RQ1 Analysis and Visuals

2025-08-04

```
#Load in all packages
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.3.3
```

```
library(nlme)
```

```
## Warning: package 'nlme' was built under R version 4.3.3
```

```
library(lme4)
```

```
## Warning: package 'lme4' was built under R version 4.3.3
```

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

```
## Warning: package 'Matrix' was built under R version 4.3.3
```

```
##
## Attaching package: 'lme4'
```

```
## The following object is masked from 'package:nlme':
##
##     lmList
```

```
library(DAAG)
```

```
## Warning: package 'DAAG' was built under R version 4.3.3
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
##
## Attaching package: 'dplyr'
## The following object is masked from ’package:nlme’:
##
##     collapse

## The following objects are masked from ’package:stats’:
##
##     filter, lag

## The following objects are masked from ’package:base’:
##
##     intersect, setdiff, setequal, union
```

```r
library(MuMIn)
```

```
## Warning: package ’MuMIn’ was built under R version 4.3.3
```

```r
library(car)
```

```
## Warning: package ’car’ was built under R version 4.3.3

## Loading required package: carData

## Warning: package ’carData’ was built under R version 4.3.3

##
## Attaching package: ’car’

## The following object is masked from ’package:dplyr’:
##
##     recode

## The following object is masked from ’package:DAAG’:
##
##     vif
```

```r
library(viridis)
```

```
## Warning: package ’viridis’ was built under R version 4.3.3

## Loading required package: viridisLite
```

```r
library(DHARMa)
```

```
## Warning: package ’DHARMa’ was built under R version 4.3.3

## This is DHARMa 0.4.7. For overview type ’?DHARMa’. For recent changes, type news(package = ’DHARMa’)
```

```r
library(FSA)
```

```
## Warning: package ’FSA’ was built under R version 4.3.3
```

```
## Registered S3 methods overwritten by ’FSA’:
##   method       from
##   confint.boot car
##   hist.boot    car
```

```
## ## FSA v0.9.6. See citation(’FSA’) if used in publication.
## ## Run fishR() for related website and fishR(’IFAR’) for related book.
```

```
##
## Attaching package: ’FSA’
```

```
## The following object is masked from ’package:car’:
##
##     bootCase
```

```r
library(emmeans)
```

```
## Warning: package ’emmeans’ was built under R version 4.3.3
```

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

```r
library(ggpubr)
```

```
## Warning: package ’ggpubr’ was built under R version 4.3.3
```

```r
library(rstatix)
```

```
## Warning: package ’rstatix’ was built under R version 4.3.3
```

```
##
## Attaching package: ’rstatix’
```

```
## The following object is masked from ’package:stats’:
##
##     filter
```

```r
library(glmmTMB)
```

```
## Warning: package ’glmmTMB’ was built under R version 4.3.3
```

```r
library(boot)
```

```
## Warning: package ’boot’ was built under R version 4.3.3

##
## Attaching package: ’boot’

## The following object is masked from ’package:car’:
##
##     logit
```

```r
library(lmPerm)
```

```
## Warning: package ’lmPerm’ was built under R version 4.3.3
```

Import the datasets and convert abundance to copies per gram

```r
#Import the dataset
data <- read.csv(file.path(
  "C:/Users/Joyalea/Documents/UBCO/Thesis",
  "Final_Files",
  "RQ1_AMFAbundance",
  "RQ1_results_cleaned_ddPCR.csv"
))

#Review data import
head(data)
```

```
##   mesocosm_id    species biomass copies_uL copies_20uL copies_rxn     P
## 1           1     Bromus   14.39         0           0          0 0.428
## 2           1    Festuca   31.19         0           0          0 0.189
## 3           1 Gaillardia    6.69         0           0          0 0.232
## 4           1  Taraxacum   24.34         0           0          0 0.316
## 5           2     Bromus   26.19         0           0          0 0.412
## 6           2    Festuca   40.62         0           0          0 0.197
##   seed_weight date_bio_harvested date_roots_havested number_plants
## 1          NA             27-Aug              27-Aug             2
## 2          NA             27-Aug              27-Aug             3
## 3          NA             27-Aug              27-Aug             3
## 4          NA             27-Aug              27-Aug             3
## 5          NA             27-Aug              27-Aug             3
## 6          NA             27-Aug              27-Aug             3
```

```r
str(data)
```

```
## ’data.frame’:    112 obs. of  11 variables:
##  $ mesocosm_id        : int  1 1 1 1 2 2 2 2 3 3 ...
##  $ species            : chr  "Bromus" "Festuca" "Gaillardia" "Taraxacum" ...
##  $ biomass            : num  14.39 31.19 6.69 24.34 26.19 ...
##  $ copies_uL          : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ copies_20uL        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ copies_rxn         : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ P                  : num  0.428 0.189 0.232 0.316 0.412 0.197 0.244 0.299 NA NA ...
##  $ seed_weight        : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ date_bio_harvested : chr  "27-Aug" "27-Aug" "27-Aug" "27-Aug" ...
##  $ date_roots_havested: chr  "27-Aug" "27-Aug" "27-Aug" "27-Aug" ...
##  $ number_plants      : int  2 3 3 3 3 3 3 3 2 3 ...
```

```
tail(data)
```

```
##     mesocosm_id    species biomass copies_uL copies_20uL copies_rxn     P
## 107          27 Gaillardia   13.09      0.56        11.2      12.32    NA
## 108          27  Taraxacum   24.22      0.22         4.4       4.84    NA
## 109          28     Bromus   23.48      0.00         0.0       0.00 0.366
## 110          28    Festuca   72.10      0.00         0.0       0.00 0.149
## 111          28 Gaillardia   12.99      0.50        10.0      11.00 0.210
## 112          28  Taraxacum   18.06      0.00         0.0       0.00 0.243
##     seed_weight date_bio_harvested date_roots_havested number_plants
## 107    0.002060             28-Aug              04-Sep             3
## 108          NA             28-Aug              04-Sep             3
## 109          NA             28-Aug              05-Sep             2
## 110          NA             28-Aug              05-Sep             3
## 111    0.001730             28-Aug              05-Sep             3
## 112    0.000505             28-Aug              05-Sep             3
```

```
#Add treatment group and growing block
data <- data %>%
  mutate(treatment = ifelse(mesocosm_id %in% c(1:7), "Control",
                            ifelse(mesocosm_id %in% c(8:14), "197198",
                                   ifelse(mesocosm_id %in% c(15:21), "240448",
                            ifelse(mesocosm_id %in% c(22:28), "240720", NA)))),
         block = ifelse(mesocosm_id %in% c(15, 20, 22, 12, 11, 14, 24, 27,
                                           19, 21, 25, 3, 2, 7), "A",
                        ifelse(mesocosm_id %in% c(4, 17, 9, 28, 6, 1, 5, 18, 8,
                                                  26, 16, 23, 13, 10), "B", NA)))

View(data)
data$mesocosm_id<-as.character(data$mesocosm_id)

#Add dilution factors of the variance mesocosms
#Read in the file
dilution_df <- read_excel(file.path(
  "C:/Users/Joyalea/Documents/UBCO/Thesis",
  "Final_Files",
  "RQ1_AMFAbundance",
  "Variance_DilutionFactors_M24_M25.xlsx"
))
```

```
## New names:
## * ‘‘ -> ‘...5‘
```

```r
colnames(dilution_df)
```

```
## [1] "Meso"            "mesocosm_id"     "species"         "meso_conc_avg"  
## [5] "...5"            "DNA"             "H2O"             "dilution_factor"
```

```r
#Clean and rename columns
dilution_df <-  dilution_df %>% mutate(
    mesocosm_id = as.character(mesocosm_id),
    species = as.factor(species)
  ) %>%
  select(species, mesocosm_id, dilution_factor)

#Merge dilution df into main dataset and apply dilution correction
data <- data %>%
  left_join(dilution_df, by = c("species", "mesocosm_id"))

#Check the merge was successful
data %>%
  filter(mesocosm_id %in% c("24", "25")) %>%
  select(species, mesocosm_id, dilution_factor)
```

```
##      species mesocosm_id dilution_factor
## 1     Bromus          24        2.512778
## 2    Festuca          24        3.673333
## 3 Gaillardia          24       15.958333
## 4  Taraxacum          24        4.474444
## 5     Bromus          25        3.286667
## 6    Festuca          25        4.028889
## 7 Gaillardia          25       16.669444
## 8  Taraxacum          25        4.661111
```

```r
#Add elution volume (30uL * 9 = 270uL variance mesocosms; 3000uL main analysis)
data <- data %>%
  mutate(elution_volume = ifelse(mesocosm_id %in% c("24", "25"), 270, 3000))

#Add root mass to the dataset
data <- data %>%
  mutate(root_mass = case_when(
    mesocosm_id %in% c("24", "25") & species %in% c("Bromus", "Festuca") ~ 0.9,
    mesocosm_id %in% c("24", "25") & species %in% c("Gaillardia", "Taraxacum") ~ 1.8,
    TRUE ~ 1.5
  ))

#Calculate total copies accounting for elution volume and dilution factor
data <- data %>%
  mutate(
    total_copies = case_when(
      mesocosm_id %in% c("24", "25") ~ copies_rxn * dilution_factor * (elution_volume / 3),
      TRUE ~ copies_rxn * (elution_volume / 3)
    )
  )
```

```r
#Calculate copies_g based on total copies (total copies/root mass extracted from)
data <- data %>%
  mutate(copies_g = total_copies / root_mass)

#Check steps were performed correctly
data %>%
  filter(mesocosm_id %in% c("24", "25")) %>%
  select(species, mesocosm_id, copies_rxn, dilution_factor,
         elution_volume, root_mass, total_copies, copies_g)
```

```
##      species mesocosm_id copies_rxn dilution_factor elution_volume root_mass
## 1     Bromus          24    2450.80        2.512778            270       0.9
## 2    Festuca          24       0.00        3.673333            270       0.9
## 3 Gaillardia          24       7.92       15.958333            270       1.8
## 4  Taraxacum          24      11.22        4.474444            270       1.8
## 5     Bromus          25       5.50        3.286667            270       0.9
## 6    Festuca          25      11.66        4.028889            270       0.9
## 7 Gaillardia          25      19.36       16.669444            270       1.8
## 8  Taraxacum          25      16.28        4.661111            270       1.8
##   total_copies   copies_g
## 1   554248.420 615831.578
## 2        0.000      0.000
## 3    11375.100   6319.500
## 4     4518.294   2510.163
## 5     1626.900   1807.667
## 6     4227.916   4697.684
## 7    29044.840  16136.022
## 8     6829.460   3794.144
```

```r
#looks good

#Convert treatment to a factor and copies per gram to numeric
data$treatment<-as.factor(data$treatment)
#Copies/g is numeric
data$copies_g<-as.numeric(data$copies_g)

#summary of dataset
summary(data$mesocosm_id) #all mesocosms have four observations
```

```
##    Length     Class      Mode
##       112 character character
```

```r
#Create data frames to view species independently
#view species independently
data_gaillardia<-data%>%
  filter(species=="Gaillardia")
data_taraxacum<-data%>%
  filter(species=="Taraxacum")
data_bromus<-data%>%
  filter(species=="Bromus")
data_festuca<-data%>%
  filter(species=="Festuca")
```

```r
#Create data frames to view treatments independently
#197198
data_197198<-data%>%
  filter(treatment == "197198")
#240448
data_240448<-data%>%
  filter(treatment == "240448")
#240720
data_240720<-data%>%
  filter(treatment == "240720")
```

Data visualisation

```r
#Create a new data frame named 'SUBSET' (remove control group)
SUBSET<-data%>%
  filter(treatment != "Control")

#Drop level "control"
SUBSET <- SUBSET %>% mutate(treatment = droplevels(treatment))

#Confirm contorl was removed
str(SUBSET)
```

```
## 'data.frame':    84 obs. of  18 variables:
##  $ mesocosm_id        : chr  "8" "8" "8" "8" ...
##  $ species            : chr  "Bromus" "Festuca" "Gaillardia" "Taraxacum" ...
##  $ biomass            : num  42.2 37.6 10.7 17.6 36 ...
##  $ copies_uL          : num  0.73 0 0.71 7.3 10.6 0.75 0.74 2 0.57 0 ...
##  $ copies_20uL        : num  14.6 0 14.2 146 212 15 14.8 40 11.4 0 ...
##  $ copies_rxn         : num  16.1 0 15.6 160.6 233.2 ...
##  $ P                  : num  0.311 0.194 0.28 0.251 NA NA NA NA NA NA ...
##  $ seed_weight        : num  NA NA NA NA NA NA NA 0.000635 NA NA ...
##  $ date_bio_harvested : chr  "27-Aug" "27-Aug" "27-Aug" "27-Aug" ...
##  $ date_roots_havested: chr  "05-Sep" "05-Sep" "05-Sep" "05-Sep" ...
##  $ number_plants      : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ treatment          : Factor w/ 3 levels "197198","240448",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ block              : chr  "B" "B" "B" "B" ...
##  $ dilution_factor    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ elution_volume     : num  3000 3000 3000 3000 3000 3000 3000 3000 3000 3000 ...
##  $ root_mass          : num  1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 1.5 ...
##  $ total_copies       : num  16060 0 15620 160600 233200 ...
##  $ copies_g           : num  10707 0 10413 107067 155467 ...
```

```r
#Check for outliers
range(SUBSET$copies_g)
```

```
## [1]      0.0 615831.6
```

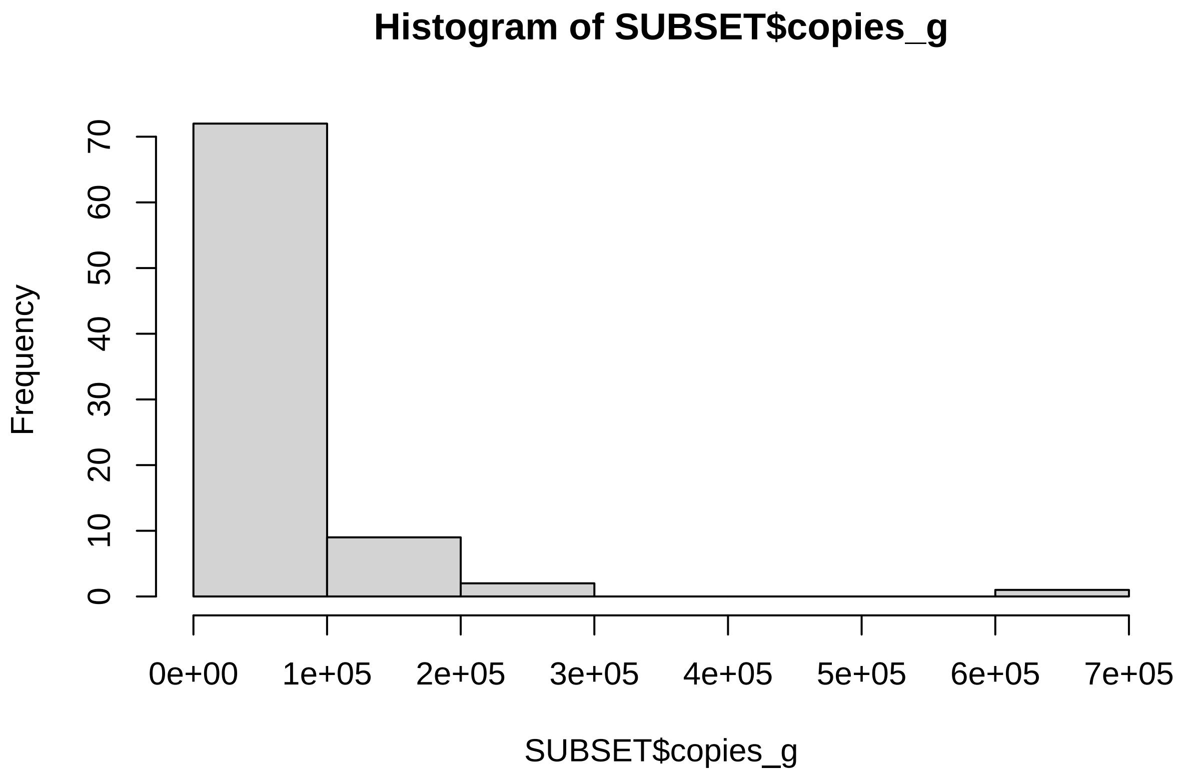
```
median(SUBSET$copies_g)
```

```
## [1] 14668.01
```

```
245080/12100
```

```
## [1] 20.25455
```

