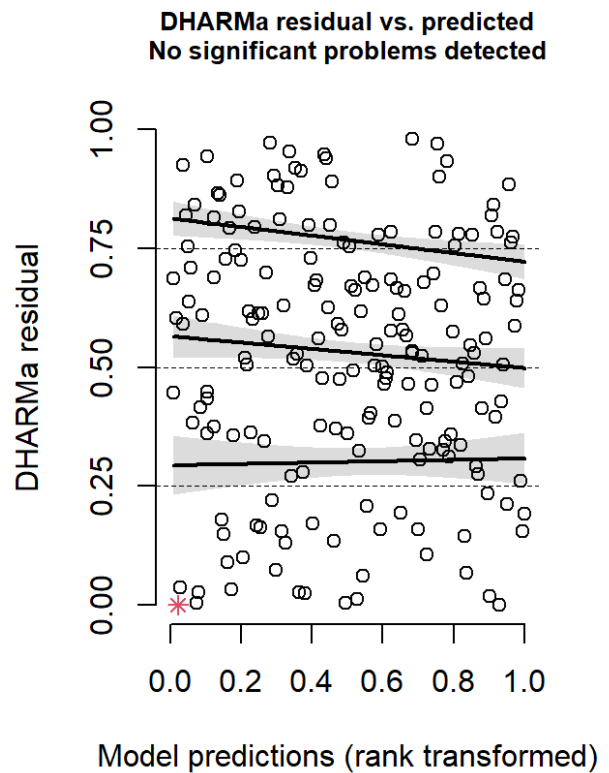
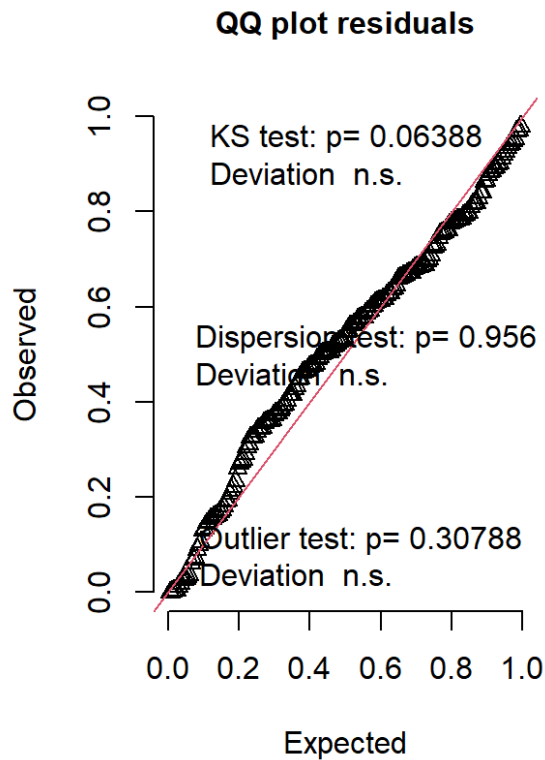
```

```

sim_res <- simulateResiduals(NeurograninModel_hetero, n = 1000)
plot(sim_res)

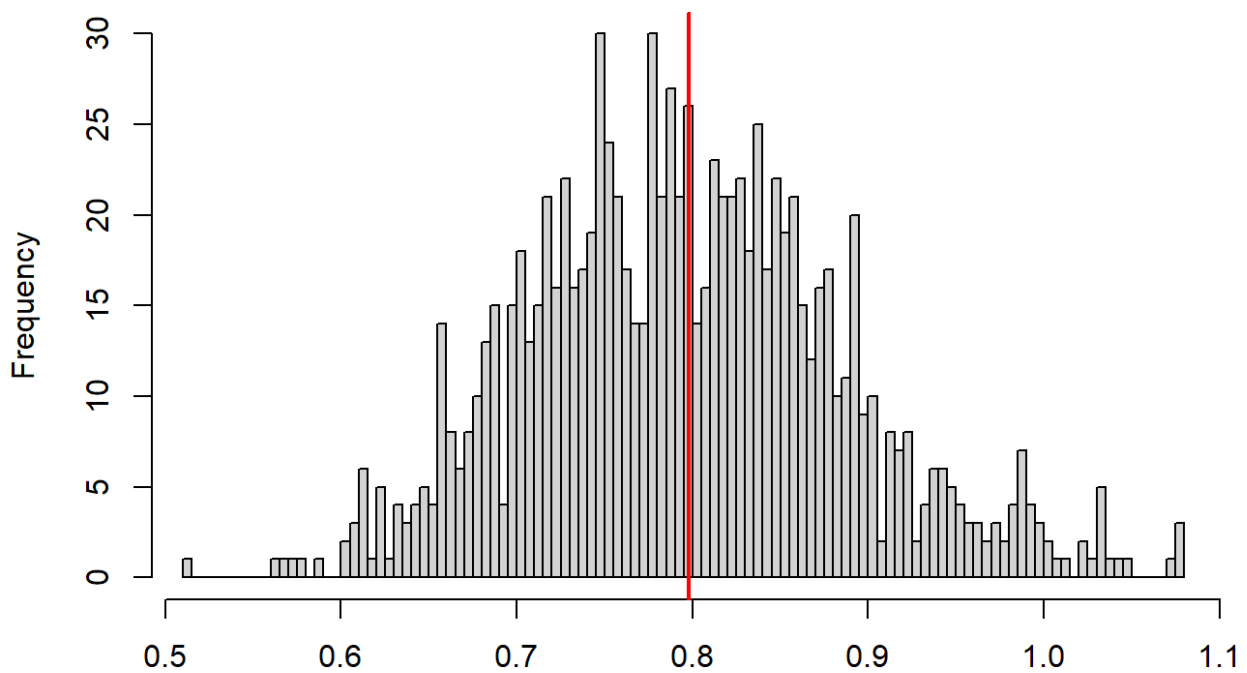
```

DHARMA residual



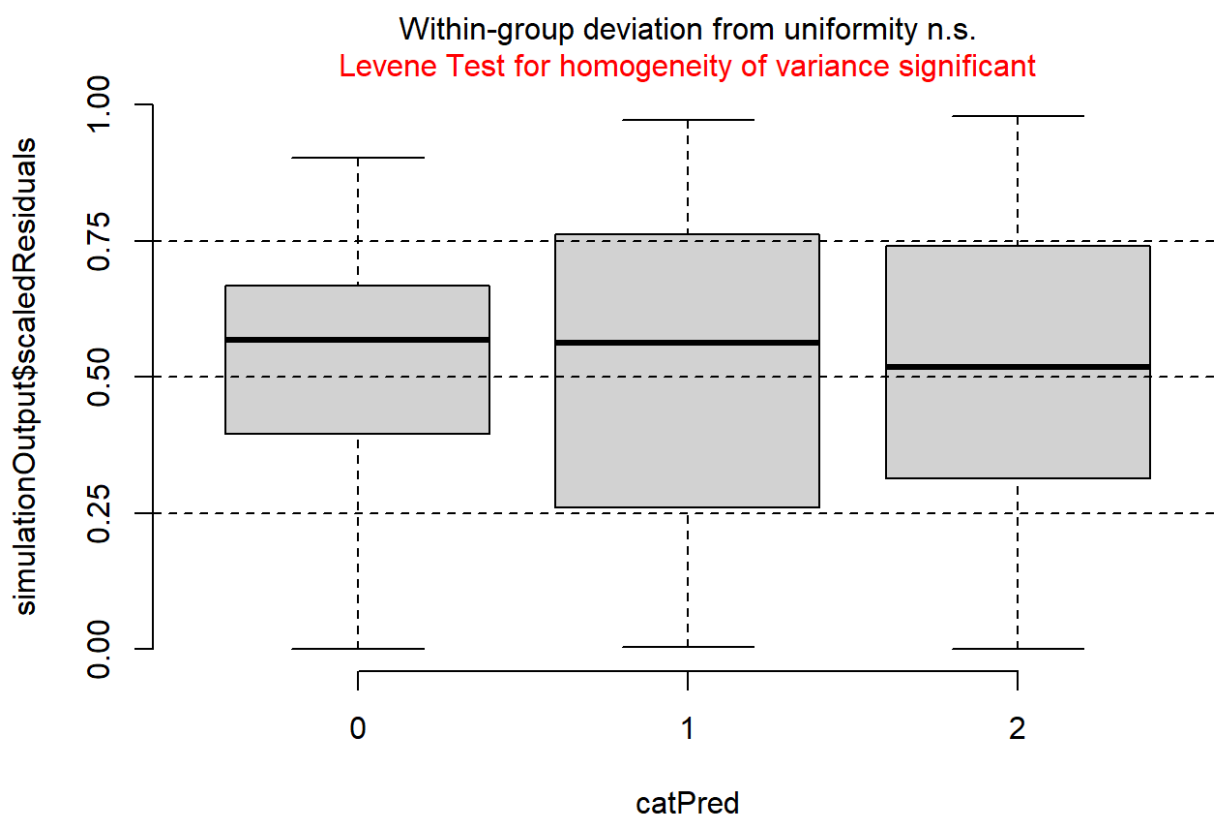
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated



```
##  
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.  
## simulated  
##  
## data: simulationOutput  
## dispersion = 1.0023, p-value = 0.956  
## alternative hypothesis: two.sided
```

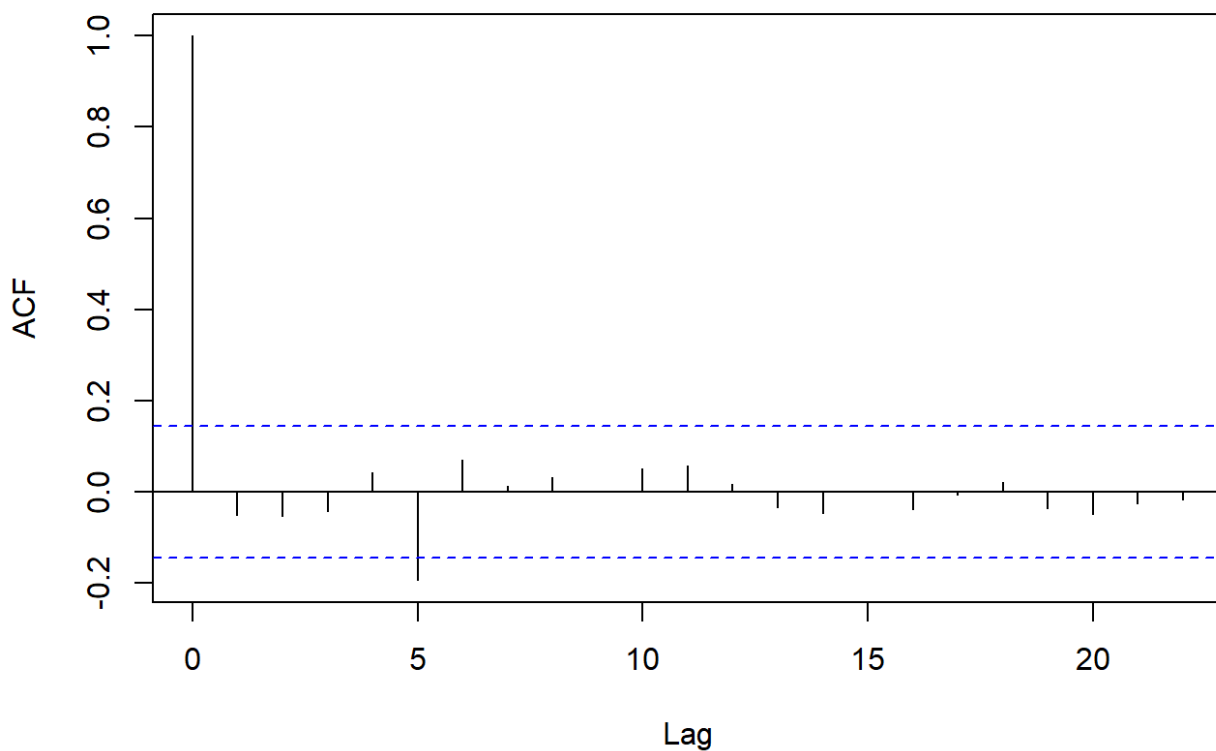
```
testQuantiles(sim_res, db_cc$t)
```



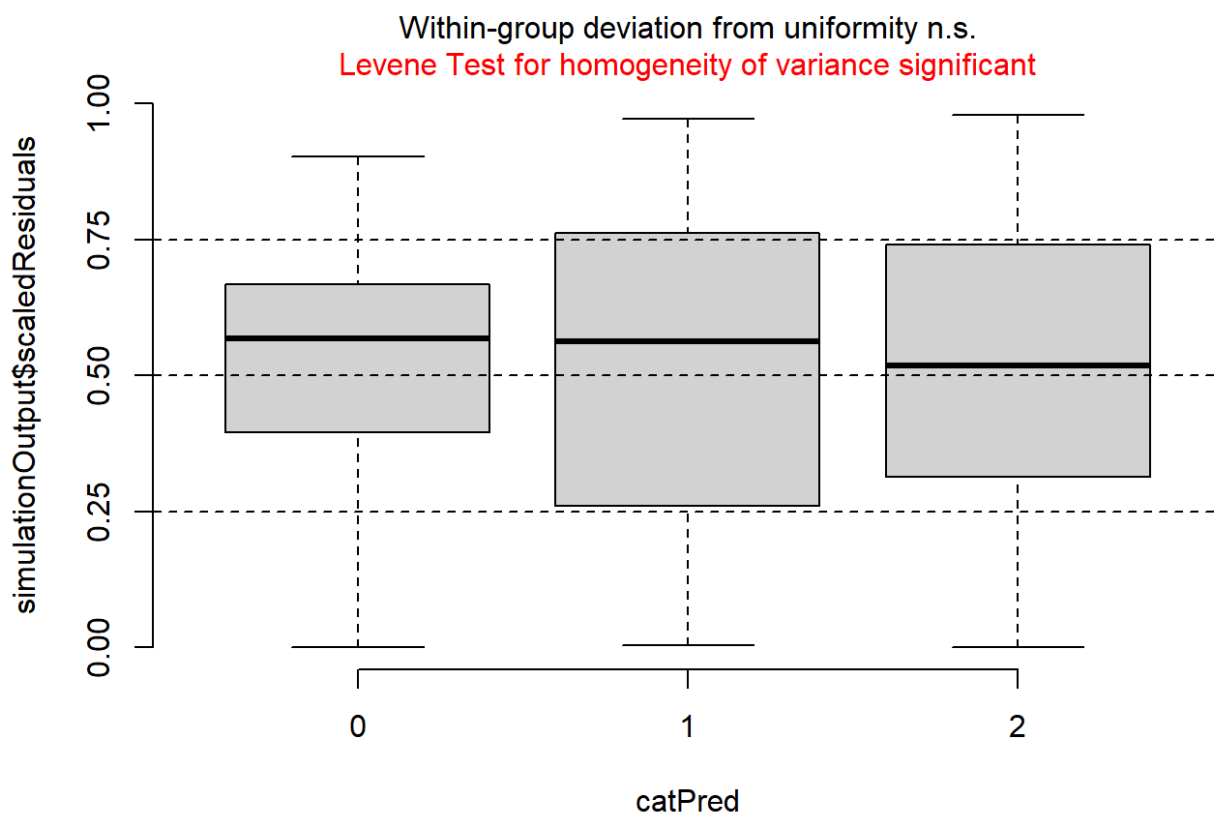
```
## NULL
```

```
acf(residuals(NeurograninModel_hetero))
```

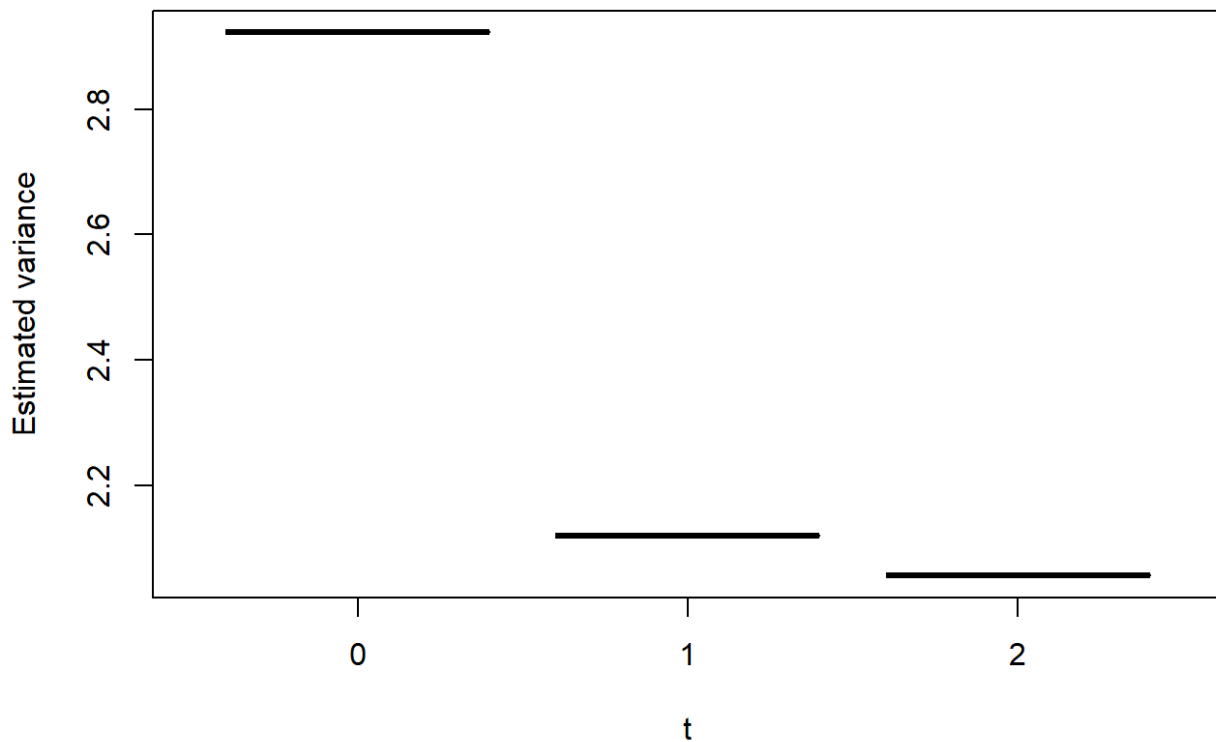
Series residuals(NeurograninModel_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

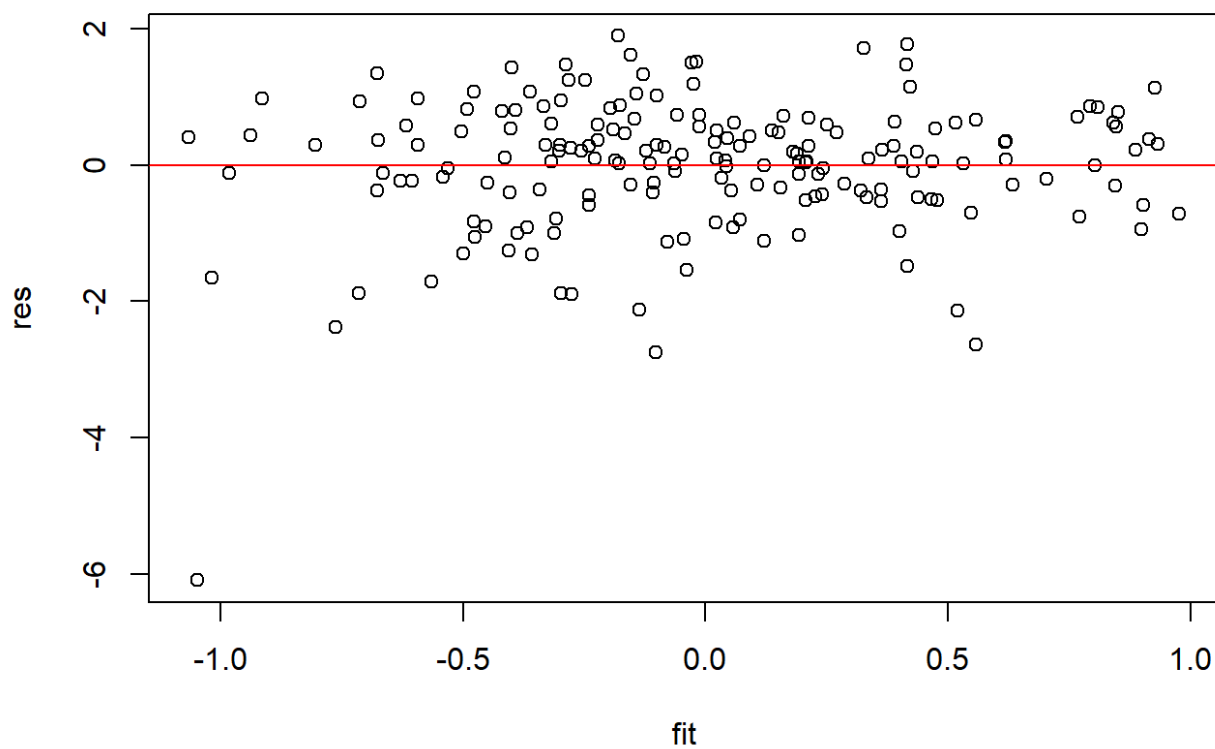



```
pred_disp <- predict(NeurograninModel_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(NeurograninModel_hetero, type="pearson")
fit <- fitted(NeurograninModel_hetero)

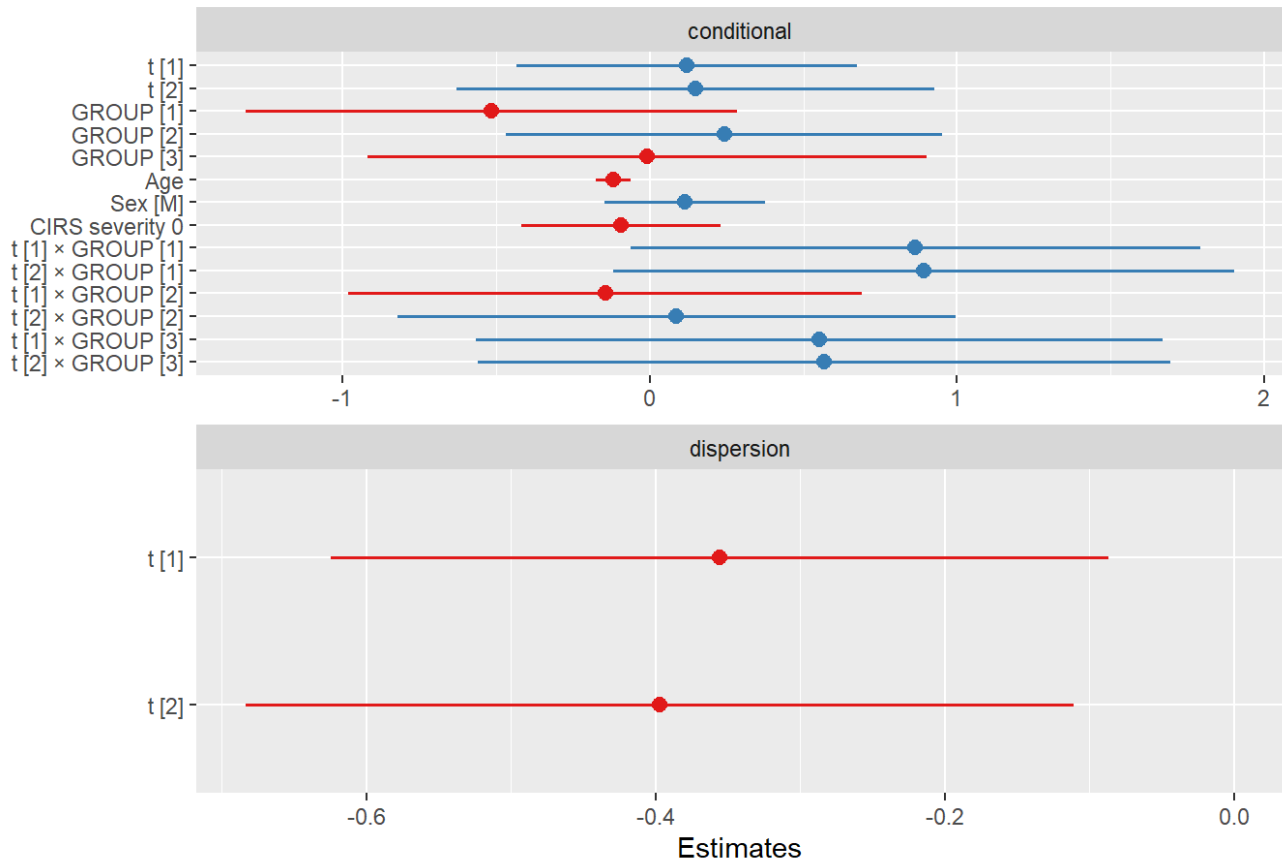
plot(fit, res)
abline(h=0, col="red")
```



18.1 Visualizzazione del modello