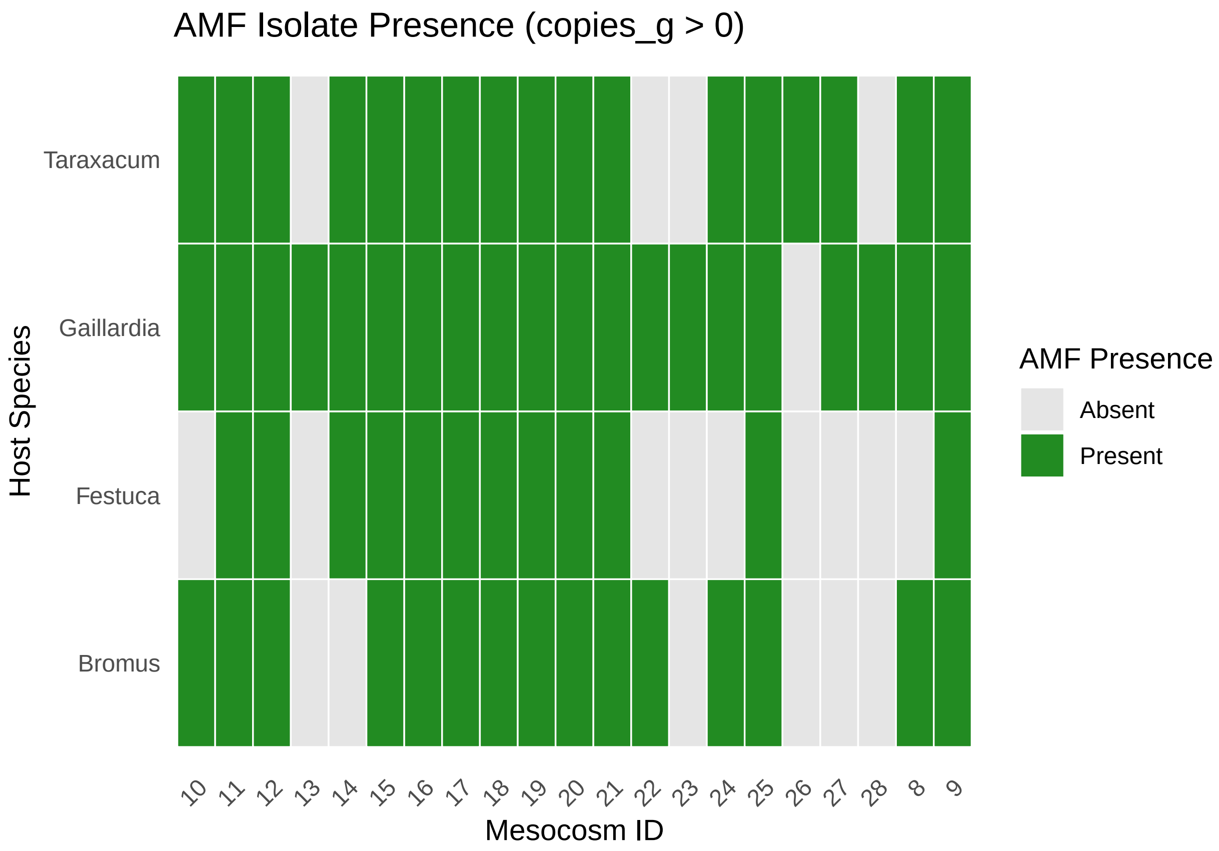
```
#One plant has unusually high abundance (>20X median abundance)
#SUBSET<-SUBSET%>%
 # filter(copies_g<=400) #Remove outlier temporarily
#summary(SUBSET)

#Check histogram shape
hist(SUBSET$copies_g) #Likely zero inflated
```



```
#AMF abundance data (ddPCR) is notoriously zero heavy, check zero cout
#Proportion of zeros
print(SUBSET%>%count(copies_g==0)) #20
```

```
##   copies_g == 0  n
## 1         FALSE 64
## 2          TRUE 20
```

```
SUBSET <- SUBSET %>%
  mutate(presence = ifelse(copies_g > 0, 1, 0))

#Plot presence by mescososm
presence_mesocosm<-ggplot(SUBSET, aes(x = mesocosm_id, y = species, fill = factor(presence))) +
  geom_tile(color = "white", linewidth = 0.3) +
  scale_fill_manual(values = c("0" = "grey90", "1" = "forestgreen"),
                    name = "AMF Presence", labels = c("Absent", "Present")) +
  labs(
    title = "AMF Isolate Presence (copies_g > 0)",
    x = "Mesocosm ID",
    y = "Host Species"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    panel.grid = element_blank()
  )

print(presence_mesocosm)
```

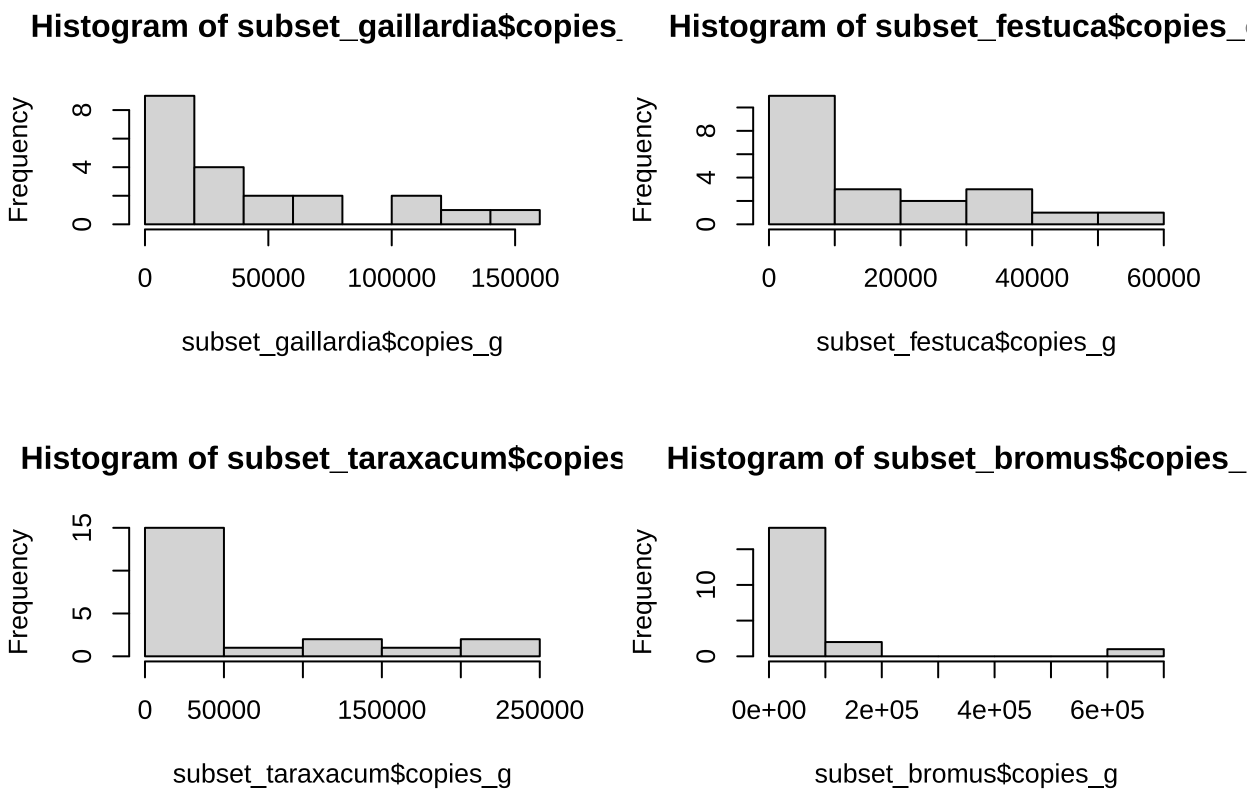


```
#Compare each species independently to see if each group follows the same
#distribution

#view species independently
```

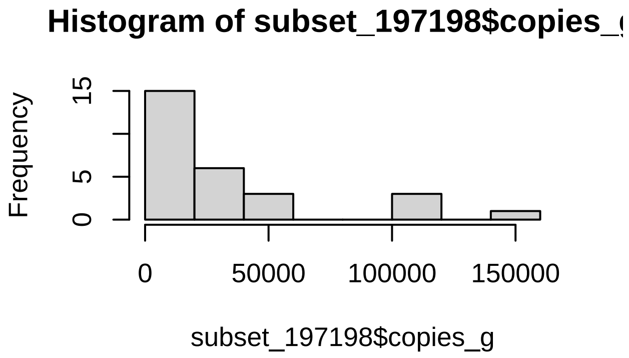
```r
subset_gaillardia<-SUBSET%>%
  filter(species=="Gaillardia")
subset_taraxacum<-SUBSET%>%
  filter(species=="Taraxacum")
subset_bromus<-SUBSET%>%
  filter(species=="Bromus")
subset_festuca<-SUBSET%>%
  filter(species=="Festuca")

#Compare to variance mesocosms
subset_festuca %>%
  filter(mesocosm_id %in% c(24)) %>%
  summarise(mean_copies = mean(copies_g, na.rm = TRUE))
```

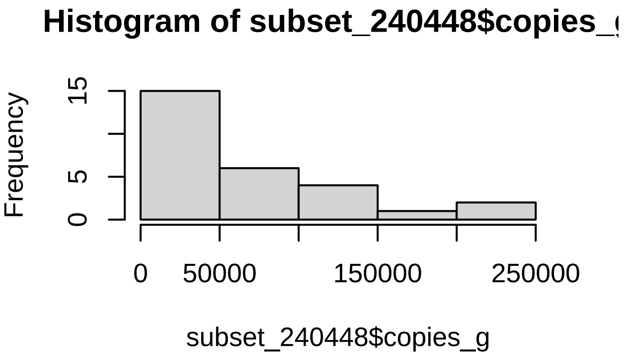
```
##   mean_copies
## 1           0
```

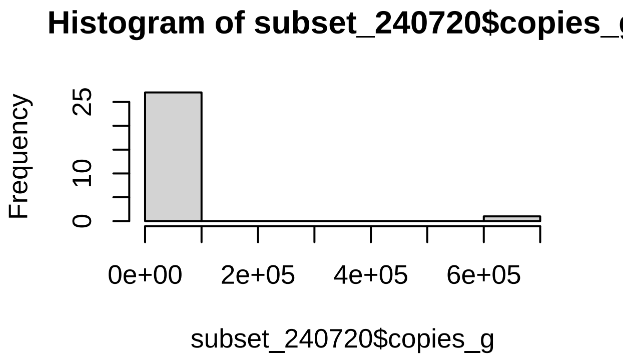
```r
#By treament
#197198
subset_197198<-SUBSET%>%
  filter(treatment == "197198")
#240448
subset_240448<-SUBSET%>%
  filter(treatment == "240448")
#240720
subset_240720<-SUBSET%>%
  filter(treatment == "240720")

par(mfrow = c(2, 2))

#View distirbtuion of gorups
#GAILLARDIA
hist(subset_gaillardia$copies_g)
#FESTUCA
hist(subset_festuca$copies_g) #non linear??
#TARAXACUM
hist(subset_taraxacum$copies_g)
#BROMUS
hist(subset_bromus$copies_g)
```



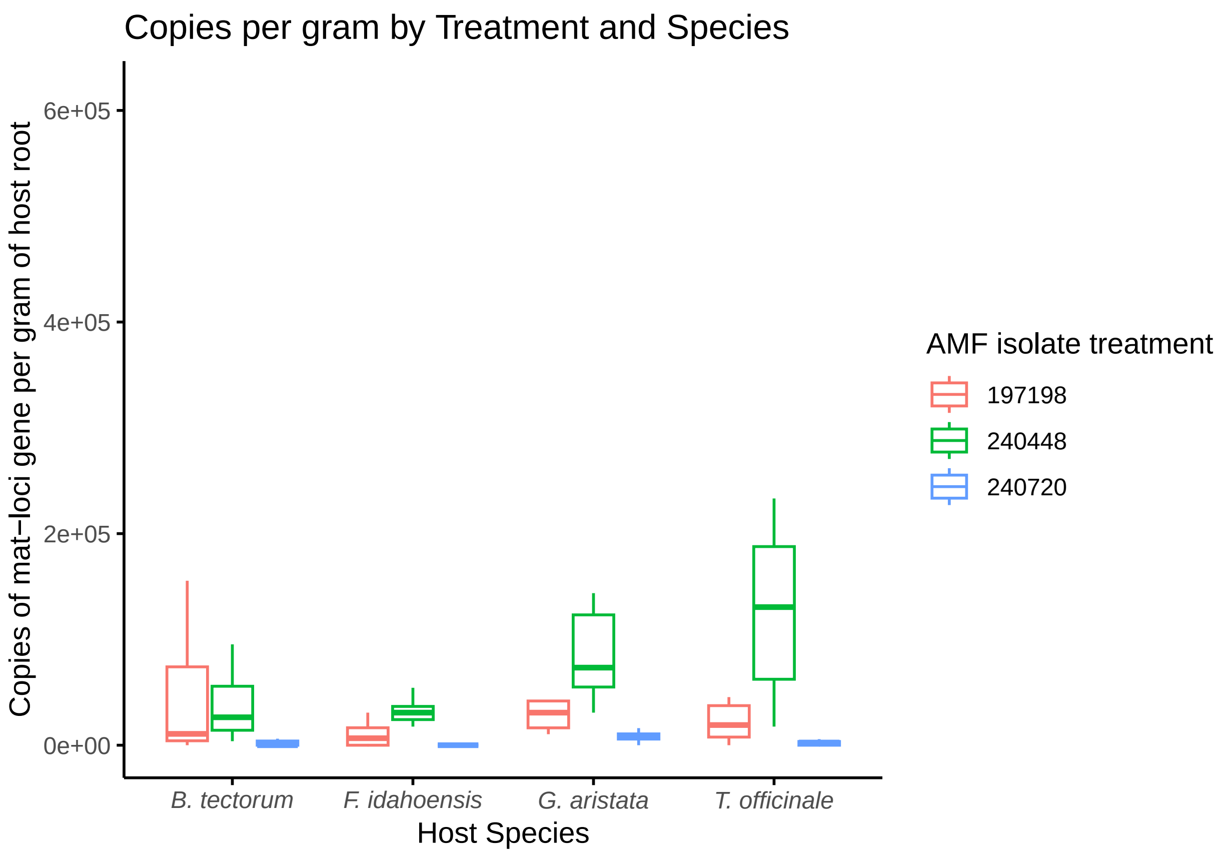
```r
#Roughly similiar distributions

#BY TREATMENT
#197198
hist(subset_197198$copies_g)
#240448
hist(subset_240448$copies_g)
#240720
hist(subset_240720$copies_g)
#all have a right skewed distribution with a spike at small values/zero.
#DAOM240720 spikes at zero and 600,000
par(mfrow = c(1, 1))
```