```
plot_model(NeurograninModel_hetero)
```

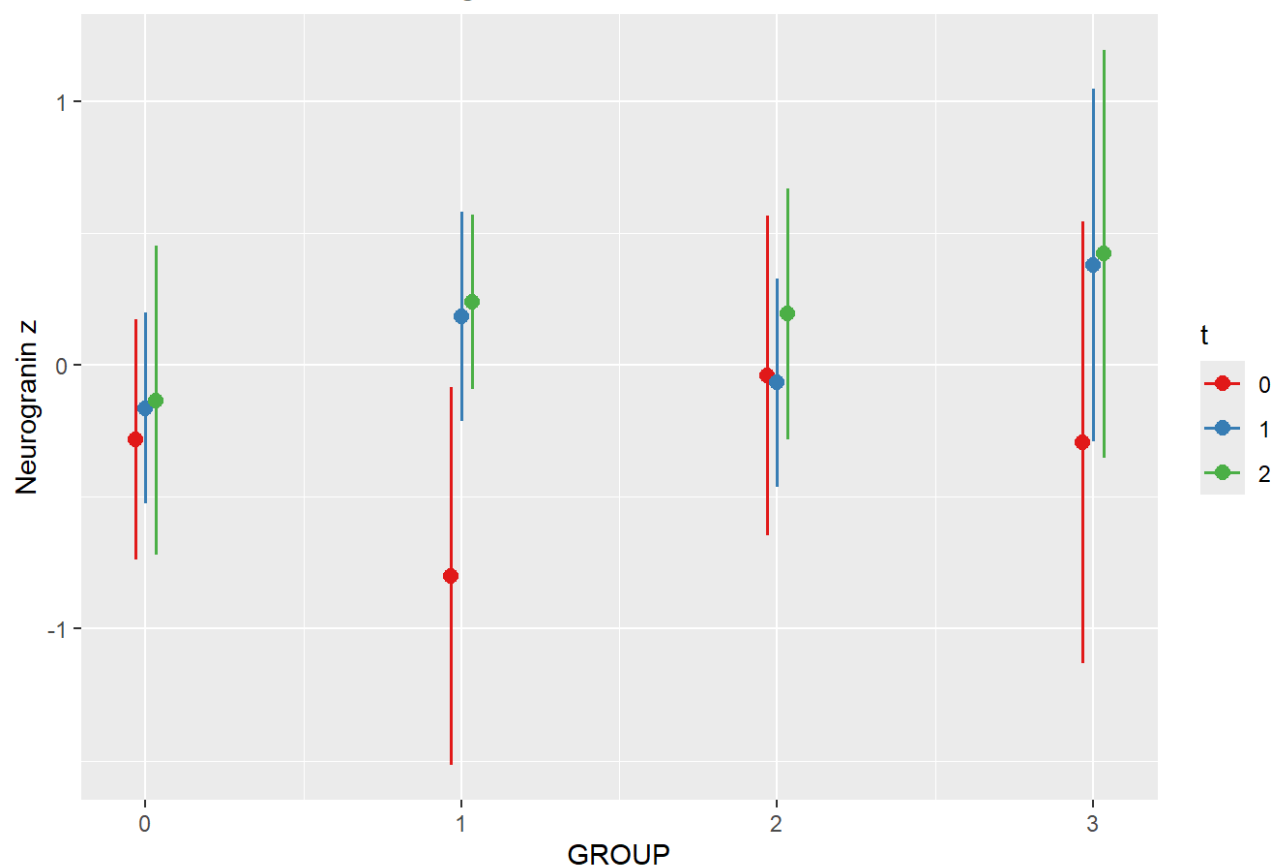
Neurogranin z



```
plot_model(NeurograninModel_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

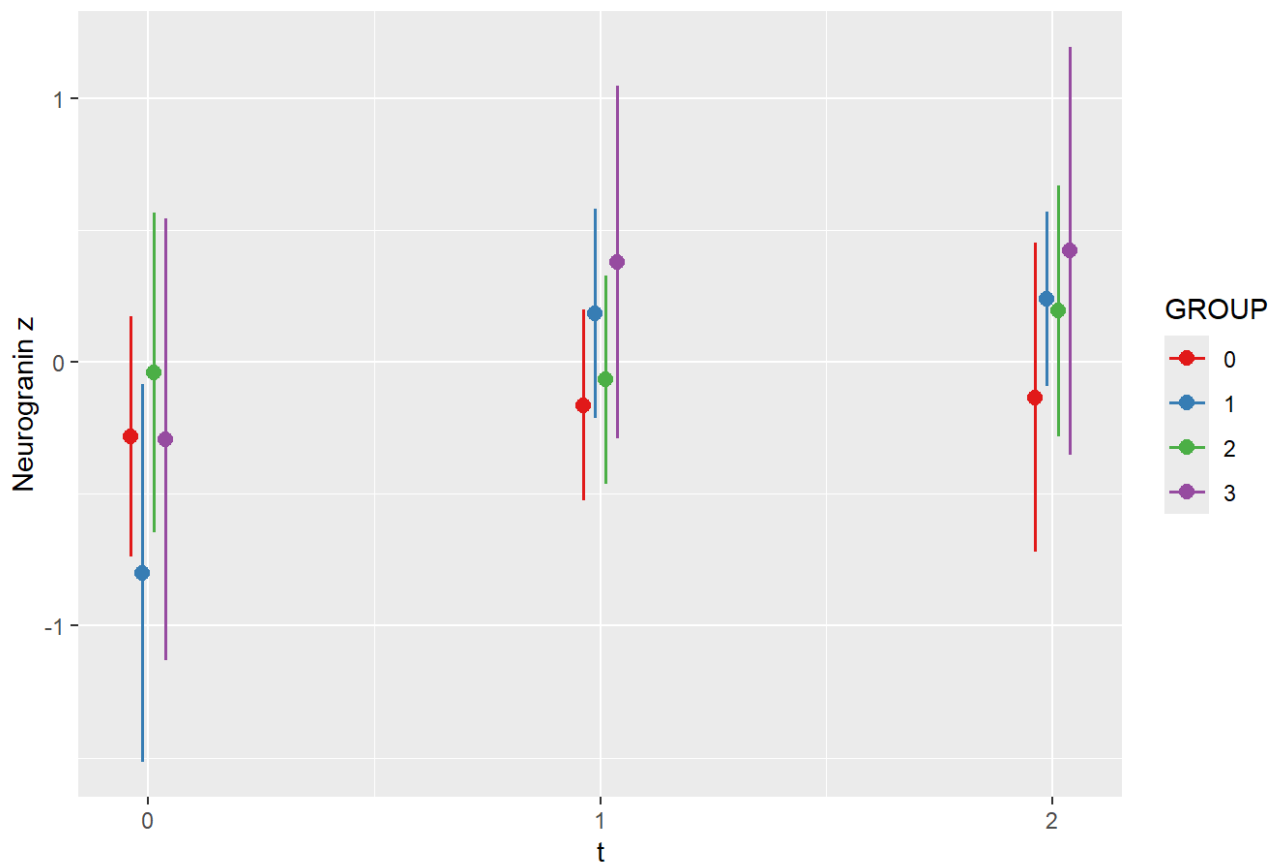
Predicted values of Neurogranin z



```
plot_model(NeurograninModel_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of Neurogranin z

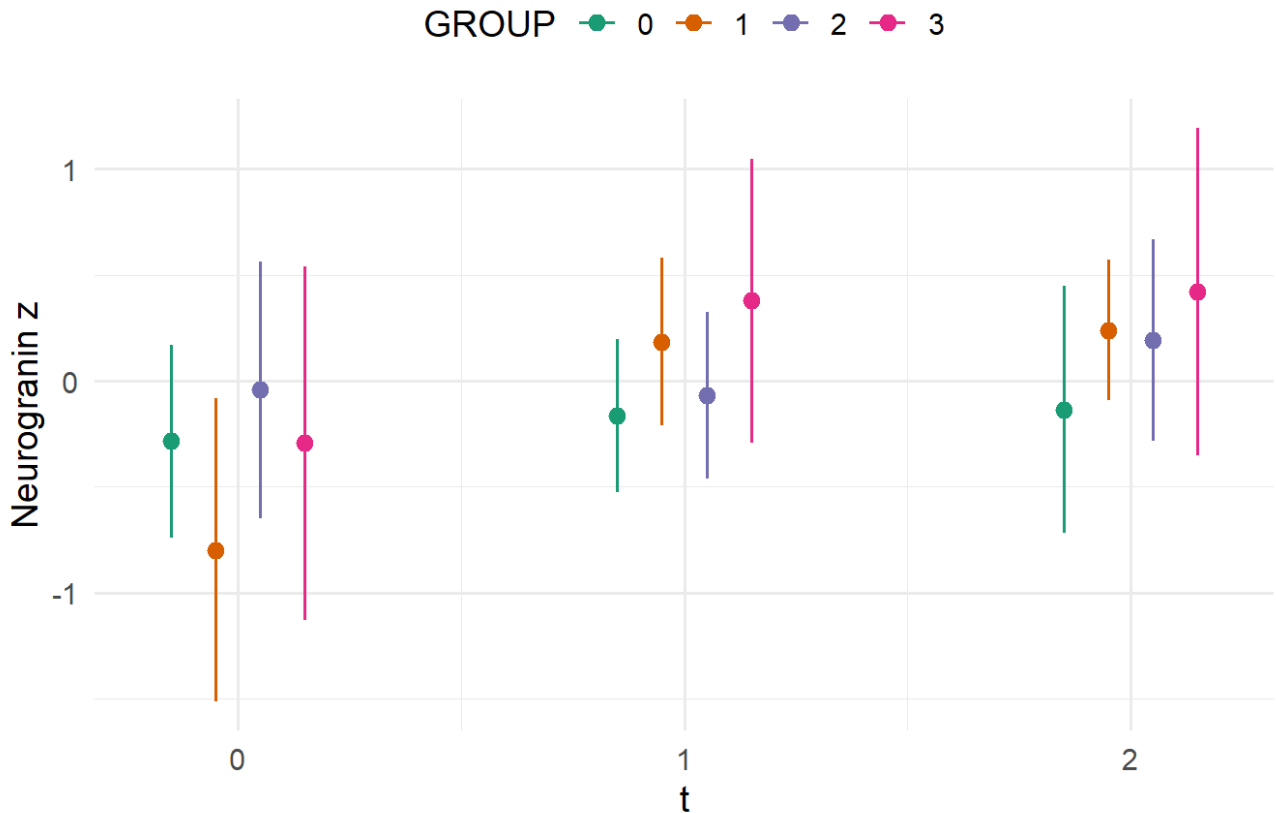


```
plot_model(NeurograninModel_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