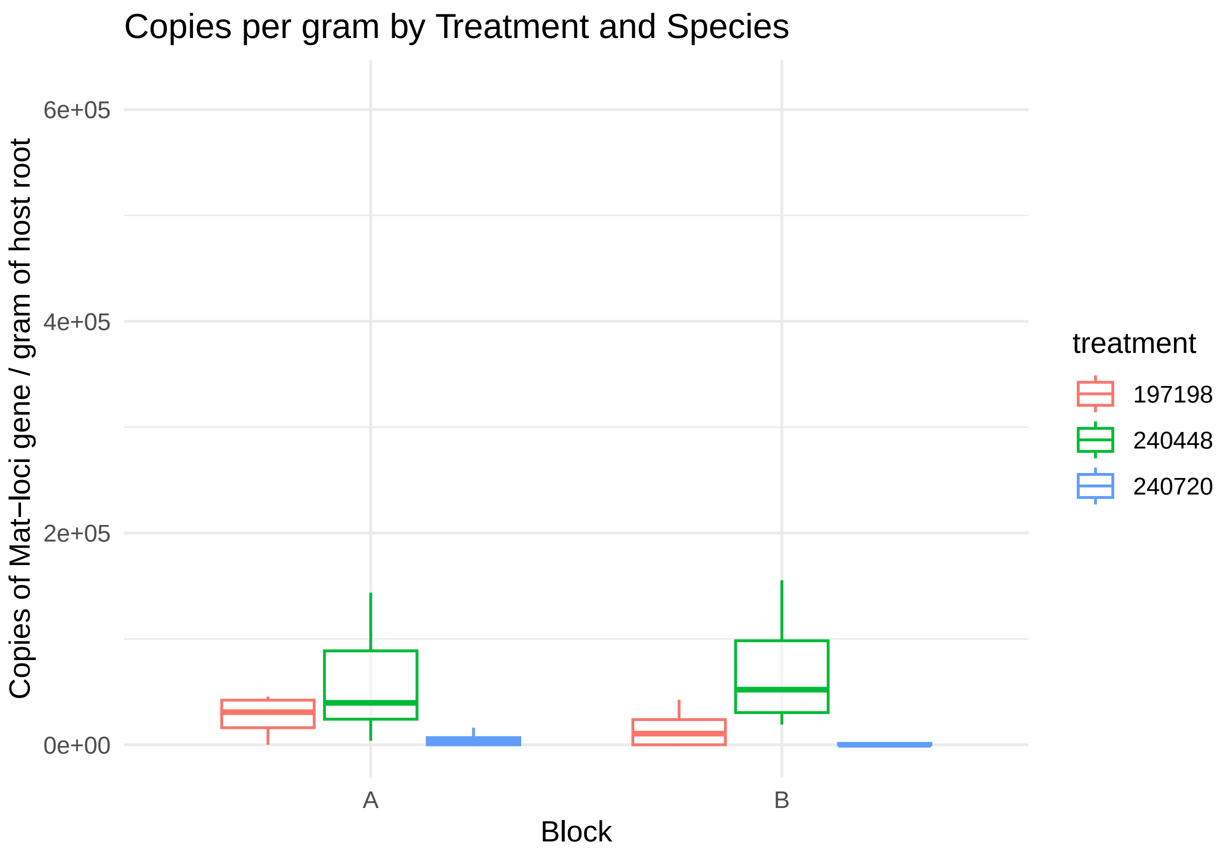




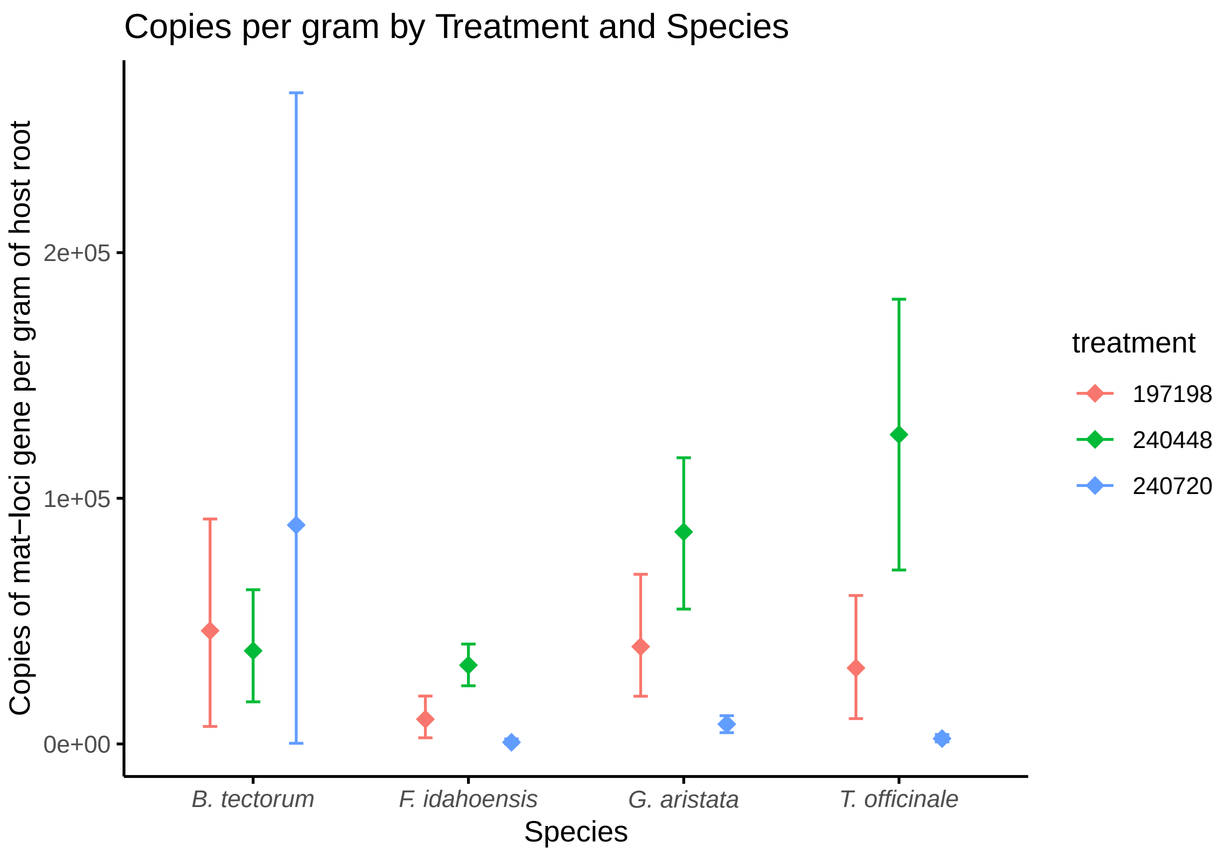
```
#Create species labels for plots
species_labels <- c(
  "Bromus"     = expression(italic("B. tectorum")),
  "Taraxacum"  = expression(italic("T. officinale")),
  "Festuca"    = expression(italic("F. idahoensis")),
  "Gaillardia" = expression(italic("G. aristata"))
)

#Plot copies/g as a response to treatment and species, Line represents median,
#Boxes show the interquartile range -- 25 percentile and 75 percentile of the
#data. Line represents the center of the data
print(plot_copies <- ggplot(SUBSET, aes(x = species, y = copies_g,
                                        color = treatment)) +
        scale_x_discrete(labels = species_labels) +
        geom_boxplot(alpha = 0.5, outlier.shape = NA) +
        labs(
          title = "Copies per gram by Treatment and Species",
          x = "Host Species",
          y = "Copies of mat-loci gene per gram of host root",
          color = "AMF isolate treatment"  # custom legend title
        ) +
        theme_minimal() +
        theme(
          panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          panel.border = element_blank(),
          axis.line = element_line(color = "black"),
          axis.ticks = element_line(color = "black")
        ))
```



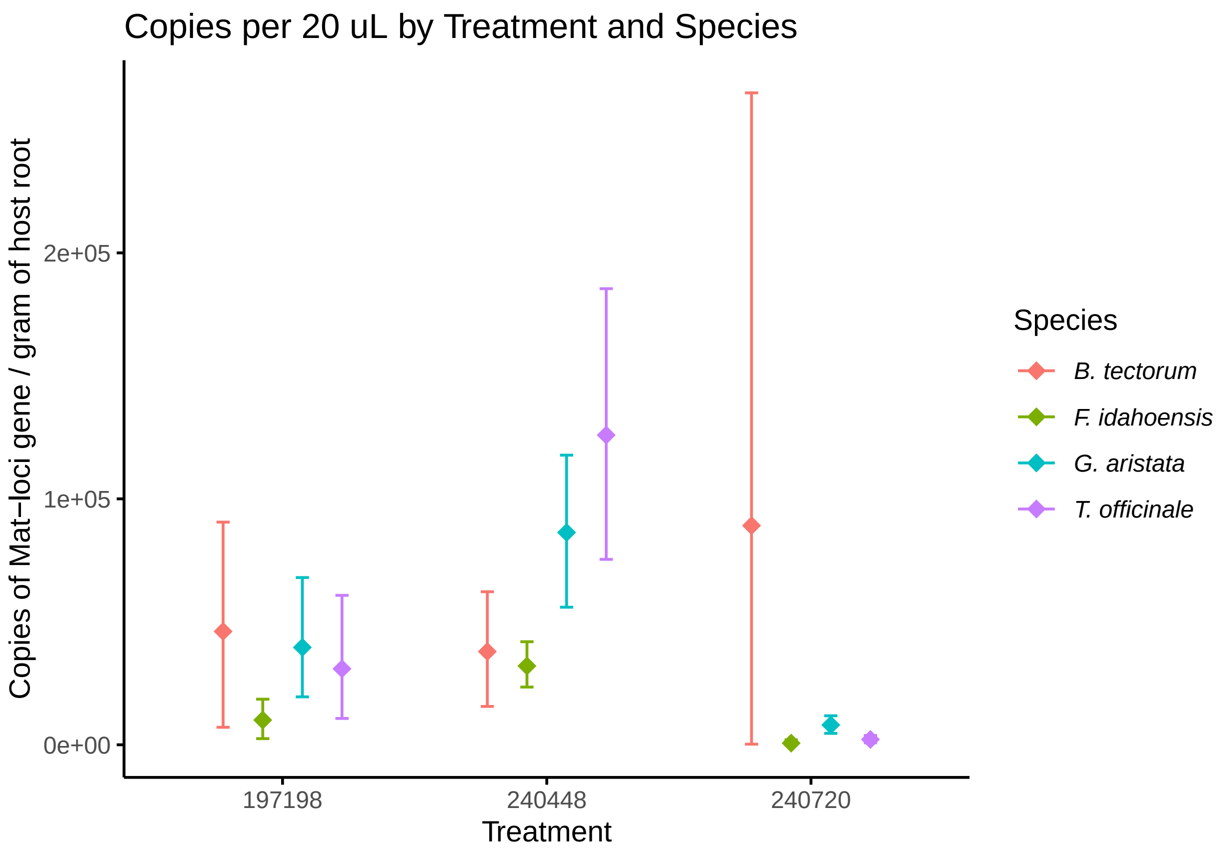
```
#197198 establishment consistent across species HOWEVER variation in 197*brom
#240448 establishment highest in taraxacum and gaillardia, most variation in Tar.
#240720 consistently low across species
#Spike in 240448 in gaillardia and taraxacum
#Visual eterogeneity of variance between treatment and species

#Plot as a response to block
print(plot_copies<-ggplot(SUBSET, aes(x = block, y = copies_g, color = treatment)) +
        #geom_jitter(width = 0.2, alpha = 0.7) + #add individual observations
        geom_boxplot(alpha = 0.5, outlier.shape = NA) +  #boxplot
        labs(title = "Copies per gram by Treatment and Species",
             x = "Block",
             y = "Copies of Mat-loci gene / gram of host root") +
        theme_minimal())
```



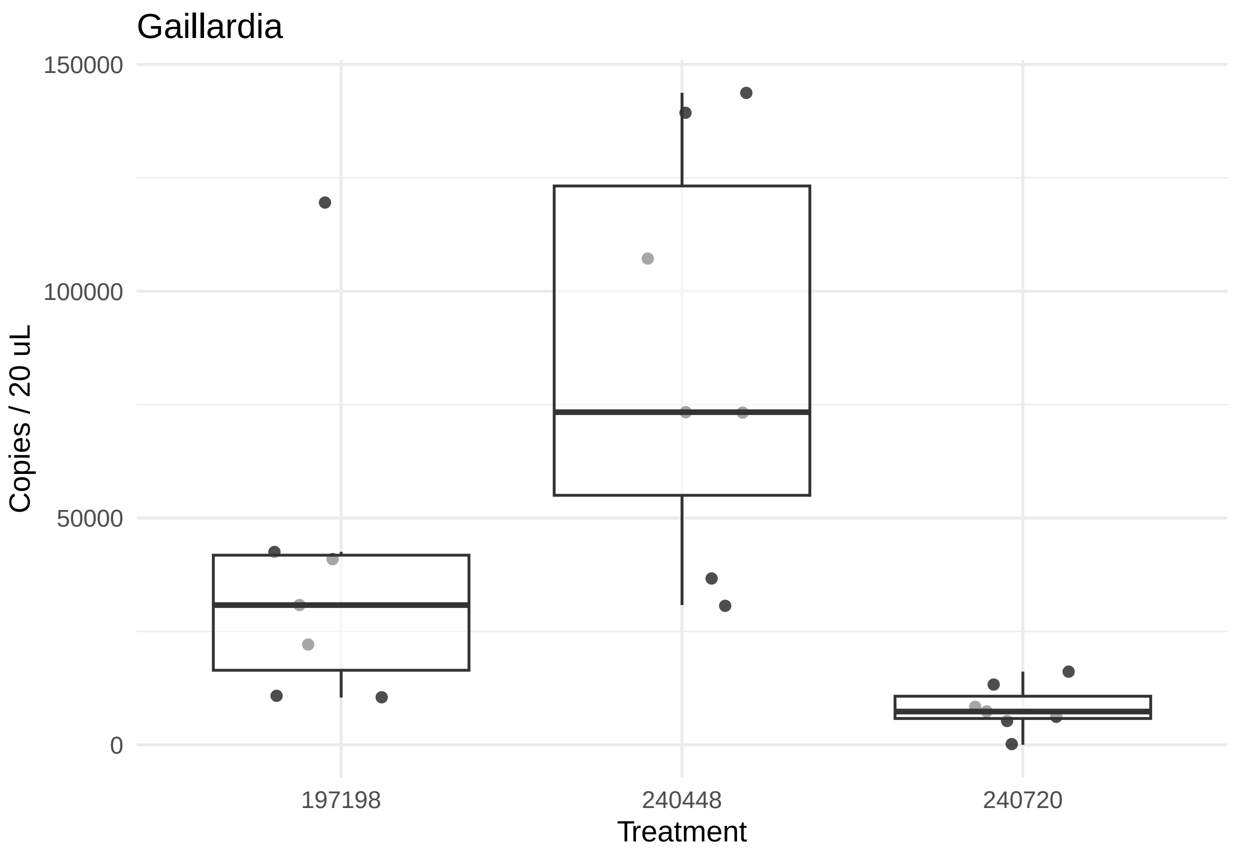
```
#No notable differences

#Mean
print(plot_copies_mean <- ggplot(SUBSET, aes(x = species, y = copies_g,
                                             color = treatment)) +
        stat_summary(fun.data = mean_cl_boot, geom = "errorbar",
                     position = position_dodge(width = 0.6), width = 0.2) +
        stat_summary(fun = mean, geom = "point", shape = 18, size = 3,
                     position = position_dodge(width = 0.6)) +
        scale_x_discrete(labels = species_labels) +
        labs(title = "Copies per gram by Treatment and Species",
             x = "Species",
             y = "Copies of mat-loci gene per gram of host root") +
        theme_minimal() +
        theme(
          panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          panel.border = element_blank(),
          axis.line = element_line(color = "black"),
          axis.ticks = element_line(color = "black")
        ))
```