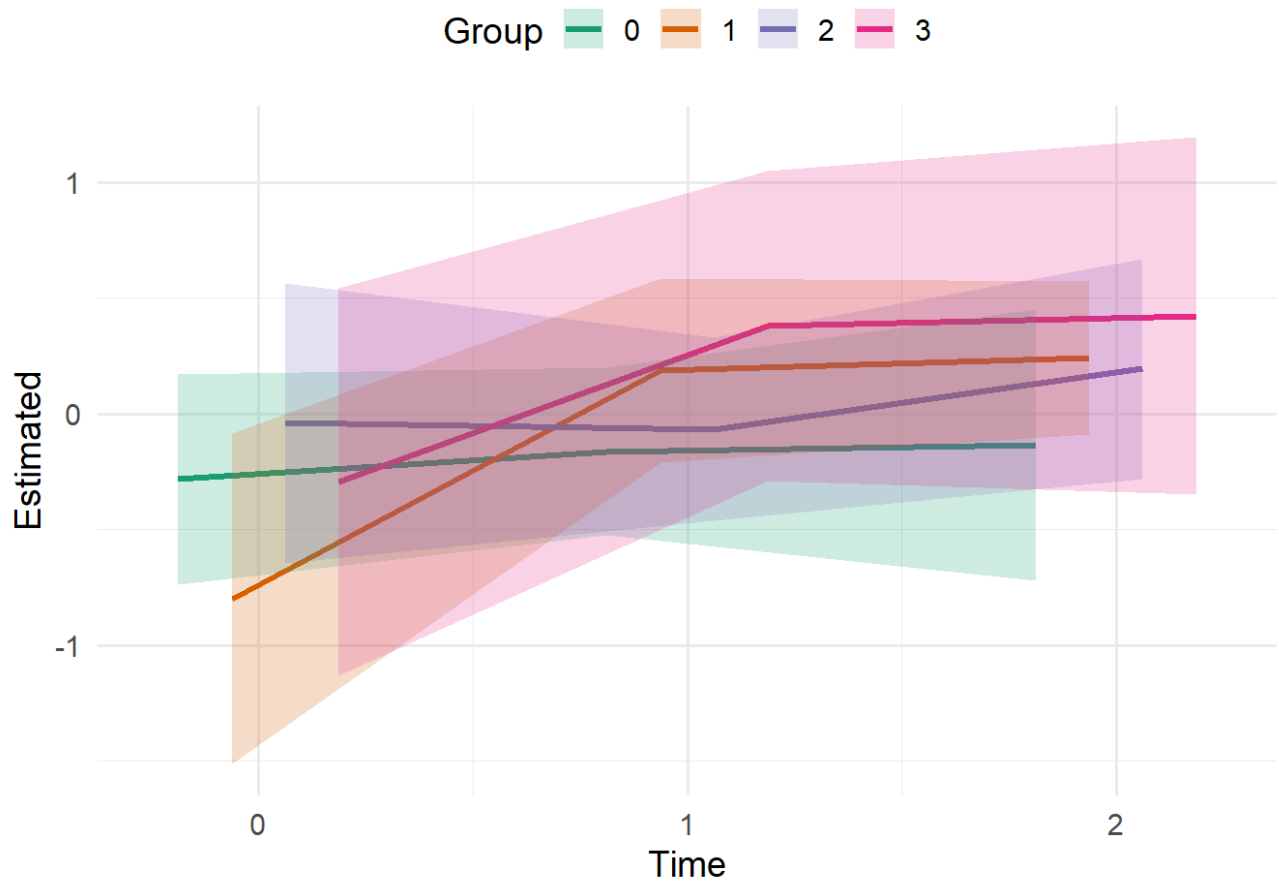
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of Neurogranin z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(NeurograninModel_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```



18.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(NeurograninModel_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP    emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.22623 0.222 Inf   -0.6613    0.2089
##  1      -0.74177 0.364 Inf   -1.4551   -0.0284
##  2       0.01603 0.308 Inf   -0.5875    0.6196
##  3      -0.23624 0.417 Inf   -1.0532    0.5807
##
## t = 1:
##  GROUP    emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.10604 0.170 Inf   -0.4400    0.2279
##  1       0.24292 0.197 Inf   -0.1440    0.6298
##  2      -0.00935 0.197 Inf   -0.3953    0.3767
##  3       0.43609 0.331 Inf   -0.2127    1.0848
##
## t = 2:
##  GROUP    emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.07715 0.291 Inf   -0.6469    0.4926
##  1       0.29817 0.163 Inf   -0.0211    0.6175
##  2       0.25089 0.237 Inf   -0.2140    0.7158
##  3       0.47985 0.384 Inf   -0.2728    1.2325
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(NeurograninModel_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.0696 0.219 Inf   -0.318  0.9889
## GROUP0 - GROUP2 -0.2223 0.189 Inf   -1.175  0.6426
## GROUP0 - GROUP3 -0.3630 0.242 Inf   -1.498  0.4385
## GROUP1 - GROUP2 -0.1528 0.213 Inf   -0.716  0.8908
## GROUP1 - GROUP3 -0.2935 0.296 Inf   -0.991  0.7548
## GROUP2 - GROUP3 -0.1407 0.268 Inf   -0.524  0.9533
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(NeurograninModel_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```



```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1    0.5155 0.408 Inf   1.264  0.5862
## GROUP0 - GROUP2   -0.2423 0.362 Inf  -0.669  0.9088
## GROUP0 - GROUP3    0.0100 0.464 Inf   0.022  1.0000
## GROUP1 - GROUP2   -0.7578 0.452 Inf  -1.676  0.3365
## GROUP1 - GROUP3   -0.5055 0.545 Inf  -0.927  0.7902
## GROUP2 - GROUP3    0.2523 0.511 Inf   0.494  0.9605
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1   -0.3490 0.279 Inf  -1.252  0.5934
## GROUP0 - GROUP2   -0.0967 0.249 Inf  -0.388  0.9801
## GROUP0 - GROUP3   -0.5421 0.356 Inf  -1.523  0.4236
## GROUP1 - GROUP2    0.2523 0.281 Inf   0.898  0.8057
## GROUP1 - GROUP3   -0.1932 0.407 Inf  -0.475  0.9646
## GROUP2 - GROUP3   -0.4454 0.378 Inf  -1.177  0.6413
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1   -0.3753 0.339 Inf  -1.108  0.6847
## GROUP0 - GROUP2   -0.3280 0.307 Inf  -1.068  0.7091
## GROUP0 - GROUP3   -0.5570 0.364 Inf  -1.532  0.4185
## GROUP1 - GROUP2    0.0473 0.287 Inf   0.165  0.9984
## GROUP1 - GROUP3   -0.1817 0.423 Inf  -0.430  0.9733
## GROUP2 - GROUP3   -0.2290 0.394 Inf  -0.582  0.9376
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(NeurograninModel_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```
## GROUP = 0:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.1202 0.283 Inf  -0.424  0.9056
## t0 - t2      -0.1491 0.397 Inf  -0.376  0.9252
## t1 - t2      -0.0289 0.333 Inf  -0.087  0.9959
##
## GROUP = 1:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.9847 0.380 Inf  -2.590  0.0260
## t0 - t2      -1.0399 0.396 Inf  -2.623  0.0237
## t1 - t2      -0.0553 0.250 Inf  -0.221  0.9735
##
## GROUP = 2:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1       0.0254 0.358 Inf   0.071  0.9972
## t0 - t2      -0.2349 0.407 Inf  -0.578  0.8320
## t1 - t2      -0.2602 0.304 Inf  -0.857  0.6672
##
## GROUP = 3:
## contrast estimate      SE  df z.ratio p.value
## t0 - t1      -0.6723 0.525 Inf  -1.281  0.4058
## t0 - t2      -0.7161 0.574 Inf  -1.247  0.4257
## t1 - t2      -0.0438 0.477 Inf  -0.092  0.9954
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates
```

19 Model TREM2

```
bptest(lm(TREM2 ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 |
id), data = db))
```

```
##
## studentized Breusch-Pagan test
##
## data:  lm(TREM2 ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +
(1 | id), data = db)
## BP = 9.8563, df = 14, p-value = 0.7726
```

```
vars <- c("TREM2", "t", "GROUP", "Age", "Sex", "CIRS_severity_0", "id")
db_cc <- db[complete.cases(db[, vars]), vars]

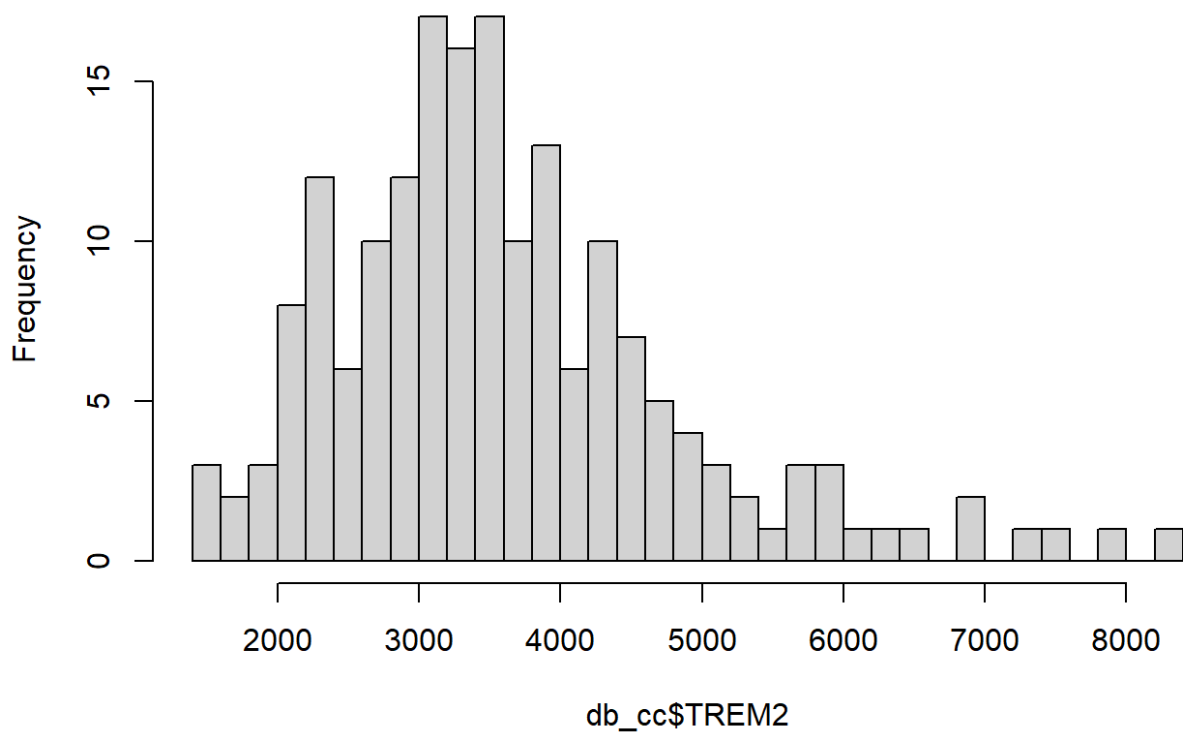
nrow(db); nrow(db_cc)
```

```
## [1] 243
```

```
## [1] 182
```

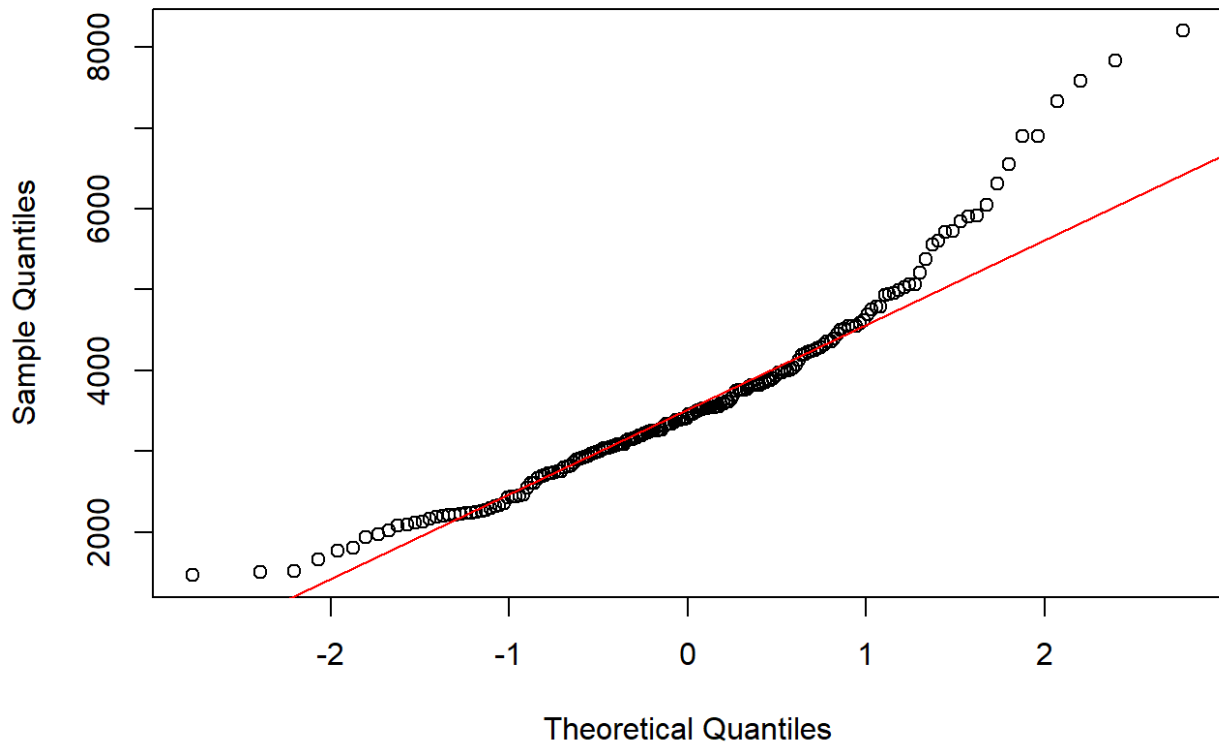
```
hist(db_cc$TREM2, breaks=30)
```

Histogram of db_cc\$TREM2



```
qqnorm(db_cc$TREM2); qqline(db_cc$TREM2, col="red")
```