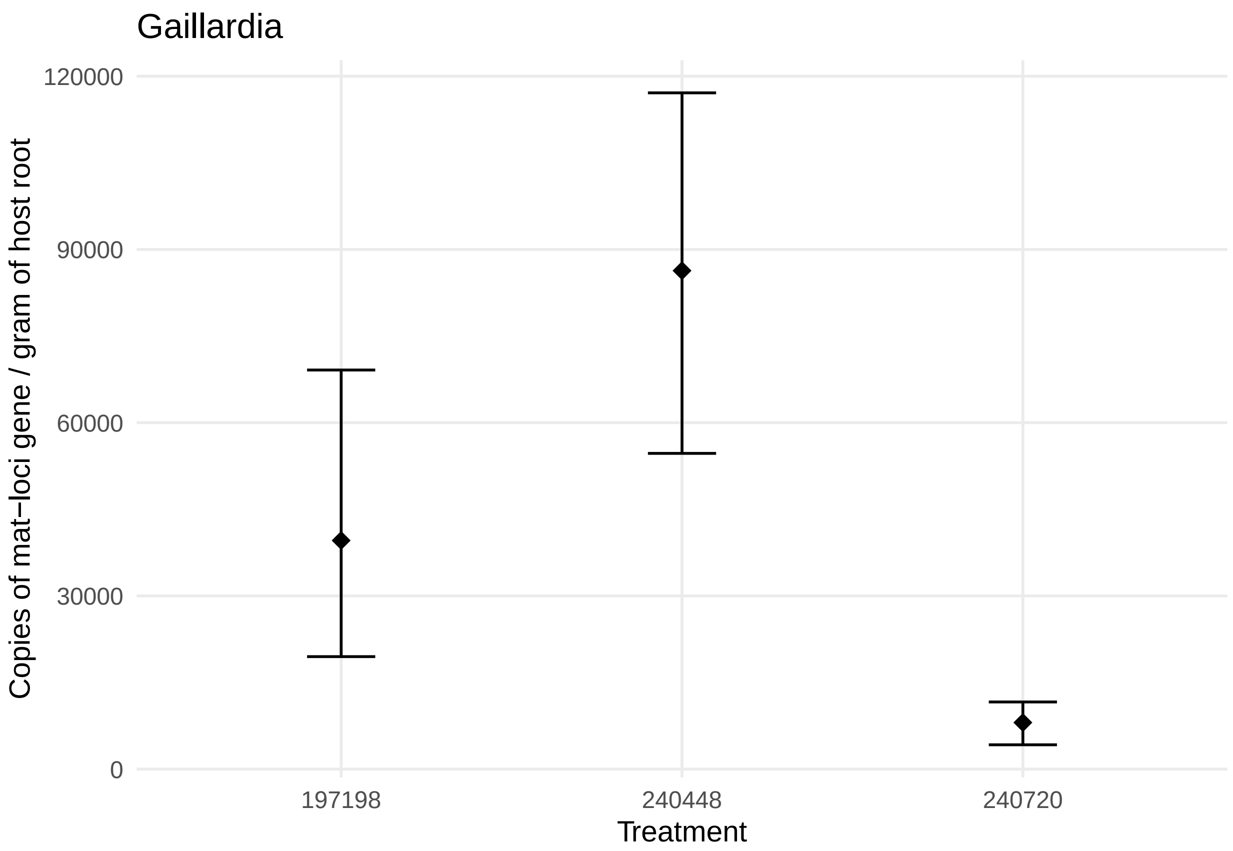
```
#Mean with treatment
print(plot_copies_mean_treatment <- ggplot(SUBSET, aes(x = treatment, y = copies_g,
                                                       color = species)) +
        stat_summary(fun.data = mean_cl_boot, geom = "errorbar",
                     position = position_dodge(width = 0.6), width = 0.2) +
        stat_summary(fun = mean, geom = "point", shape = 18, size = 3,
                     position = position_dodge(width = 0.6)) +
        scale_color_discrete(labels = c(
          "Gaillardia" = expression(italic("G. aristata")),
          "Taraxacum"  = expression(italic("T. officinale")),
          "Bromus"     = expression(italic("B. tectorum")),
          "Festuca"    = expression(italic("F. idahoensis"))
        )) +
        labs(title = "Copies per 20 uL by Treatment and Species",
             x = "Treatment",
             y = "Copies of Mat-loci gene / gram of host root",
             color = "Species") +
        theme_minimal() +
        theme(
          panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          panel.border = element_blank(),
          axis.line = element_line(color = "black"),
          axis.ticks = element_line(color = "black")
        ))
```



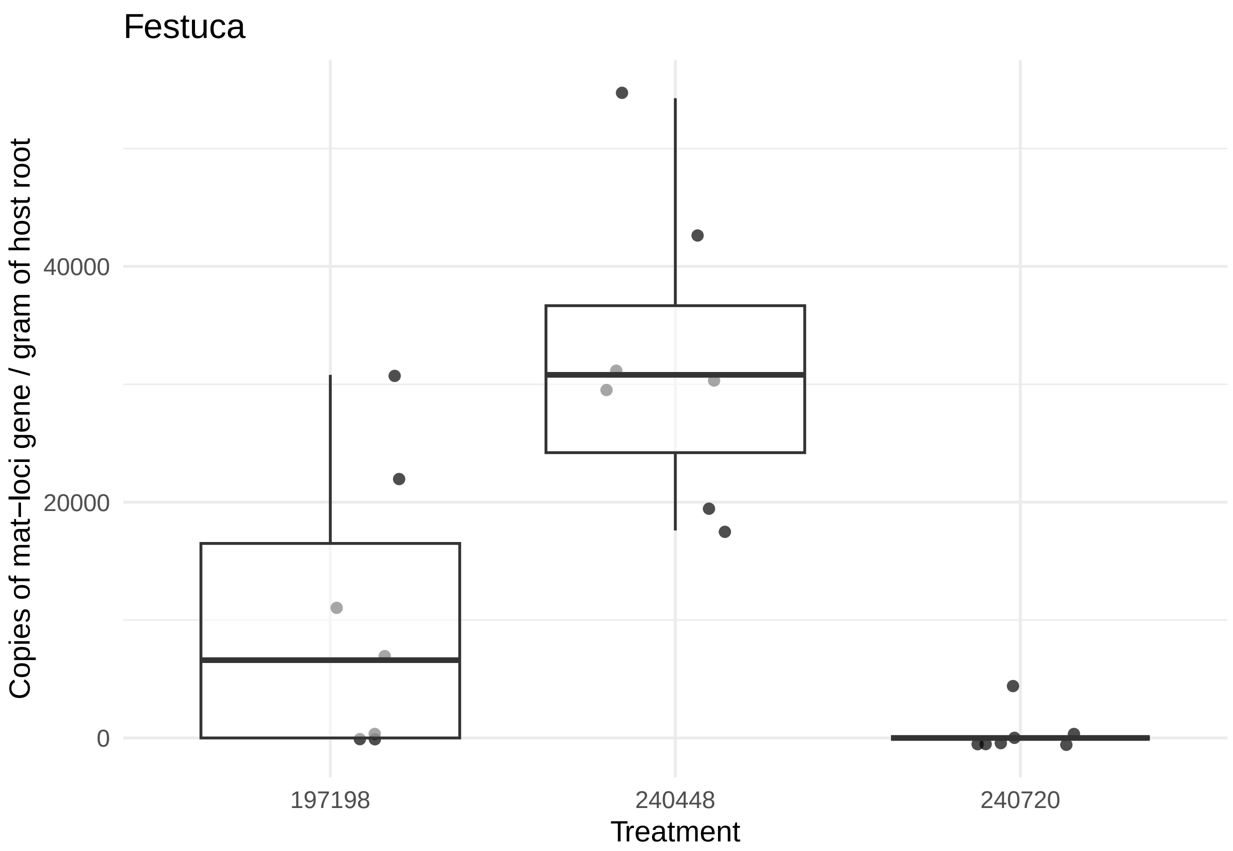
```
#View each species independently
#GAILLARDIA
print(copies_gaillardia<-ggplot(subset_gaillardia, aes(x = treatment, y = copies_g)) +
        geom_jitter(width = 0.2, alpha = 0.7) +  #jitter
        geom_boxplot(alpha = 0.5, outlier.shape = NA) +
        labs(title = "Gaillardia",
             x = "Treatment",
             y = "Copies / 20 uL") +
        theme_minimal())
```



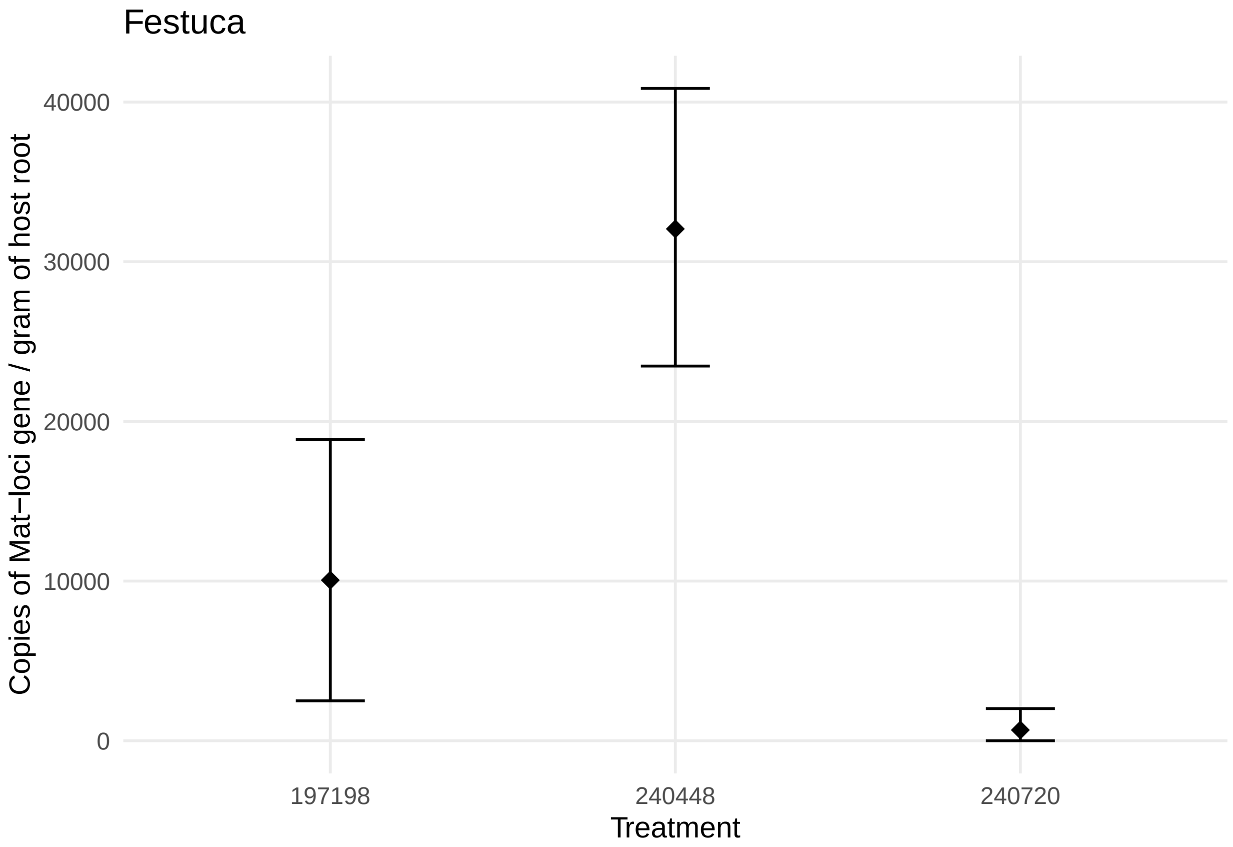
```
#Uptick in establishment in 240448, larger variance spread
#mean
print(copies_gaillardia_mean <- ggplot(subset_gaillardia, aes(x = treatment,
                                                              y = copies_g)) +
        stat_summary(fun.data = mean_cl_boot, geom = "errorbar",
                     position = position_dodge(width = 0.6), width = 0.2) +
        stat_summary(fun = mean, geom = "point", shape = 18, size = 3,
                     position = position_dodge(width = 0.6)) +
        labs(title = "Gaillardia",
             x = "Treatment",
             y = "Copies of mat-loci gene / gram of host root") +
        #scale_y_continuous(limits = c(0, 250000), breaks = seq(0, 250, by = 50)) +
        theme_minimal() +
        theme(panel.grid.minor = element_blank()))
```



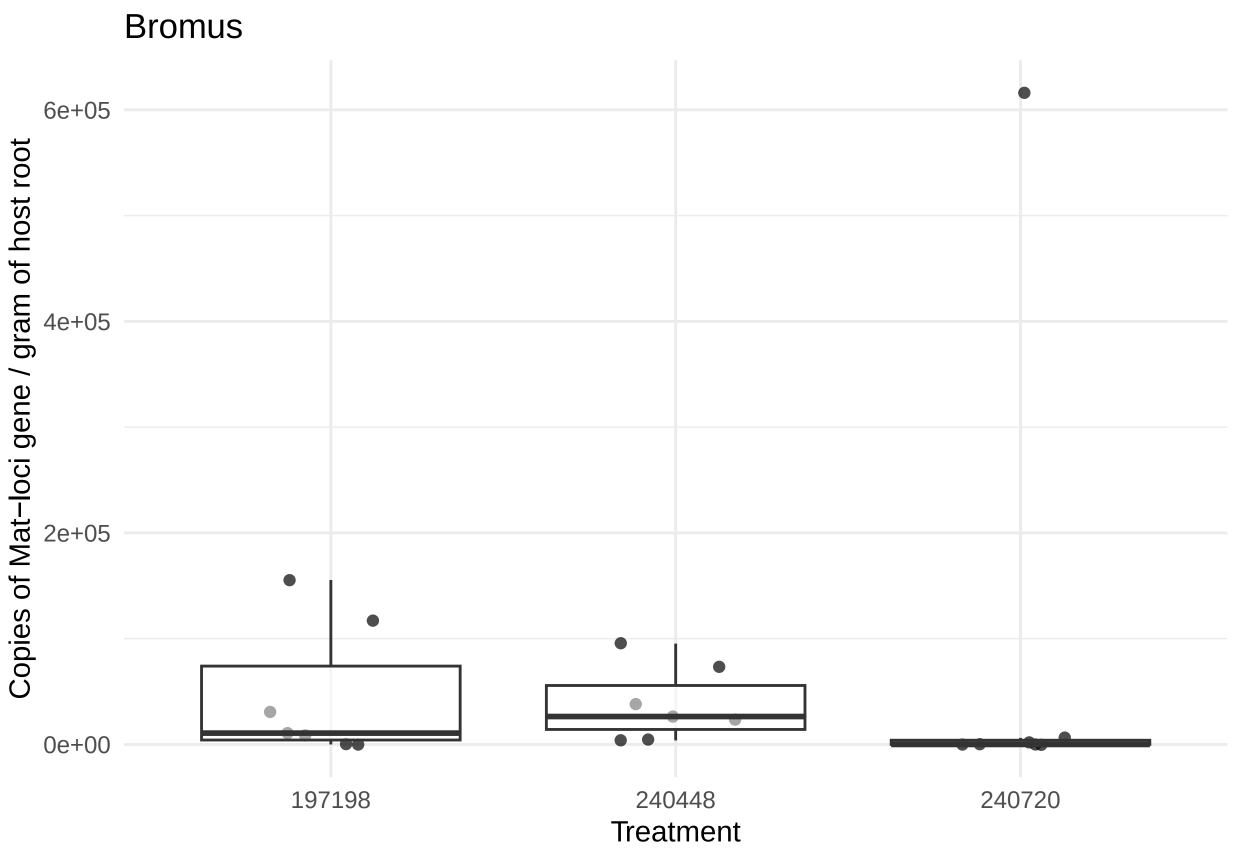
```
#FESTUCA
print(copies_festuca<-ggplot(subset_festuca, aes(x = treatment, y = copies_g)) +
        geom_jitter(width = 0.2, alpha = 0.7) +
        geom_boxplot(alpha = 0.5, outlier.shape = NA) +
        labs(title = "Festuca",
             x = "Treatment",
             y = "Copies of mat-loci gene / gram of host root") +
        theme_minimal())
```



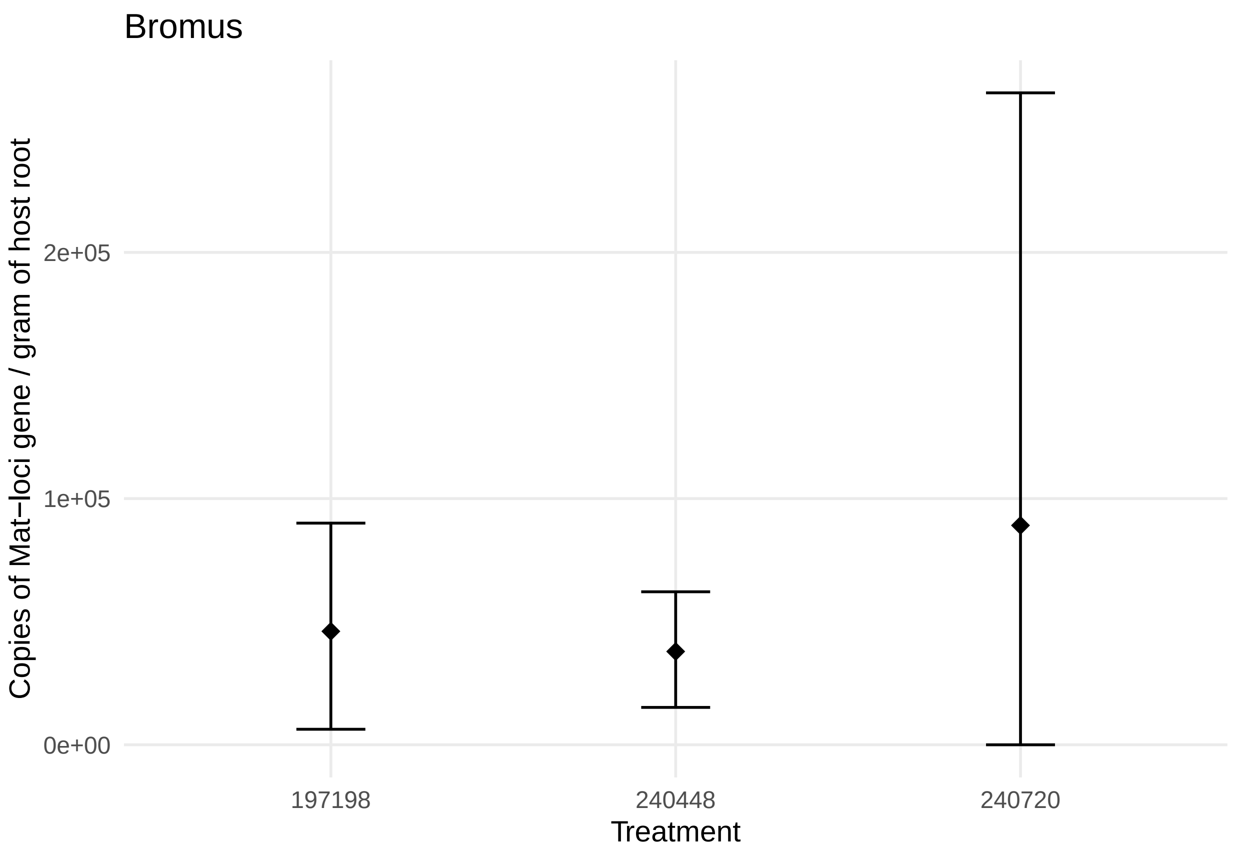
```r
#mean
print(copies_festuca_mean <- ggplot(subset_festuca, aes(x = treatment,
                                                         y = copies_g)) +
        stat_summary(fun.data = mean_cl_boot, geom = "errorbar",
                     position = position_dodge(width = 0.6), width = 0.2) +
        stat_summary(fun = mean, geom = "point", shape = 18, size = 3,
                     position = position_dodge(width = 0.6)) +
        labs(title = "Festuca",
             x = "Treatment",
             y = "Copies of Mat-loci gene / gram of host root") +
    #scale_y_continuous(limits = c(0, 100000), breaks = seq(0, 250, by = 50)) +
        theme_minimal() +
        theme(panel.grid.minor = element_blank()))
```



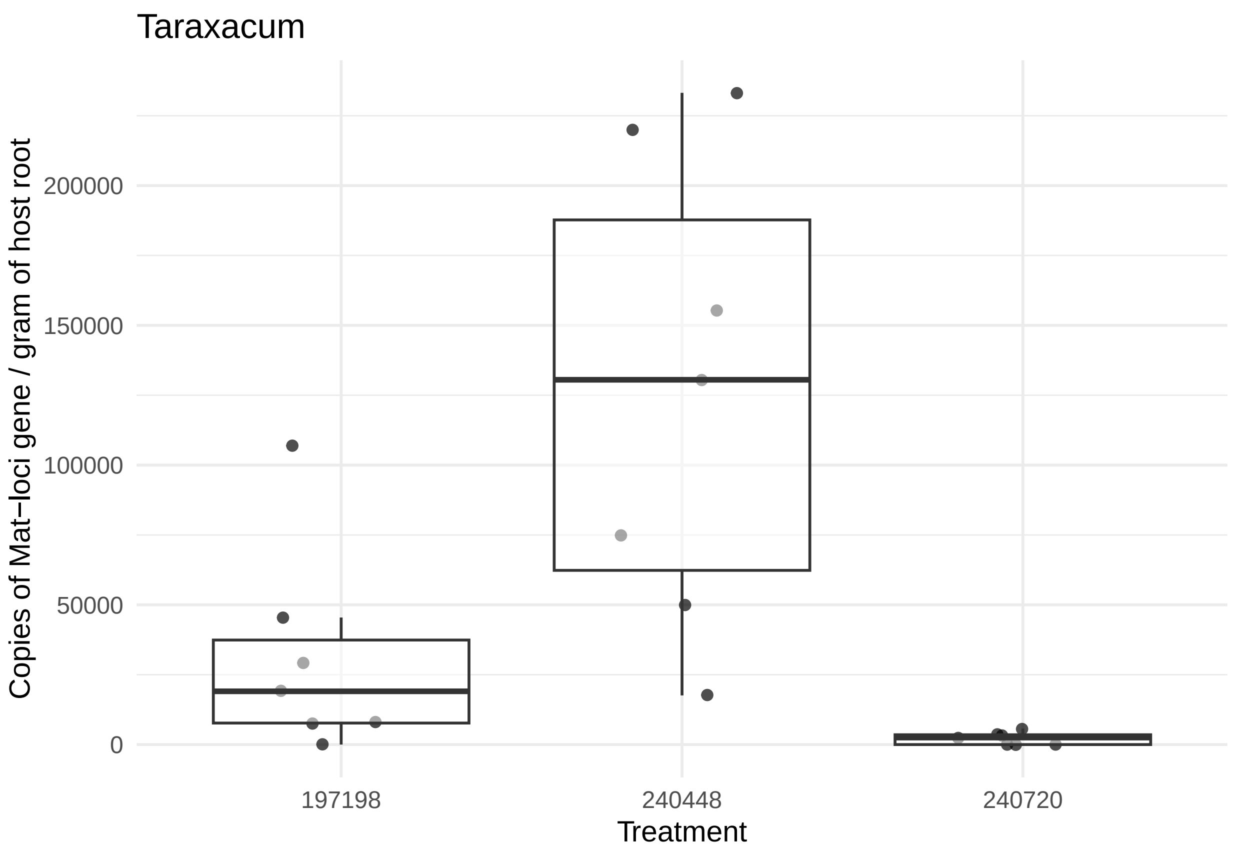
```
#BROMUS
print(copies_bromus<-ggplot(subset_bromus, aes(x = treatment, y = copies_g)) +
        geom_jitter(width = 0.2, alpha = 0.7) +
        geom_boxplot(alpha = 0.5, outlier.shape = NA) +
        labs(title = "Bromus",
             x = "Treatment",
             y = "Copies of Mat-loci gene / gram of host root") +
        theme_minimal())
```



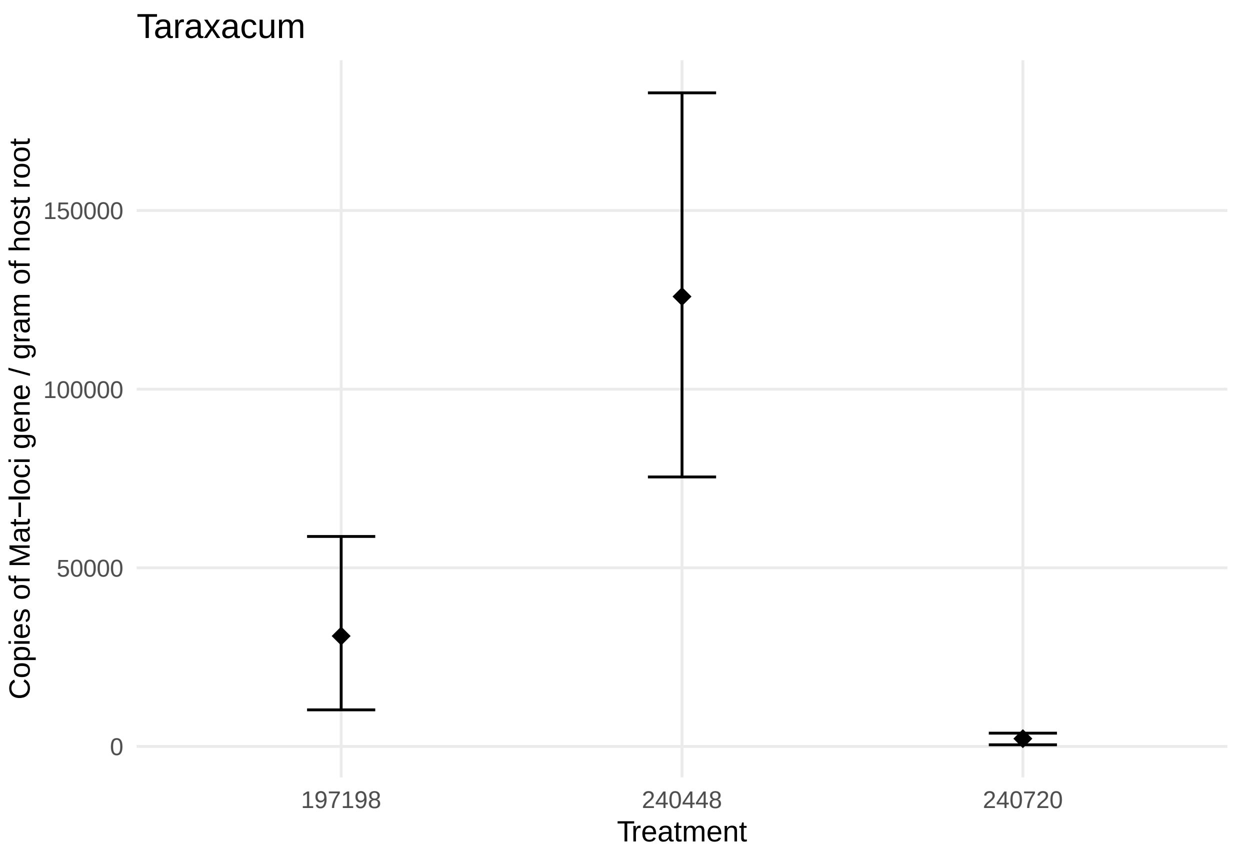
```
#mean
print(copies_bromus_mean <- ggplot(subset_bromus, aes(x = treatment,
                                                       y = copies_g)) +
        stat_summary(fun.data = mean_cl_boot, geom = "errorbar",
                     position = position_dodge(width = 0.6), width = 0.2) +
        stat_summary(fun = mean, geom = "point", shape = 18, size = 3,
                     position = position_dodge(width = 0.6)) +
        labs(title = "Bromus",
             x = "Treatment",
             y = "Copies of Mat-loci gene / gram of host root") +
       # scale_y_continuous(limits = c(0, 200), breaks = seq(0, 250, by = 50)) +
        theme_minimal() +
        theme(panel.grid.minor = element_blank()))
```



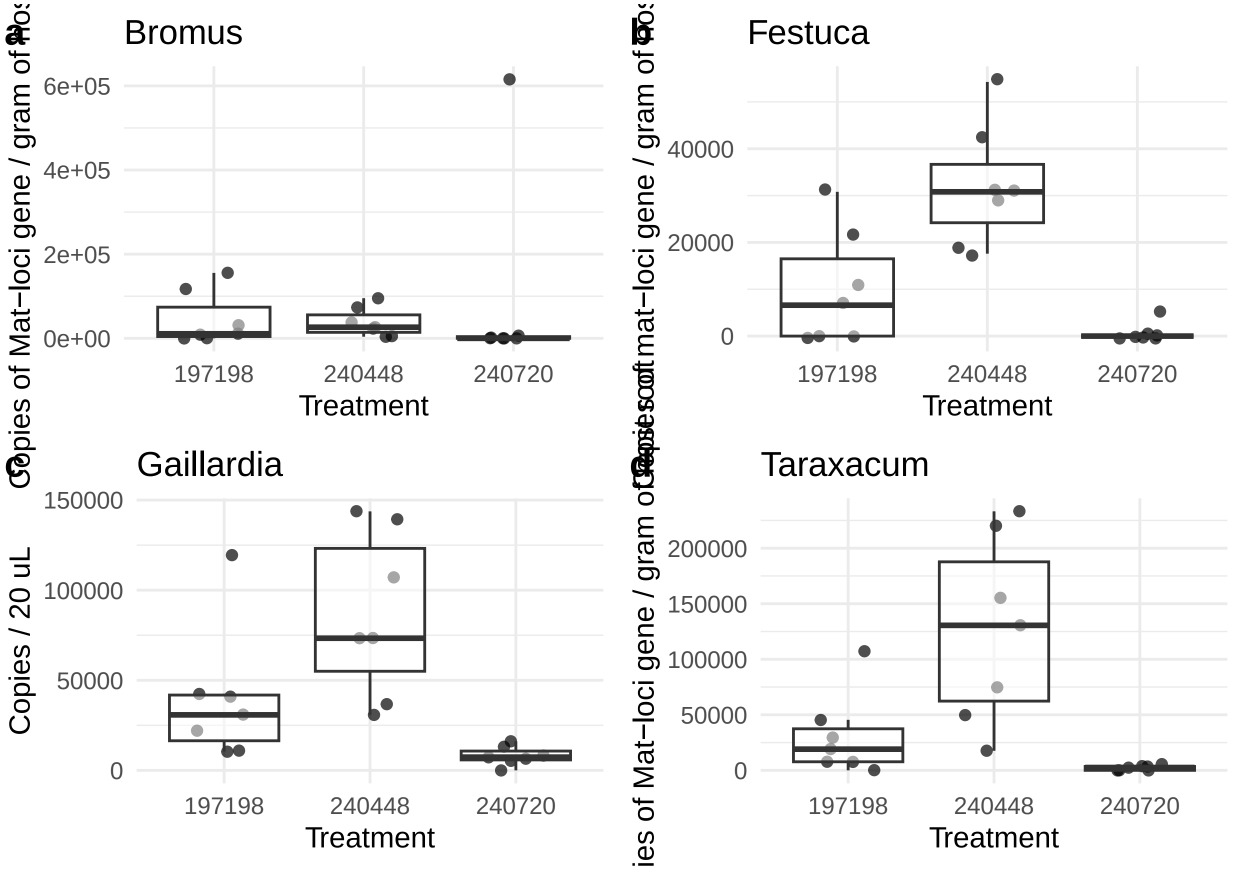
```
#TARAXACUM
print(copies_taraxacum<-ggplot(subset_taraxacum, aes(x = treatment, y = copies_g)) +
        geom_jitter(width = 0.2, alpha = 0.7) +
        geom_boxplot(alpha = 0.5, outlier.shape = NA) +
        labs(title = "Taraxacum",
             x = "Treatment",
             y = "Copies of Mat-loci gene / gram of host root") +
        theme_minimal())
```