Normal Q-Q Plot



```
summary(db_cc$TREM2)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	1465	2811	3432	3623	4224	8202

```
min(db_cc$TREM2, na.rm=TRUE)
```

```
## [1] 1465.277
```

```
db$TREM2_z <- scale(log(db$TREM2))
db$Age_z <- scale(db$Age)
db$CIRS_severity_0_z <- scale(db$CIRS_severity_0)

db_cc$TREM2_z <- scale(log(db_cc$TREM2))
db_cc$Age_z <- scale(db_cc$Age)
db_cc$CIRS_severity_0_z <- scale(db_cc$CIRS_severity_0)

TREM2Model_hetero <- glmmTMB(
  TREM2_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  dispformula = ~ t,
  family = gaussian(),
  data = db_cc
)
summary(TREM2Model_hetero)
```

```

## Family: gaussian ( identity )
## Formula:
## TREM2_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_severity_0 +      (1
| id)
## Dispersion:                ~t
## Data: db_cc
##
##           AIC           BIC      logLik -2*log(L)  df.resid
##          406.7          467.5      -184.3     368.7      163
##
## Random effects:
##
## Conditional model:
##   Groups   Name              Variance Std.Dev.
##   id       (Intercept) 0.5503   0.7418
##   Residual                NA         NA
## Number of obs: 182, groups: id, 80
##
## Conditional model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.24391    2.84755  -0.086  0.93174
## t1              0.23348    0.19442   1.201  0.22979
## t2            -0.22984    0.37510  -0.613  0.54005
## GROUP1        -0.37823    0.32295  -1.171  0.24153
## GROUP2        -0.87975    0.28781  -3.057  0.00224 **
## GROUP3       -1.06658    0.38338  -2.782  0.00540 **
## Age             0.00183    0.03700   0.049  0.96056
## SexM          -0.28030    0.18659  -1.502  0.13303
## CIRS_severity_0 0.07697    0.12898   0.597  0.55070
## t1:GROUP1       0.61874    0.25331   2.443  0.01458 *
## t2:GROUP1       1.40477    0.33510   4.192 2.76e-05 ***
## t1:GROUP2       1.04490    0.20272   5.154 2.54e-07 ***
## t2:GROUP2       1.82365    0.28088   6.493 8.44e-11 ***
## t1:GROUP3       1.36532    0.26806   5.093 3.52e-07 ***
## t2:GROUP3       2.02528    0.33473   6.050 1.44e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Dispersion model:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.67233    0.13050  -5.152 2.58e-07 ***
## t1          -0.53242    0.28662  -1.858  0.0632 .
## t2          -0.03203    0.18035  -0.178  0.8590
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

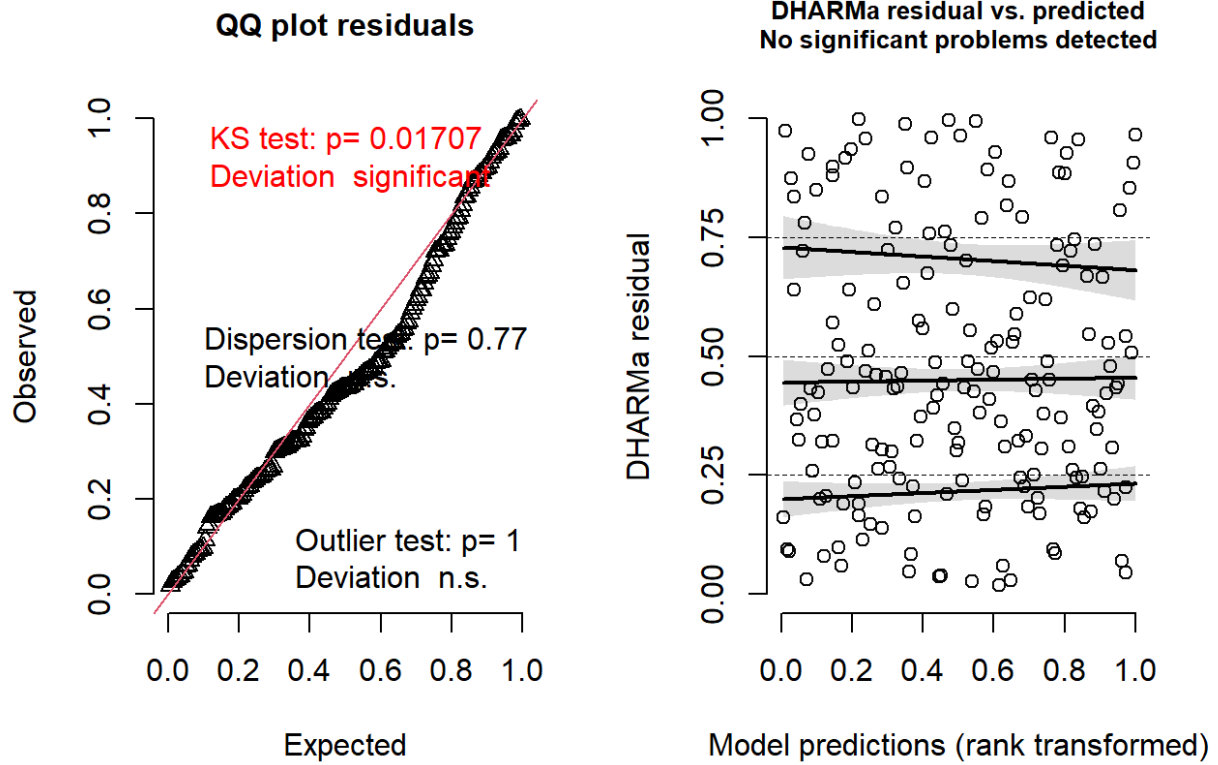
```
TREM2Model_homo <- glmmTMB(
  TREM2_z ~ t + GROUP + GROUP*t + Age + Sex + CIRS_severity_0 + (1 | id),
  family = gaussian(),
  data = db_cc
)

anova(TREM2Model_homo, TREM2Model_hetero, test="Chisq")
```

```
## Data: db_cc
## Models:
## TREM2Model_homo: TREM2_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_seve
rity_0 + , zi=~0, disp=~1
## TREM2Model_homo:      (1 | id), zi=~0, disp=~t
## TREM2Model_hetero: TREM2_z ~ t + GROUP + GROUP * t + Age + Sex + CIRS_se
verity_0 + , zi=~0, disp=~1
## TREM2Model_hetero:      (1 | id), zi=~0, disp=~t
##
##           Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Ch
isq)
## TREM2Model_homo    17 407.94 462.41 -186.97   373.94
## TREM2Model_hetero  19 406.65 467.53 -184.33   368.65 5.2872      2    0.0
7111 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

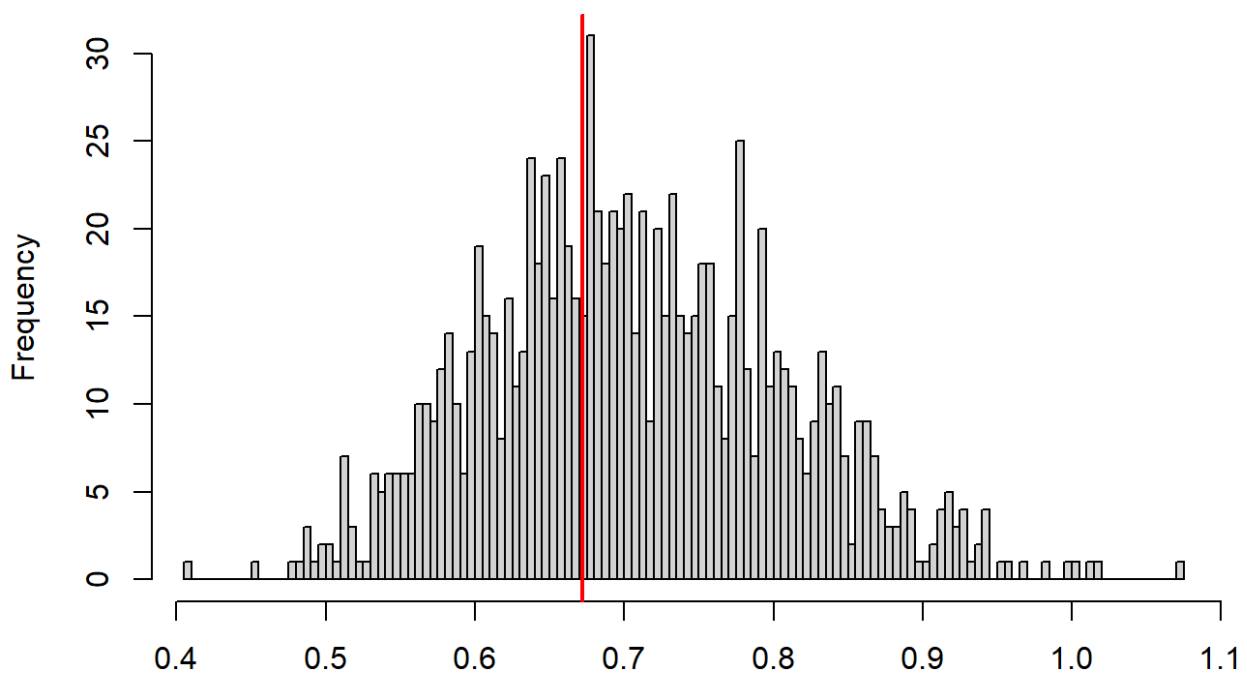
```
sim_res <- simulateResiduals(TREM2Model_hetero, n = 1000)
plot(sim_res)
```

DHARMA residual



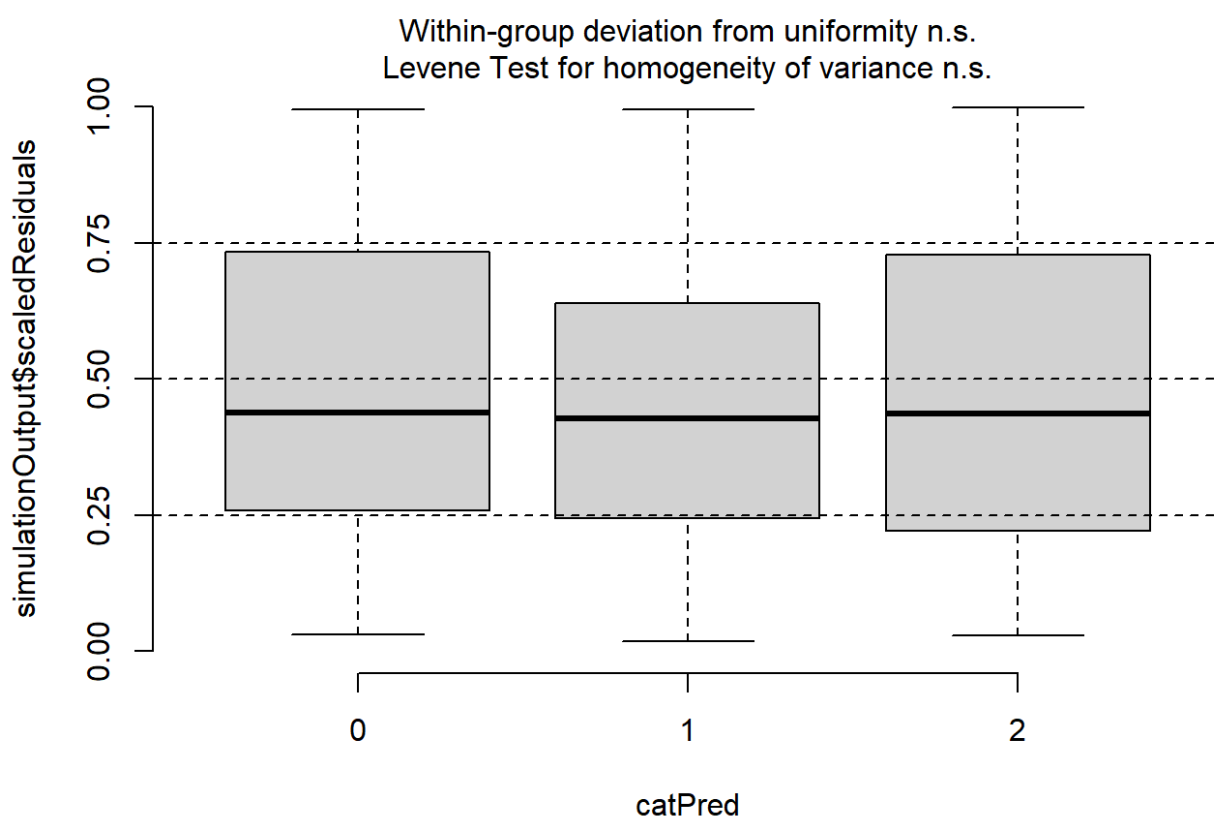
```
testDispersion(sim_res)
```

DHARMA nonparametric dispersion test via sd of residuals fitted vs. simulated




```
##
## DHARMA nonparametric dispersion test via sd of residuals fitted vs.
## simulated
##
## data: simulationOutput
## dispersion = 0.95093, p-value = 0.77
## alternative hypothesis: two.sided
```

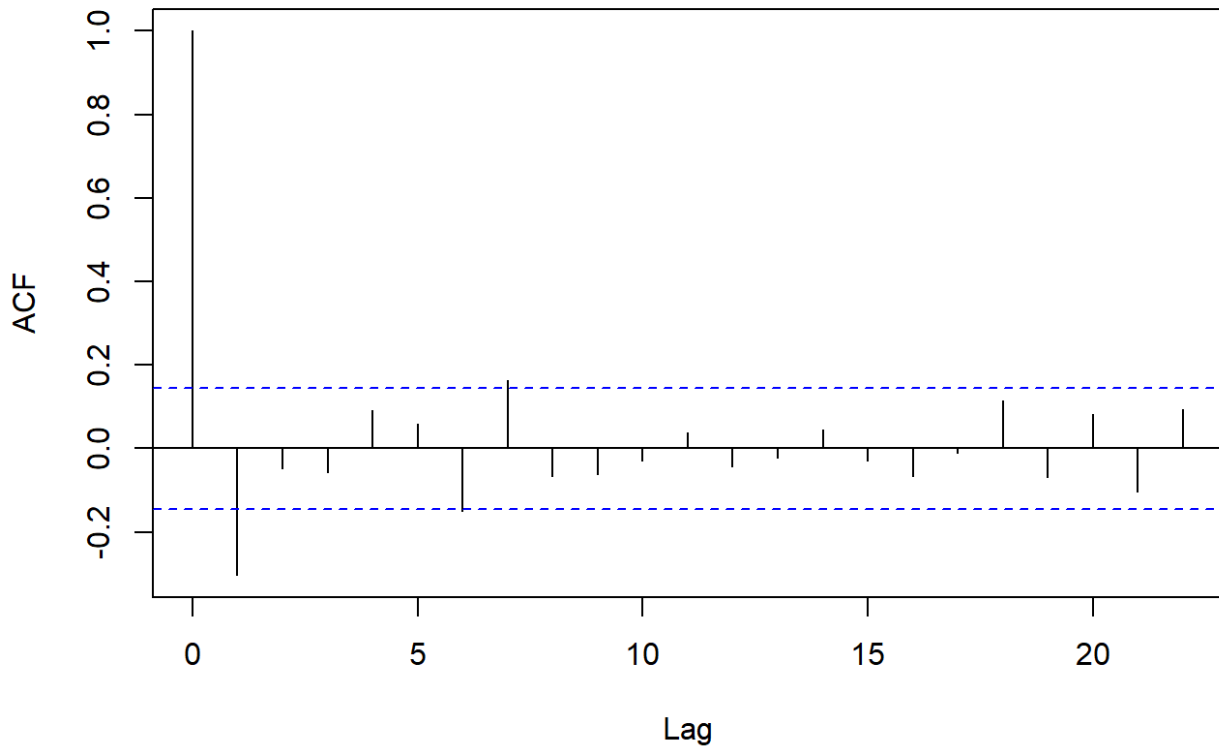
```
testQuantiles(sim_res, db_cc$t)
```



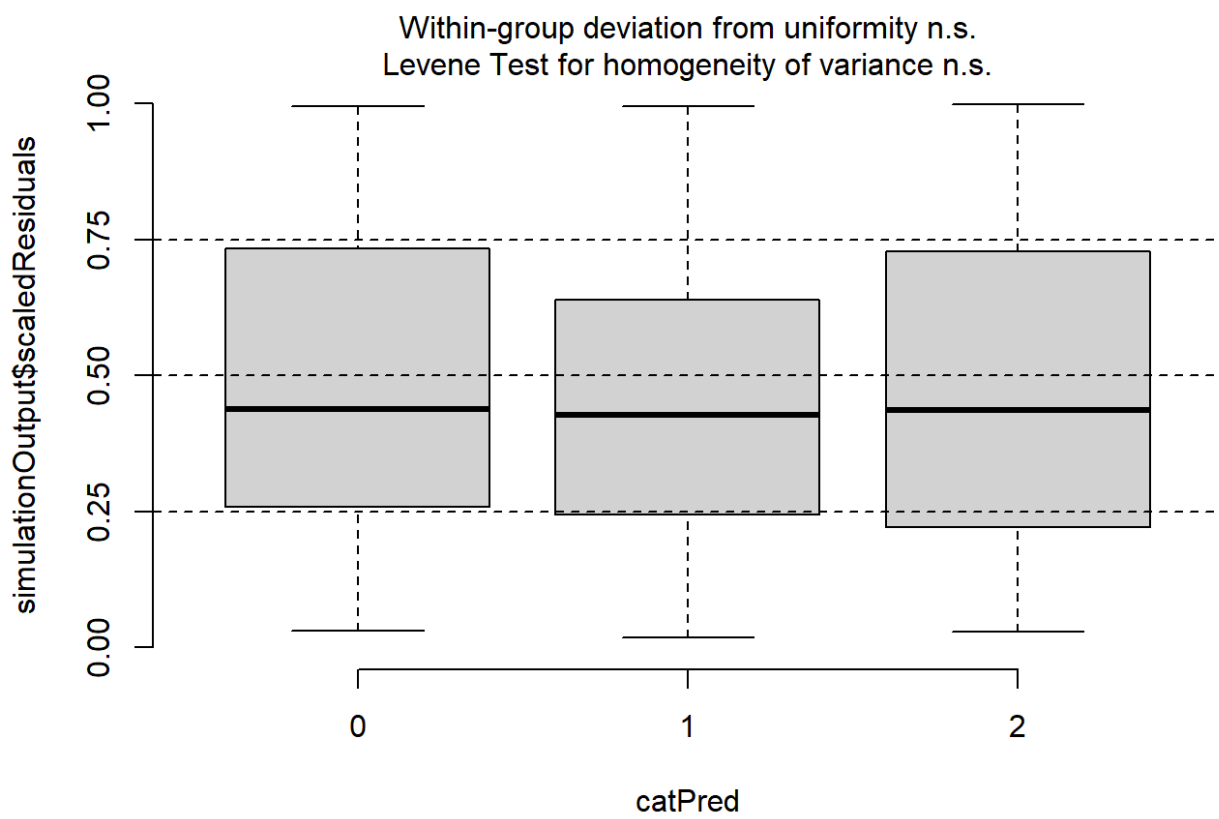
```
## NULL
```

```
acf(residuals(TREM2Model_hetero))
```

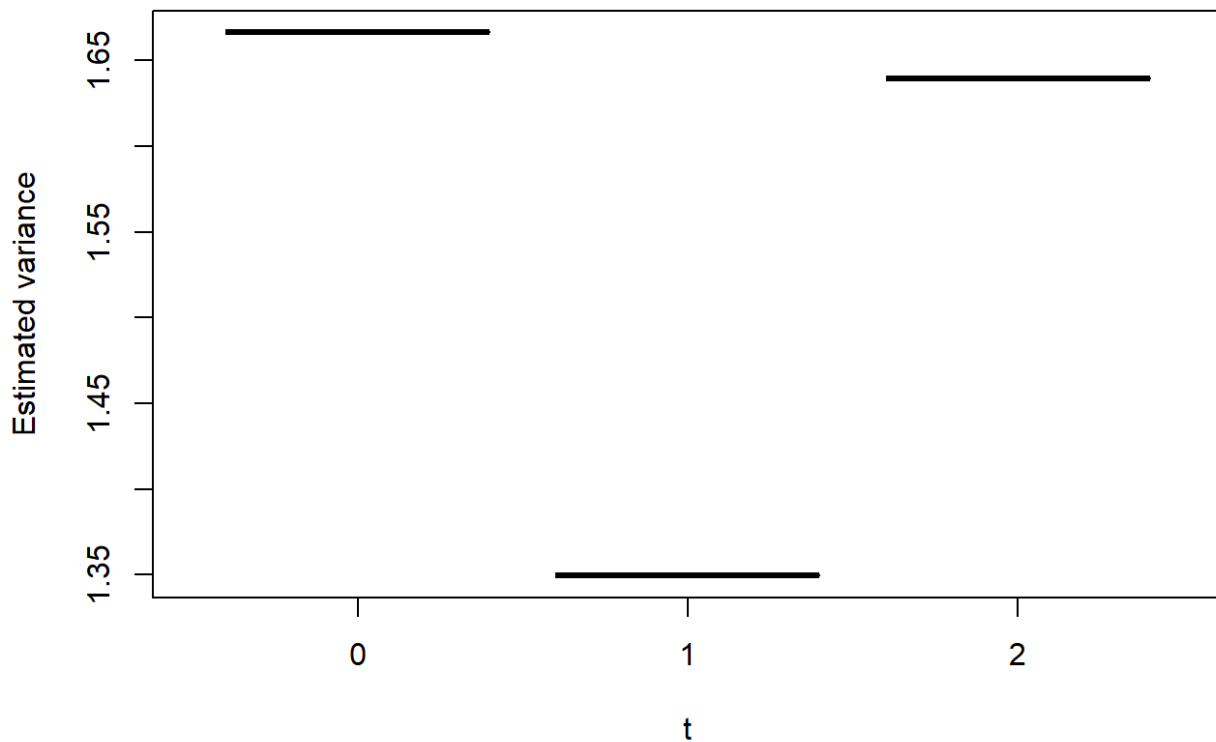
Series residuals(TREM2Model_hetero)



```
plotResiduals(sim_res, db_cc$t)
```

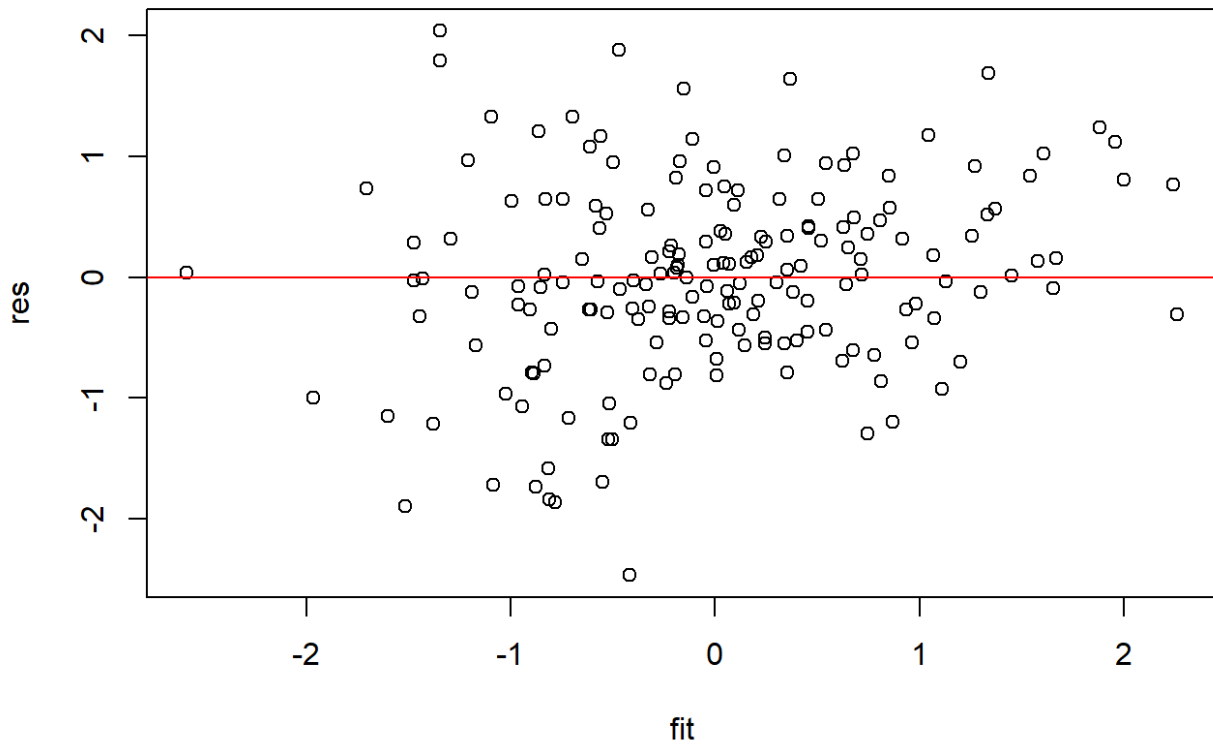