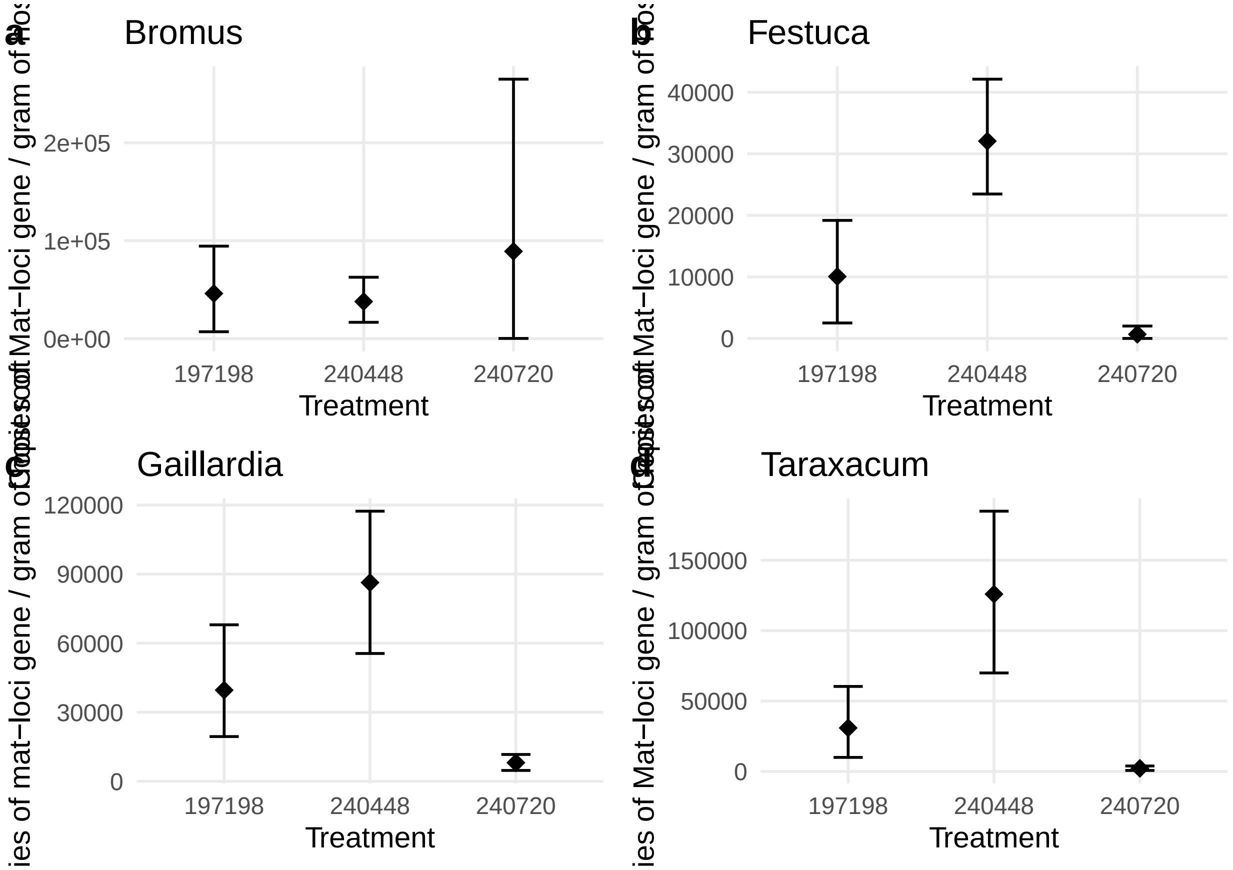
```
#mean
print(copies_taraxacum_mean <- ggplot(subset_taraxacum, aes(x = treatment,
                                                            y = copies_g)) +
        stat_summary(fun.data = mean_cl_boot, geom = "errorbar",
                     position = position_dodge(width = 0.6), width = 0.2) +
        stat_summary(fun = mean, geom = "point", shape = 18, size = 3,
                     position = position_dodge(width = 0.6)) +
        labs(title = "Taraxacum",
             x = "Treatment",
             y = "Copies of Mat-loci gene / gram of host root") +
       # scale_y_continuous(limits = c(0, 250), breaks = seq(0, 250, by = 50)) +
        theme_minimal() +
        theme(panel.grid.minor = element_blank()))
```



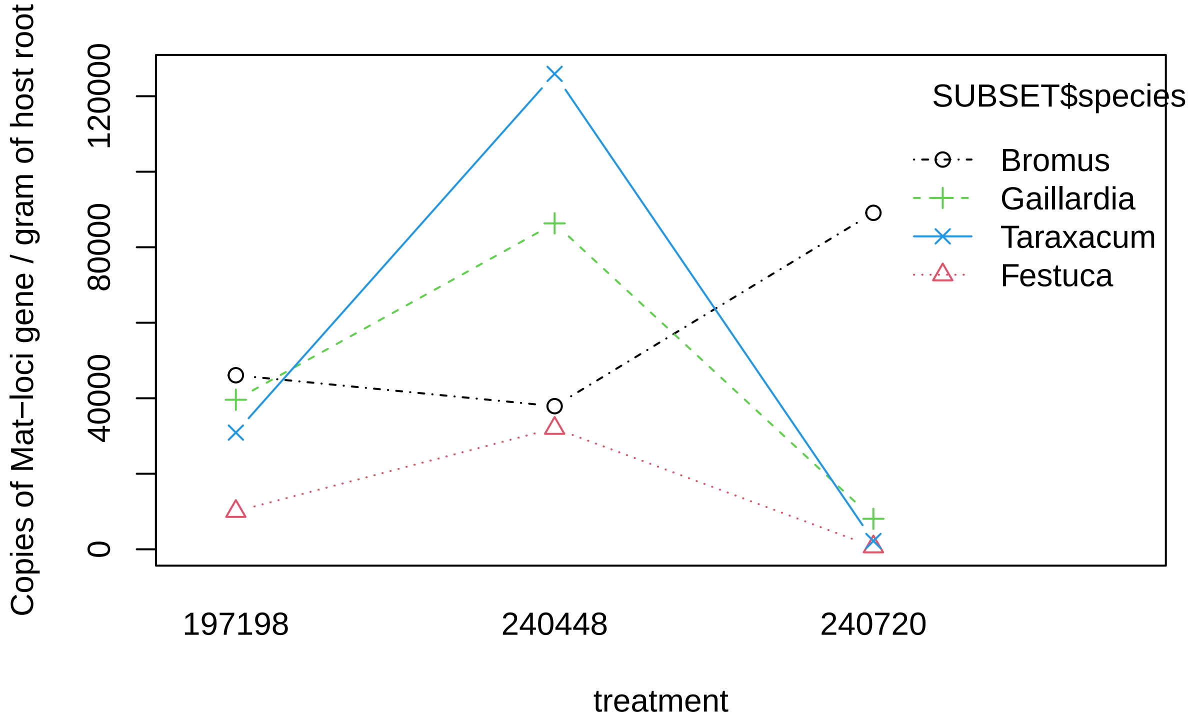
```
#Arrange median plots together
#Arange plots together in a single figure
multi_figure_copies <- ggarrange(copies_bromus, copies_festuca,
                                 copies_gaillardia, copies_taraxacum,
                                 ncol = 2, nrow = 2,
                                 labels = c("a", "b", "c", "d"))

#Display the combined figure. NOTE DIFFERENT Y AXIS VALUES!
print(multi_figure_copies)
```



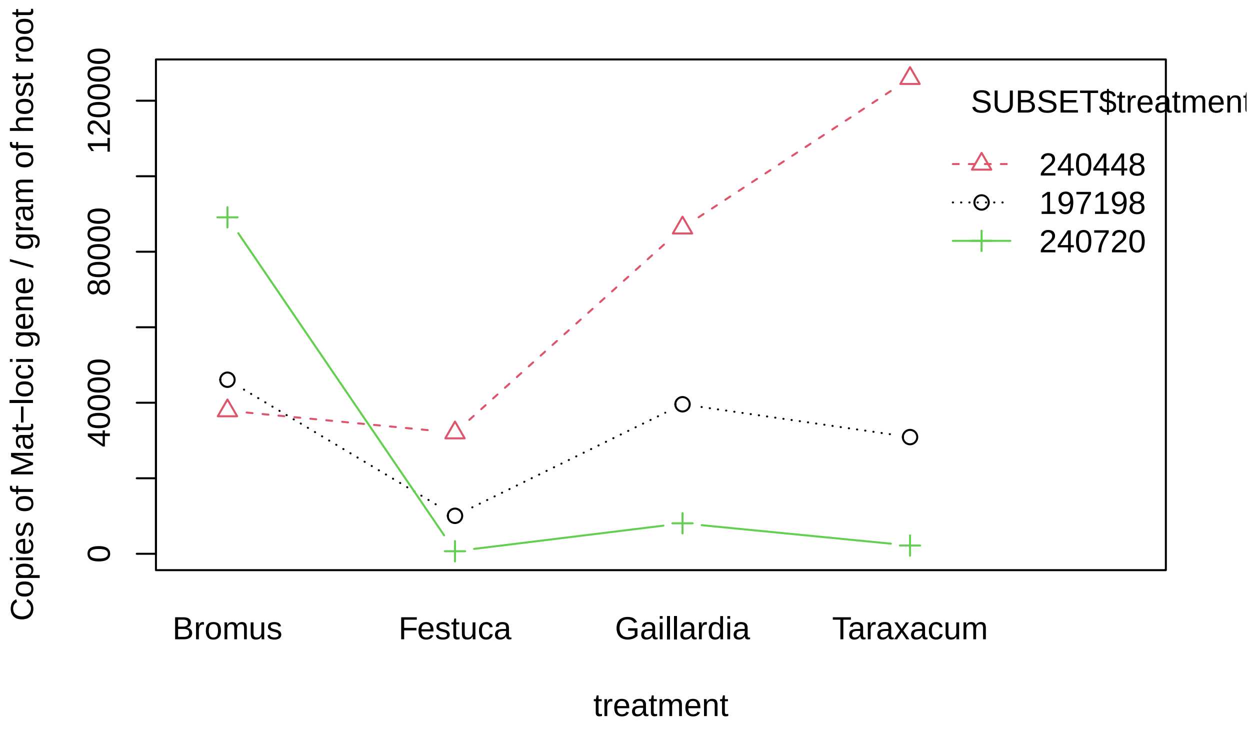
```
#240448 is consistently high across all host species, excepting bromus 197198

#Arange mean plots together in a single figure
multi_copies_mean <- ggarrange(copies_bromus_mean, copies_festuca_mean,
                               copies_gaillardia_mean, copies_taraxacum_mean,
                               ncol = 2, nrow = 2,
                               labels = c("a", "b", "c", "d"))
print(multi_copies_mean)
```



```
#Create interaction plots
#Plot inoculation*hostID
interaction.plot(SUBSET$treatment, SUBSET$species, SUBSET$copies_g,
                 xlab= "treatment",
                 ylab= "Copies of Mat-loci gene / gram of host root",
                 legend= TRUE,
                 col= 1:length(unique(SUBSET$species)),
                 pch= 1:length(unique(SUBSET$species)),
                 type= "b")
```



```
#The slope is a similiar shape acroos treatment*species, excepting 197198 whereas
#abundance does not increase in 240448 relative to 197198. This suggests a
#197198*bromus interaction that changes the direction of effect.
#240448 abundance appears to increase most in taraxacum (steepest slope),
#suggesting a possible interaction between Taraxacum*240448.

#Flip plot
interaction.plot(SUBSET$species, SUBSET$treatment, SUBSET$copies_g,
                 xlab= "treatment",
                 ylab= "Copies of Mat-loci gene / gram of host root",
                 legend= TRUE,
                 col= 1:length(unique(data$species)),
                 pch= 1:length(unique(data$species)),
                 type= "b")
```



```
#240720 stays consistently low, 240448 spikes in gaillardia and taraxacum.
```

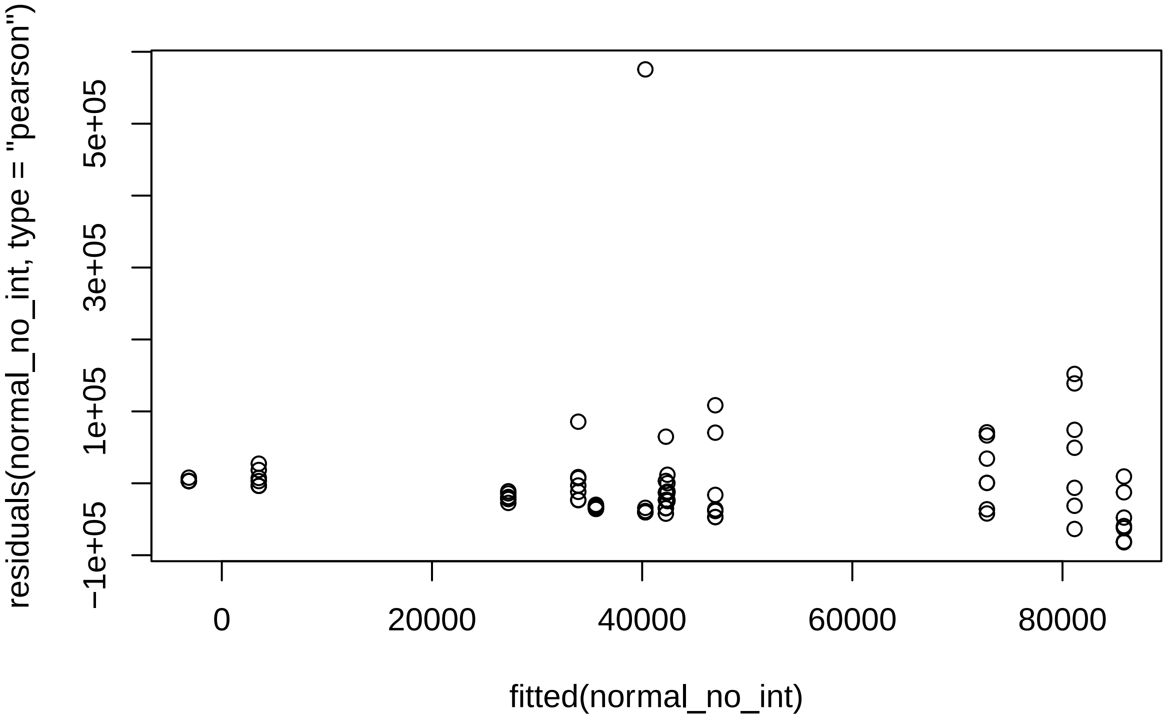
Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