```
pred_disp <- predict(TREM2Model_hetero, type="disp")
plot(db_cc$t, exp(pred_disp),
     xlab="t",
     ylab="Estimated variance")
```



```
res <- residuals(TREM2Model_hetero, type="pearson")
fit <- fitted(TREM2Model_hetero)

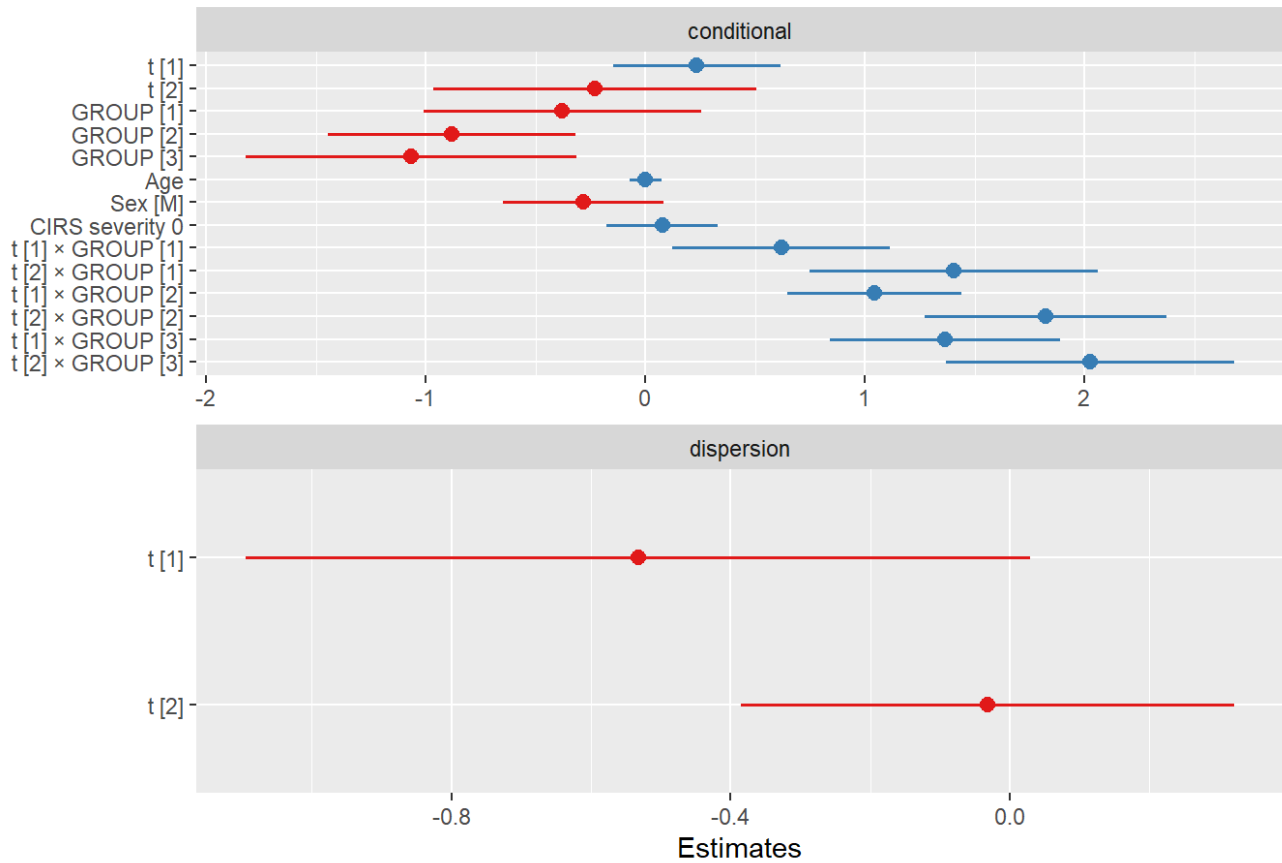
plot(fit, res)
abline(h=0, col="red")
```



19.1 Visualizzazione del modello

```
plot_model(TREM2Model_hetero)
```

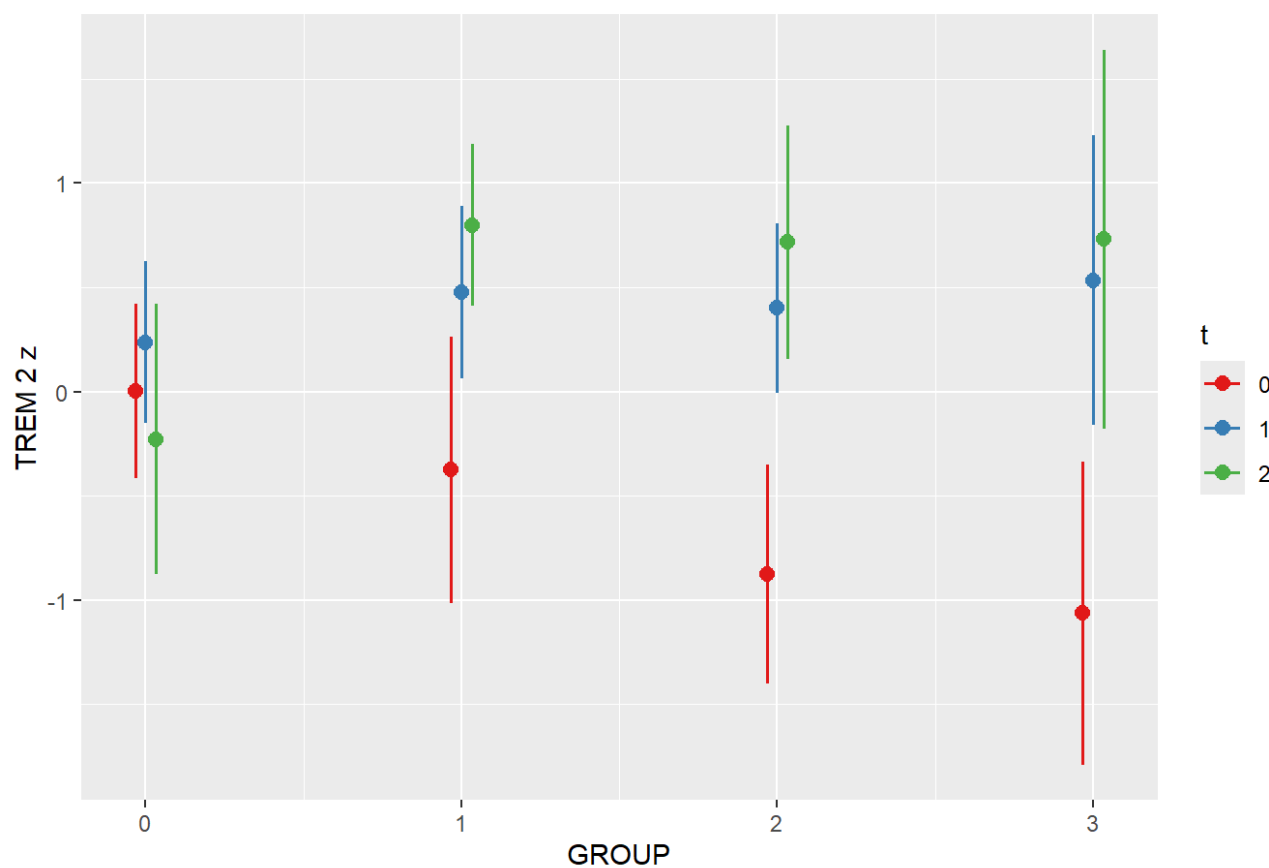
TREM 2 z



```
plot_model(TREM2Model_hetero, type = "pred",
            terms = c("GROUP", "t"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "t"
## • shape : "t"
```

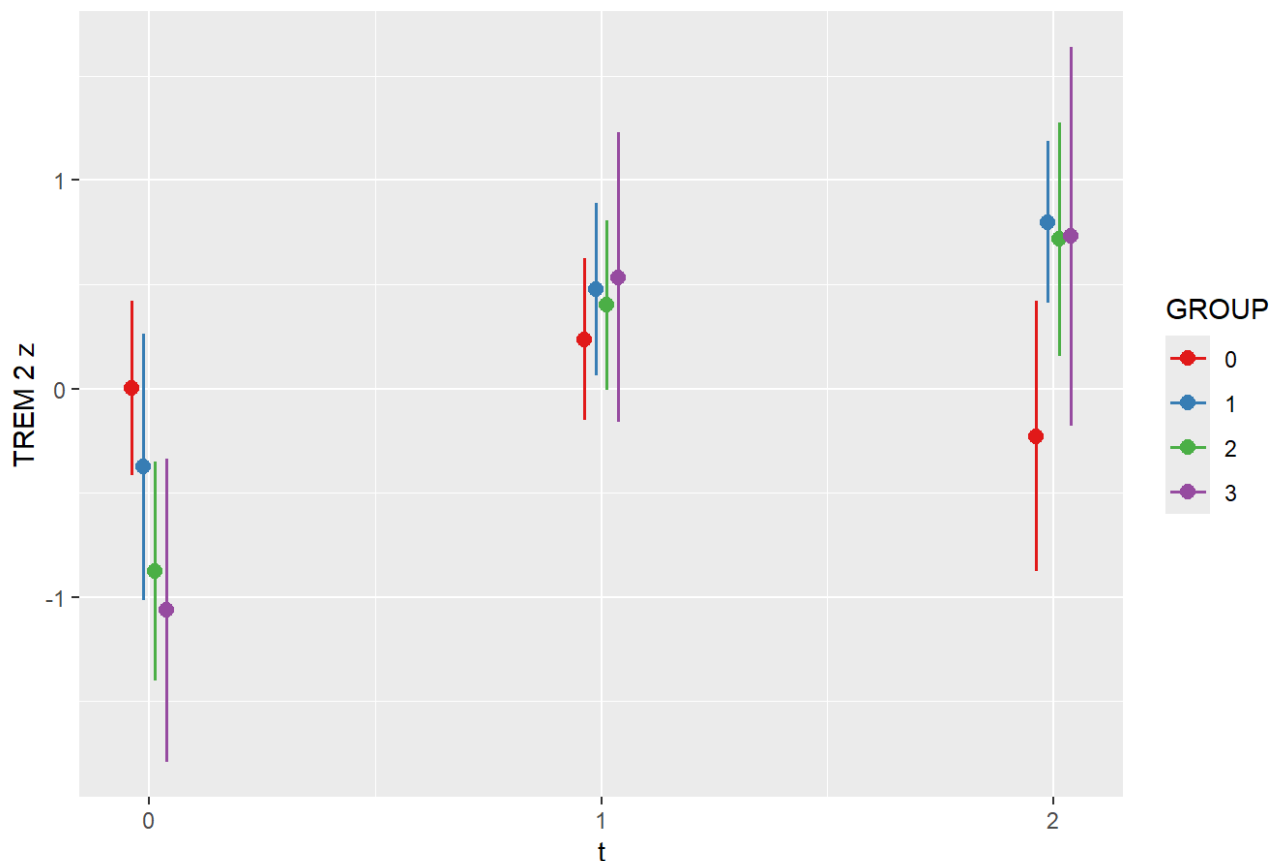
Predicted values of TREM 2 z



```
plot_model(TREM2Model_hetero, type = "pred",
            terms = c("t", "GROUP"),
            pred.type = "fe")
```

```
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of TREM 2 z

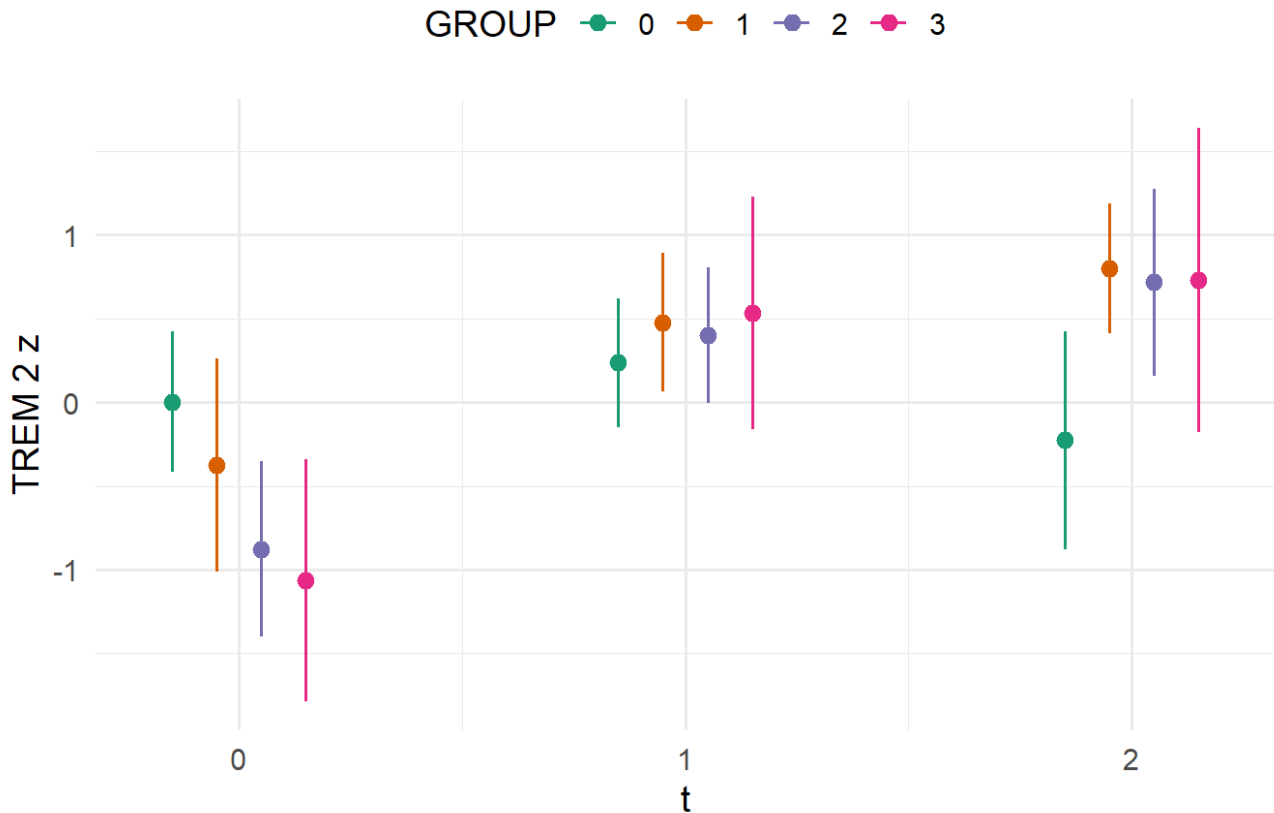


```
plot_model(TREM2Model_hetero,
            type = "pred",
            terms = c("t", "GROUP"),
            dodge = 0.4) +
  scale_color_brewer(palette = "Dark2") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top")
```

```
## Warning in check_dots(..., .action = "warning"): unknown arguments: dodg
e
```

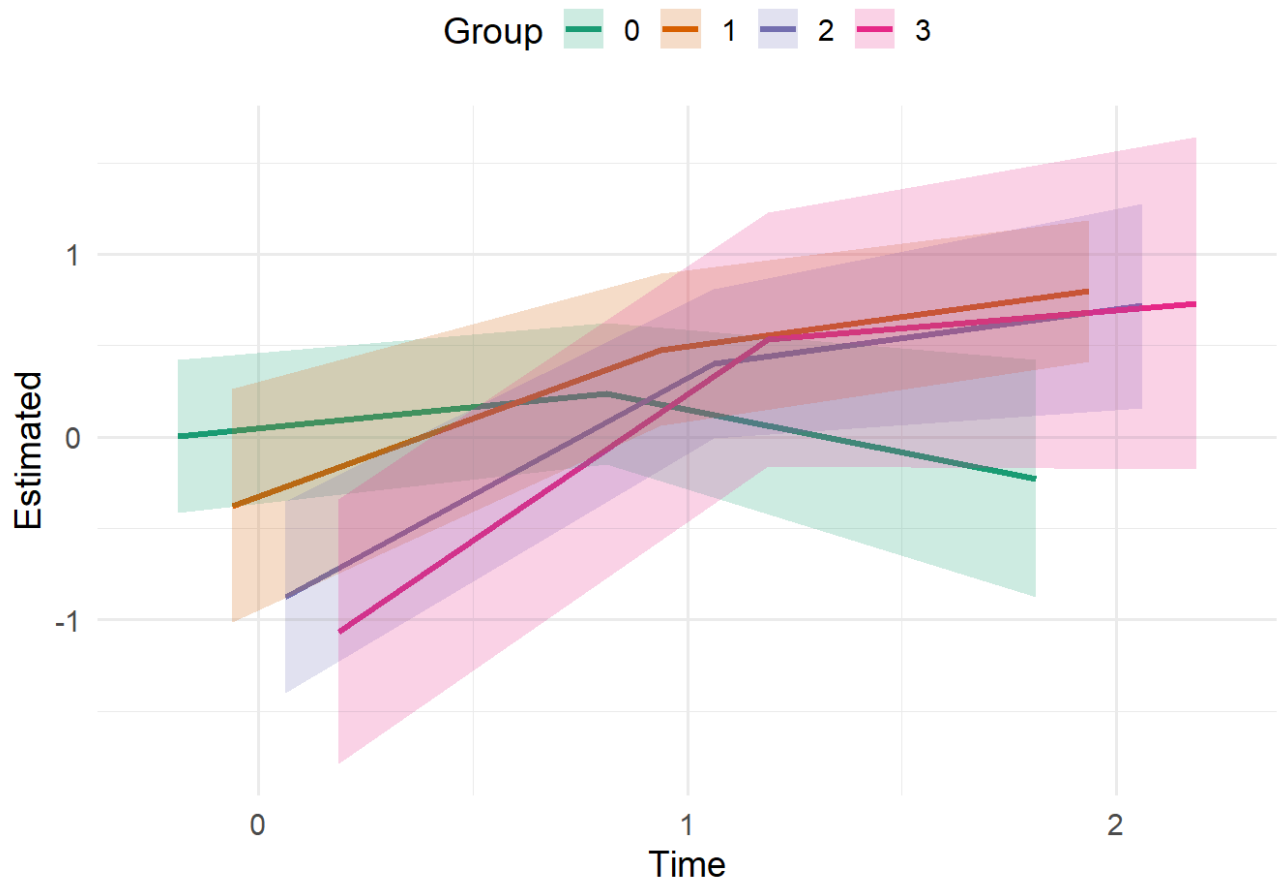
```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
## Ignoring unknown labels:
## • linetype : "GROUP"
## • shape : "GROUP"
```

Predicted values of TREM 2 z



```
# Personalizzazione per distanziare visivamente le linee (separazione per gruppo)
pred <- ggpredict(TREM2Model_hetero, terms = c("t", "GROUP"))

ggplot(pred, aes(x = as.numeric(as.factor(x)), y = predicted, color = group)) +
  geom_line(aes(group = group),
            size = 1.2,
            position = position_dodge(width = 0.5)) +
  geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group),
            alpha = 0.2,
            color = NA,
            position = position_dodge(width = 0.5)) +
  scale_x_continuous(
    breaks = 1:length(unique(pred$x)),
    labels = unique(pred$x)
  ) +
  labs(x = "Time", y = "Estimated", color = "Group", fill = "Group") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "top") +
  scale_color_brewer(palette = "Dark2") +
  scale_fill_brewer(palette = "Dark2")
```

19.2 Confronti Pairwise (post-hoc)

```
emm <- emmeans(TREM2Model_hetero, ~ GROUP | t)
emm
```

```
## t = 0:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0      -0.1378 0.194 Inf   -0.5181    0.243
##  1      -0.5160 0.328 Inf   -1.1589    0.127
##  2      -1.0175 0.266 Inf   -1.5397   -0.495
##  3      -1.2044 0.349 Inf   -1.8877   -0.521
##
## t = 1:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0       0.0957 0.169 Inf   -0.2357    0.427
##  1       0.3362 0.208 Inf   -0.0717    0.744
##  2       0.2608 0.201 Inf   -0.1326    0.654
##  3       0.3944 0.330 Inf   -0.2517    1.041
##
## t = 2:
##  GROUP  emmean    SE  df asymp.LCL asymp.UCL
##  0     -0.3676 0.313 Inf   -0.9808    0.245
##  1       0.6589 0.190 Inf    0.2874    1.030
##  2       0.5763 0.276 Inf    0.0345    1.118
##  3       0.5911 0.444 Inf   -0.2785    1.461
##
## Results are averaged over the levels of: Sex
## Confidence level used: 0.95
```

```
# Pairwise confronti GROUP
emmeans(TREM2Model_hetero, ~ GROUP) %>%
  pairs(adjust = "tukey")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  contrast          estimate    SE  df z.ratio p.value
##  GROUP0 - GROUP1  -0.2963 0.282 Inf   -1.052  0.7187
##  GROUP0 - GROUP2  -0.0764 0.252 Inf   -0.303  0.9904
##  GROUP0 - GROUP3  -0.0636 0.340 Inf   -0.187  0.9977
##  GROUP1 - GROUP2   0.2198 0.275 Inf    0.801  0.8541
##  GROUP1 - GROUP3   0.2327 0.403 Inf    0.577  0.9389
##  GROUP2 - GROUP3   0.0128 0.373 Inf    0.034  1.0000
##
## Results are averaged over the levels of: t, Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti GROUP stratificati per tempo
emmeans(TREM2Model_hetero, ~ GROUP | t) %>%
  pairs(adjust = "tukey")
```

```
## t = 0:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1  0.3782 0.323 Inf   1.171  0.6451
## GROUP0 - GROUP2  0.8797 0.288 Inf   3.057  0.0120
## GROUP0 - GROUP3  1.0666 0.383 Inf   2.782  0.0277
## GROUP1 - GROUP2  0.5015 0.338 Inf   1.484  0.4474
## GROUP1 - GROUP3  0.6884 0.453 Inf   1.521  0.4248
## GROUP2 - GROUP3  0.1868 0.421 Inf   0.443  0.9709
##
## t = 1:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -0.2405 0.293 Inf  -0.822  0.8441
## GROUP0 - GROUP2 -0.1652 0.255 Inf  -0.647  0.9166
## GROUP0 - GROUP3 -0.2987 0.352 Inf  -0.848  0.8314
## GROUP1 - GROUP2  0.0754 0.288 Inf   0.262  0.9937
## GROUP1 - GROUP3 -0.0582 0.419 Inf  -0.139  0.9990
## GROUP2 - GROUP3 -0.1336 0.383 Inf  -0.348  0.9855
##
## t = 2:
## contrast      estimate    SE  df z.ratio p.value
## GROUP0 - GROUP1 -1.0265 0.360 Inf  -2.851  0.0226
## GROUP0 - GROUP2 -0.9439 0.319 Inf  -2.956  0.0165
## GROUP0 - GROUP3 -0.9587 0.405 Inf  -2.367  0.0835
## GROUP1 - GROUP2  0.0826 0.324 Inf   0.255  0.9942
## GROUP1 - GROUP3  0.0678 0.479 Inf   0.142  0.9990
## GROUP2 - GROUP3 -0.0148 0.441 Inf  -0.034  1.0000
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
# Pairwise confronti tempo stratificati per GROUP
emmeans(TREM2Model_hetero, ~ t | GROUP) %>%
  pairs(adjust = "tukey")
```

```

## GROUP = 0:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.233 0.194 Inf  -1.201  0.4526
## t0 - t2       0.230 0.375 Inf   0.613  0.8132
## t1 - t2       0.463 0.269 Inf   1.721  0.1974
##
## GROUP = 1:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -0.852 0.238 Inf  -3.588  0.0010
## t0 - t2      -1.175 0.339 Inf  -3.463  0.0015
## t1 - t2      -0.323 0.193 Inf  -1.675  0.2148
##
## GROUP = 2:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -1.278 0.218 Inf  -5.866 <0.0001
## t0 - t2      -1.594 0.359 Inf  -4.436 <0.0001
## t1 - t2      -0.315 0.231 Inf  -1.365  0.3594
##
## GROUP = 3:
## contrast estimate    SE  df z.ratio p.value
## t0 - t1      -1.599 0.283 Inf  -5.656 <0.0001
## t0 - t2      -1.795 0.438 Inf  -4.100  0.0001
## t1 - t2      -0.197 0.319 Inf  -0.617  0.8108
##
## Results are averaged over the levels of: Sex
## P value adjustment: tukey method for comparing a family of 3 estimates

```