Model Selection

```
#Start with the simplest models
#Normal distribution
#No interaction
normal_no_int<-glm(copies_g~treatment+species,
                   family=gaussian(link="identity"),
                   data=SUBSET)
summary(normal_no_int) #overdipsersed
```

```
##
## Call:
## glm(formula = copies_g ~ treatment + species, family = gaussian(link = "identity"),
##     data = SUBSET)
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)            46960      21148   2.221   0.0293 *
## treatment240448      38888      21148   1.839   0.0697 .
## treatment240720      -6656      21148  -0.315   0.7538
## speciesFestuca      -43442      24419  -1.779   0.0791 .
## speciesGaillardia   -13040      24419  -0.534   0.5949
## speciesTaraxacum     -4702      24419  -0.193   0.8478
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 6261155537)
##
##     Null deviance: 5.4627e+11  on 83  degrees of freedom
## Residual deviance: 4.8837e+11  on 78  degrees of freedom
## AIC: 2141
##
## Number of Fisher Scoring iterations: 2
```

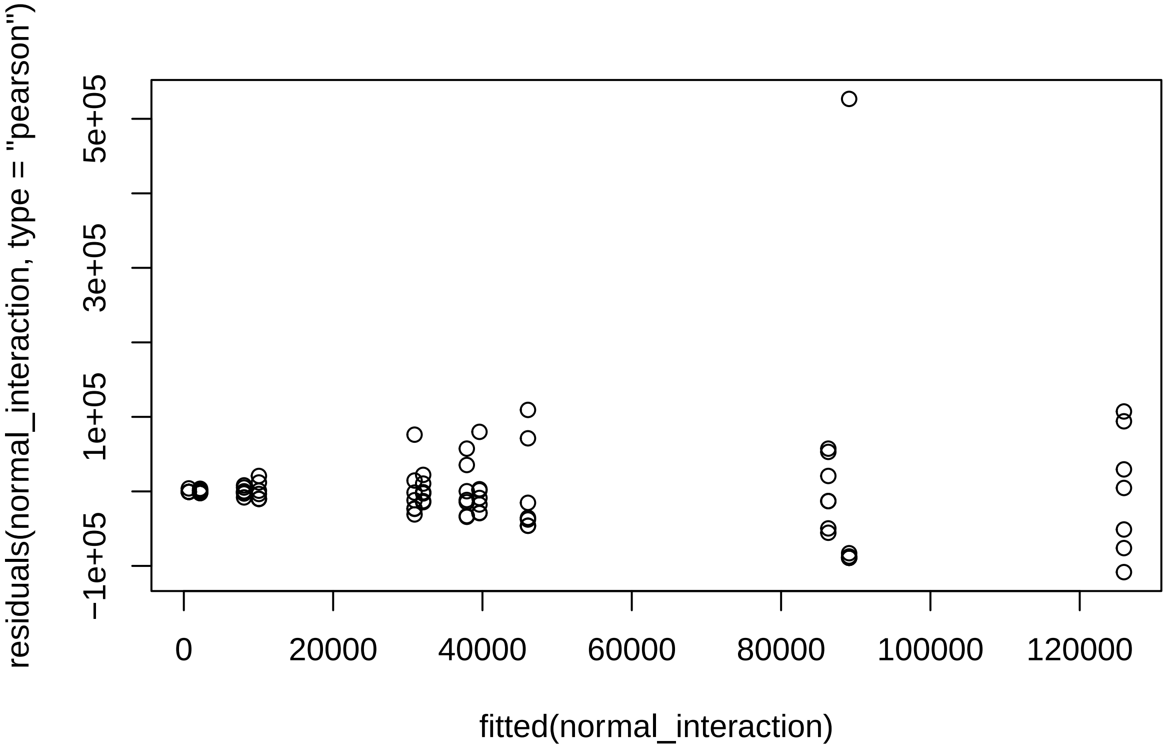
```
plot(fitted(normal_no_int), residuals(normal_no_int, type= "pearson"))
```



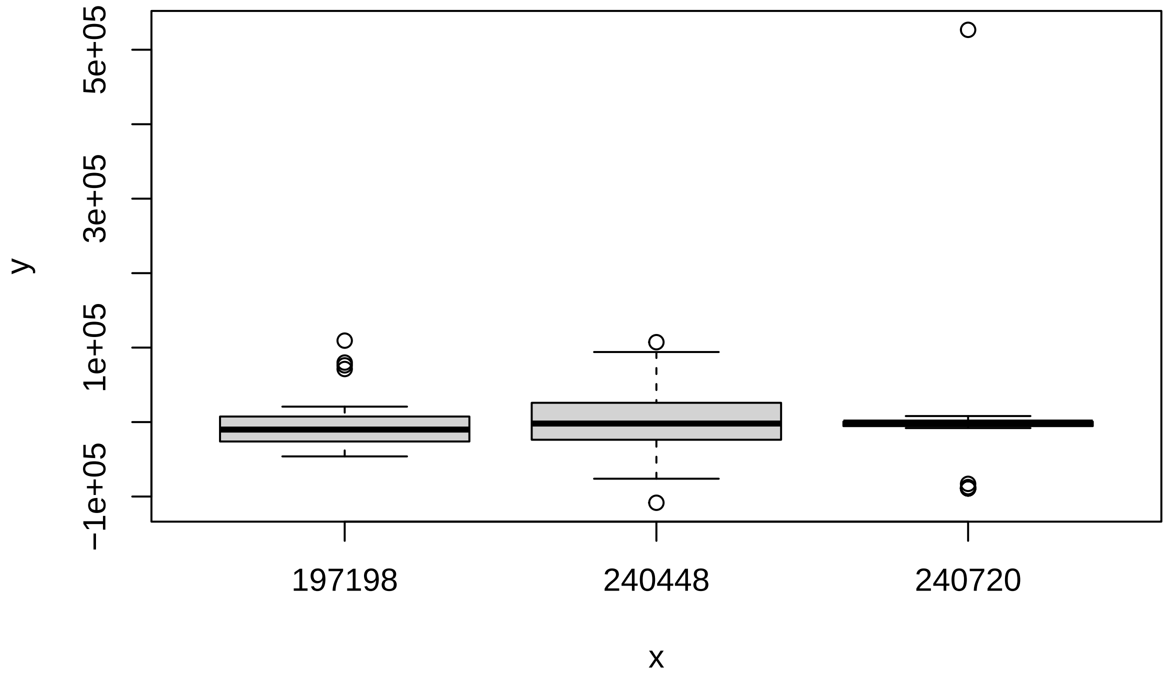
```
#Awful, floor like pattern,
#heteroskedastic fan spread

#Look at residuals of a normal distibrution with an interaction term
#INTERACTION
normal_interaction<-glm(copies_g~treatment*species,
                        family=gaussian(link="identity"),
                        data=SUBSET)
summary(normal_interaction) #overdispersed
```

```
##
## Call:
## glm(formula = copies_g ~ treatment * species, family = gaussian(link = "identity"),
##     data = SUBSET)
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          46095      29128   1.583   0.1179
## treatment240448                      -8192      41193  -0.199   0.8429
## treatment240720                      43019      41193   1.044   0.2998
## speciesFestuca                      -36038      41193  -0.875   0.3846
## speciesGaillardia                    -6495      41193  -0.158   0.8752
## speciesTaraxacum                    -15190      41193  -0.369   0.7134
## treatment240448:speciesFestuca       30192      58255   0.518   0.6059
## treatment240720:speciesFestuca      -52405      58255  -0.900   0.3713
## treatment240448:speciesGaillardia    54916      58255   0.943   0.3490
## treatment240720:speciesGaillardia   -74550      58255  -1.280   0.2048
## treatment240448:speciesTaraxacum    103211      58255   1.772   0.0807 .
## treatment240720:speciesTaraxacum    -71745      58255  -1.232   0.2221
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 5938957379)
##
##     Null deviance: 5.4627e+11  on 83  degrees of freedom
## Residual deviance: 4.2760e+11  on 72  degrees of freedom
## AIC: 2141.8
##
## Number of Fisher Scoring iterations: 2
```

```
plot(fitted(normal_interaction), residuals(normal_interaction, type = "pearson"))
```



```r
#Still poor, heteroskedasti fan shape, floor pattern less pronounced

#View residuals with each predictor
plot(SUBSET$treatment, residuals(normal_interaction, type="pearson"))
```

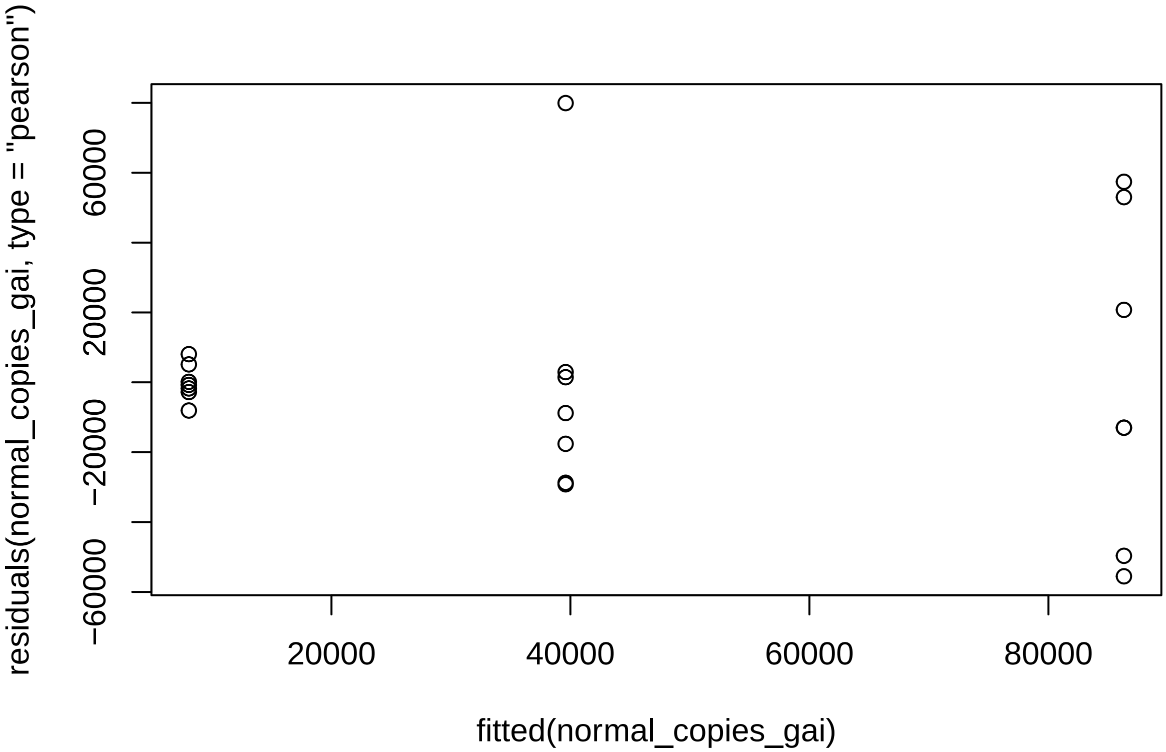


```
#240720 has a much smaller residual range
#Festuca has a much smaller residual range than all others. Gaillardia range is
#smaller than bromus and festuca. Residual variance is varied across species.

#View each species and treatment independently to assess where the
#heteroskedasticity and floor like pattern (assumed zeroes) arises  --
#which components drive the heteorskedasticity and overdispersion?

#GAILLARDIA
normal_copies_gai<-glm(copies_g~treatment,
                       family=gaussian(link="identity"),
                       data=subset_gaillardia)
summary(normal_copies_gai)
```

```
##
## Call:
## glm(formula = copies_g ~ treatment, family = gaussian(link = "identity"),
##     data = subset_gaillardia)
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)          39600      12927   3.063  0.00669 **
## treatment240448      46724      18282   2.556  0.01986 *
## treatment240720     -31531      18282  -1.725  0.10170
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for gaussian family taken to be 1169766618)
##
##     Null deviance: 4.2759e+10  on 20  degrees of freedom
## Residual deviance: 2.1056e+10  on 18  degrees of freedom
## AIC: 502.84
##
## Number of Fisher Scoring iterations: 2
```

```
plot(fitted(normal_copies_gai), residuals(normal_copies_gai, type = "pearson"))
```

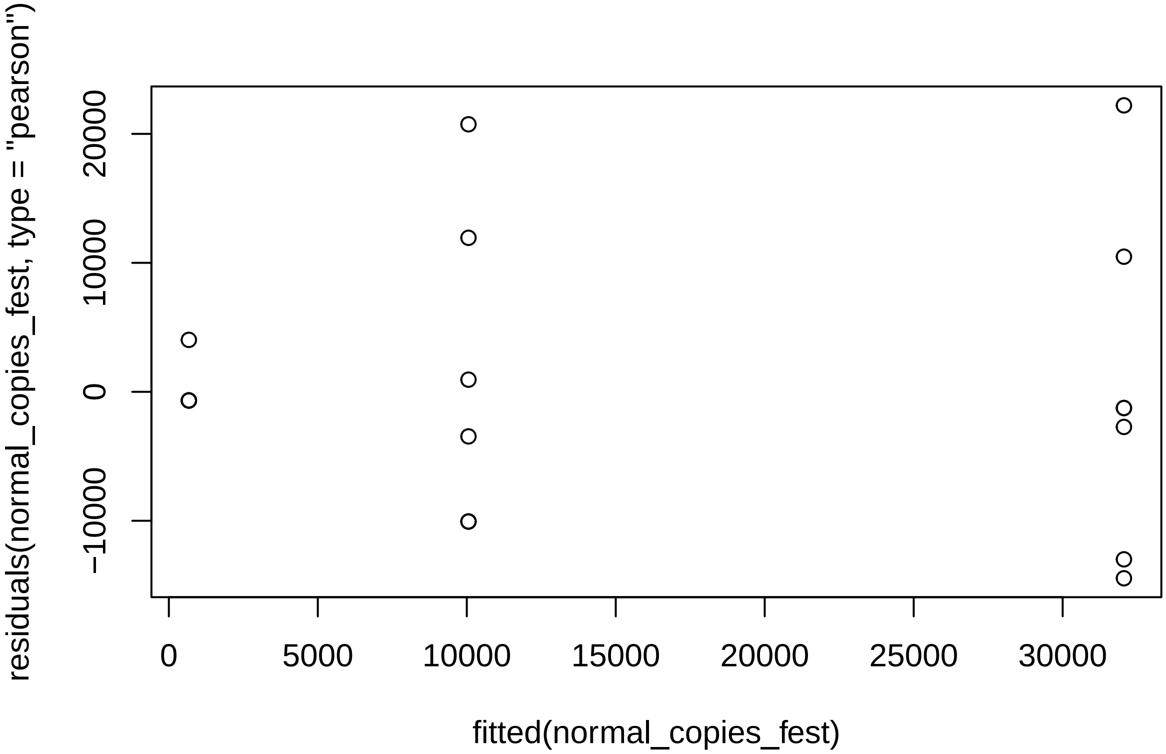


```
#Heteroskedastic and over dispersed

#FESTUCA
normal_copies_fest<-glm(copies_g~treatment,
                        family=gaussian(link="identity"),
                        data=subset_festuca)
summary(normal_copies_fest)
```

```
##
## Call:
## glm(formula = copies_g ~ treatment, family = gaussian(link = "identity"),
##     data = subset_festuca)
##
## Coefficients:
```

```
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)          10057       3883   2.590 0.018473 *
## treatment240448      22000       5491   4.007 0.000828 ***
## treatment240720      -9386       5491  -1.709 0.104564
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 105524872)
##
##     Null deviance: 5532871589  on 20  degrees of freedom
## Residual deviance: 1899447697  on 18  degrees of freedom
## AIC: 452.32
##
## Number of Fisher Scoring iterations: 2
```

```
plot(fitted(normal_copies_fest), residuals(normal_copies_fest, type = "pearson"))
```

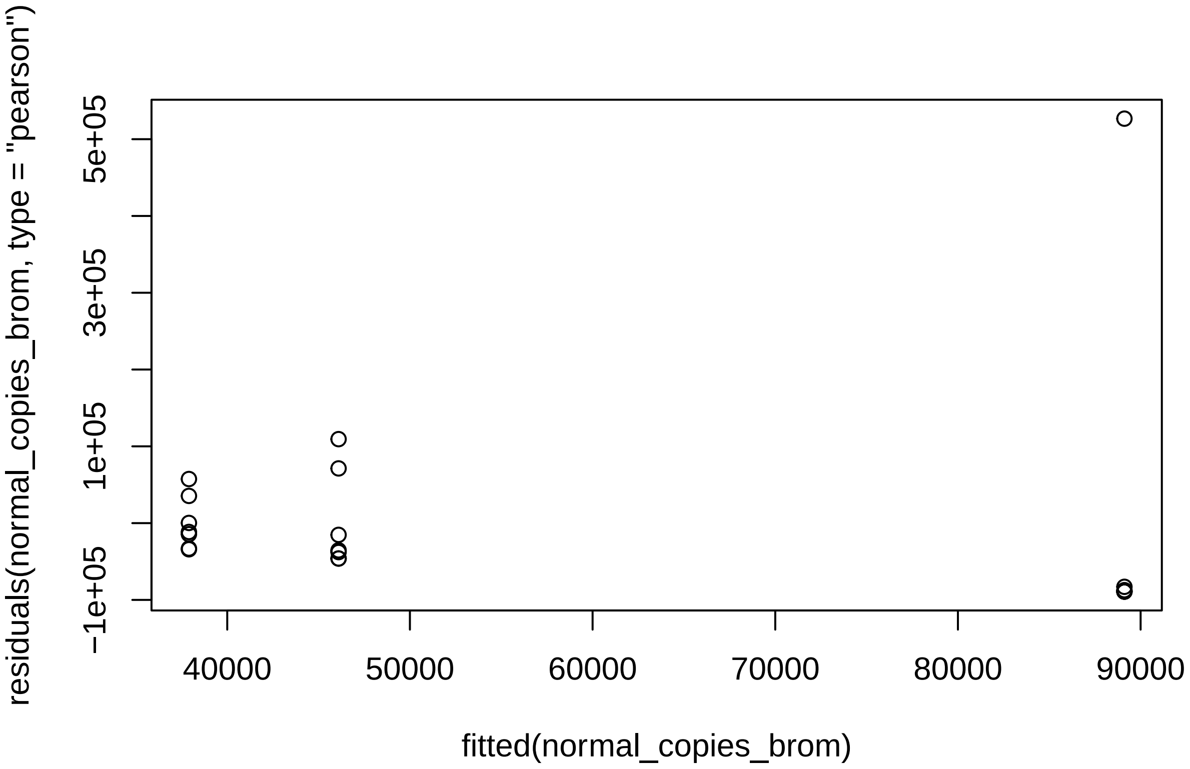


```
#heteroskedastic and overdispersed, less residual dispersion than gaillardia

#BROMUS
normal_copies_brom<-glm(copies_g~treatment,
                        family=gaussian(link="identity"),
                        data=subset_bromus)
summary(normal_copies_brom)
```

```
##
```

```
## Call:
## glm(formula = copies_g ~ treatment, family = gaussian(link = "identity"),
##     data = subset_bromus)
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)       46095      53083   0.868    0.397
## treatment240448   -8192      75071  -0.109    0.914
## treatment240720   43019      75071   0.573    0.574
##
## (Dispersion parameter for gaussian family taken to be 19724819364)
##
##     Null deviance: 3.6564e+11  on 20  degrees of freedom
## Residual deviance: 3.5505e+11  on 18  degrees of freedom
## AIC: 562.17
##
## Number of Fisher Scoring iterations: 2
```

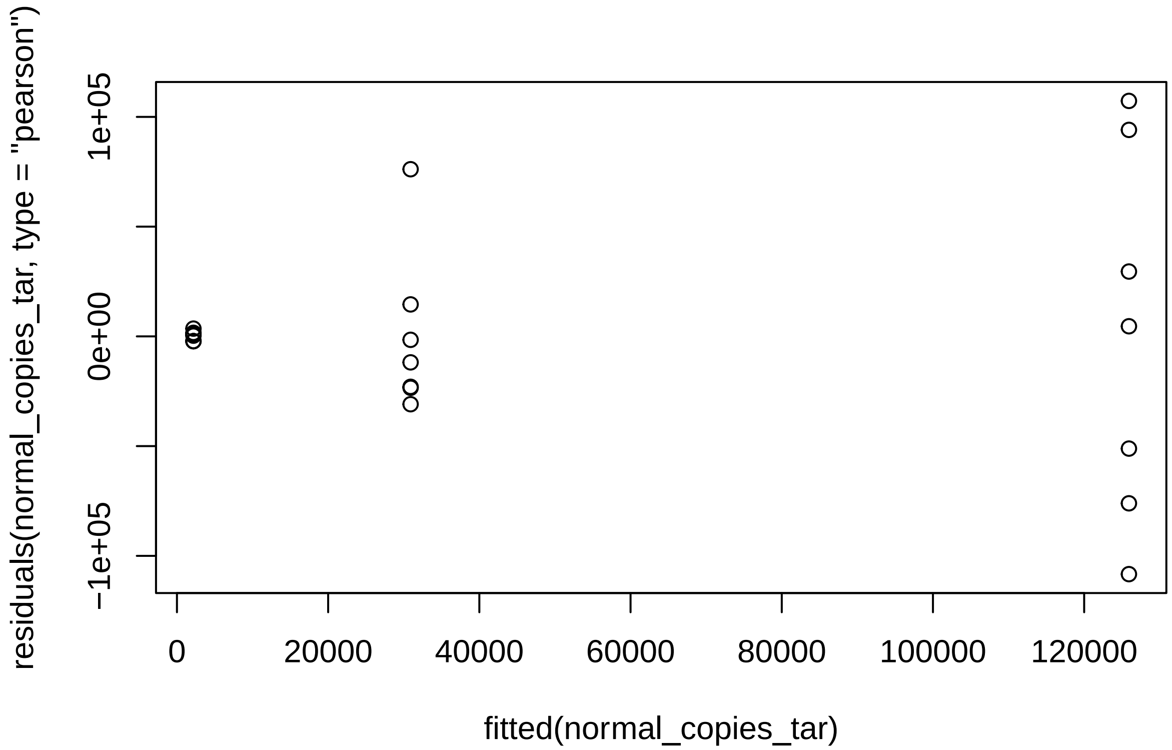
```
plot(fitted(normal_copies_brom), residuals(normal_copies_brom, type = "pearson"))
```



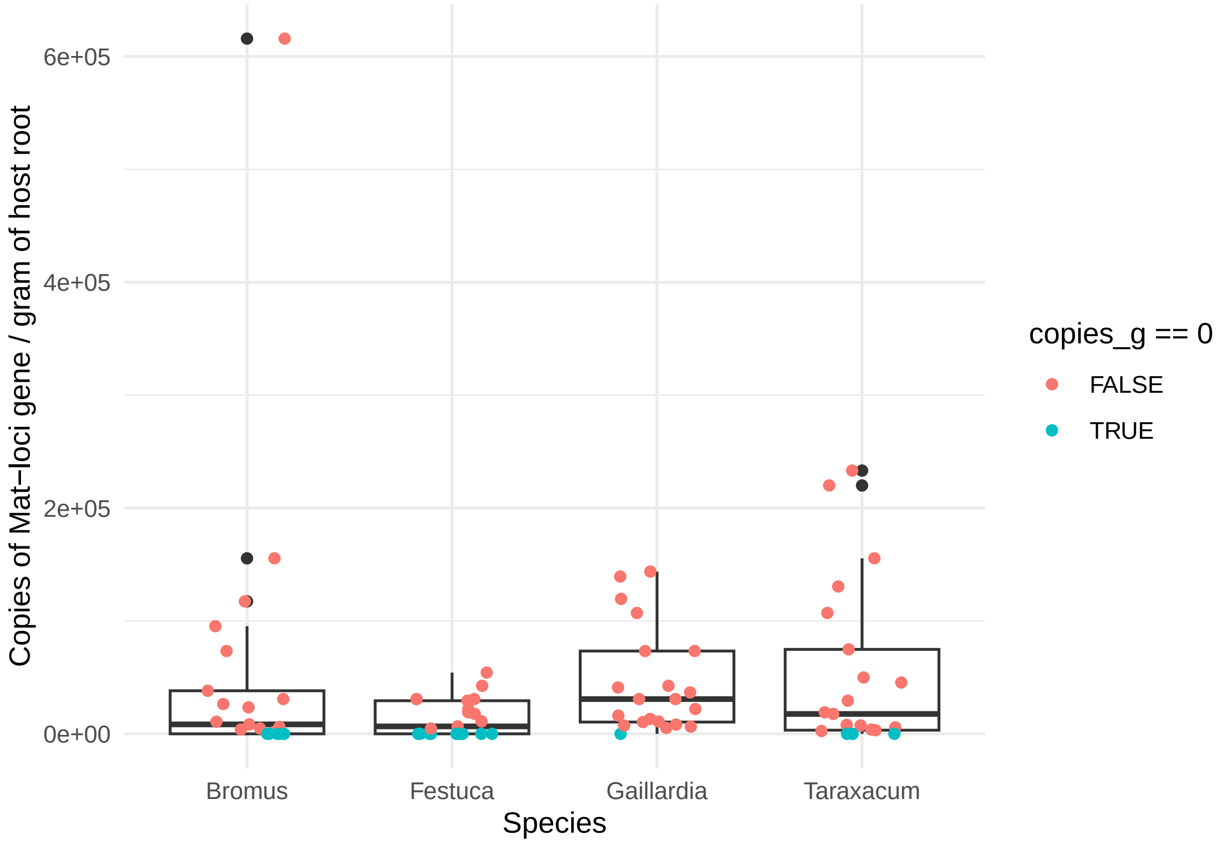
```
#Very heteroskedastic and overdispersed, no fitted values between 5 and 50

#TARAXACUM
normal_copies_tar<-glm(copies_g~treatment,
                       family=gaussian(link="identity"),
                       data=subset_taraxacum)
summary(normal_copies_tar)
```

```
##
## Call:
## glm(formula = copies_g ~ treatment, family = gaussian(link = "identity"),
##     data = subset_taraxacum)
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)        30905      19841   1.558  0.13674
## treatment240448    95019      28060   3.386  0.00329 **
## treatment240720   -28726      28060  -1.024  0.31952
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for gaussian family taken to be 2755718660)
##
##     Null deviance: 1.0833e+11  on 20  degrees of freedom
## Residual deviance: 4.9603e+10  on 18  degrees of freedom
## AIC: 520.83
##
## Number of Fisher Scoring iterations: 2
```

```
plot(fitted(normal_copies_tar), residuals(normal_copies_tar, type = "pearson"))
```



```
#Heteroskedastic and overdispersed, no fitted values between 50 and 170

## Review zero counts between groups
#Variation in zeroes between groups may be driving floor like pattern and
#heteroskedasticity. Visually assess zero counts in species groups and
#treatment groups.
#Visualise zero counts between species
ggplot(SUBSET, aes(x = species, y = copies_g)) +
  geom_boxplot() +
  geom_jitter(width = 0.2, aes(color = copies_g == 0)) +
  theme_minimal() +
  labs(x = "Species", y = "Copies of Mat-loci gene / gram of host root")
```



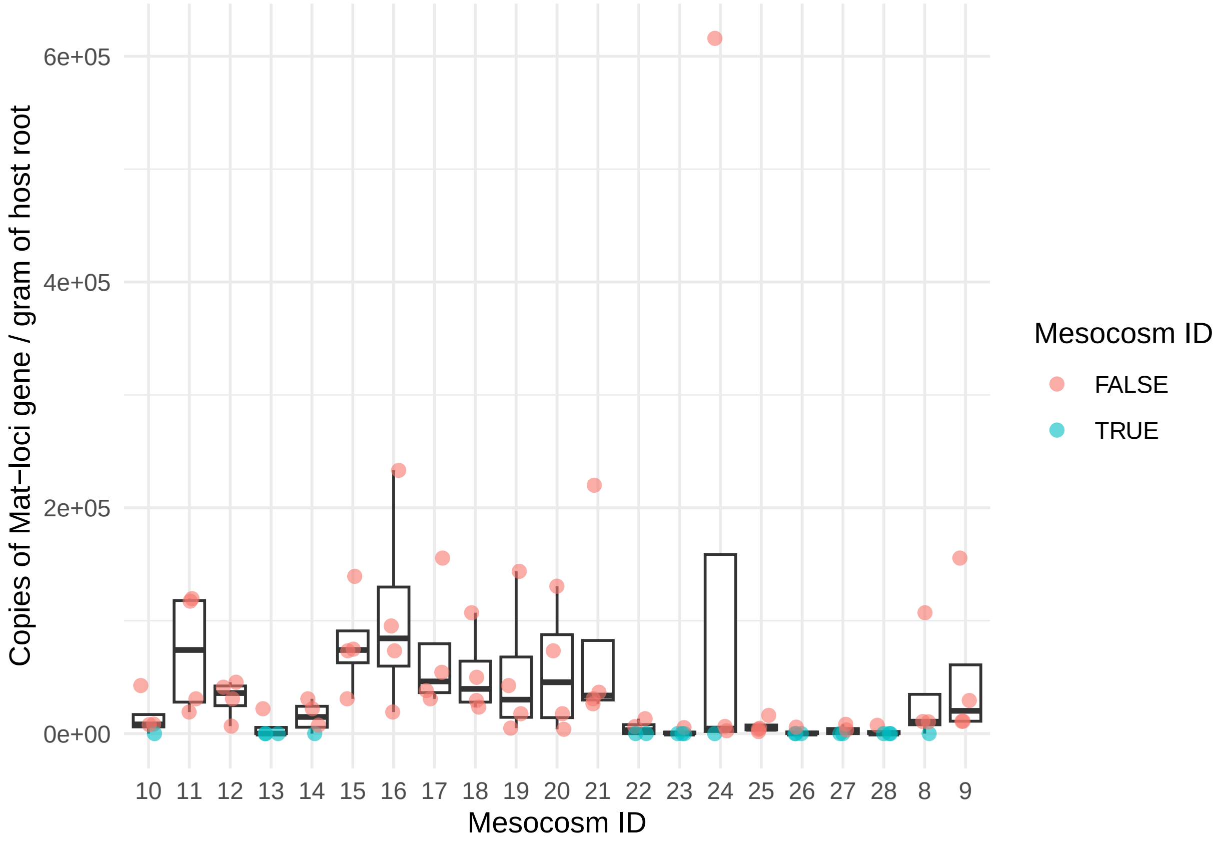
```
#Count the # of zeroes between species
#Count the number of zeros per mesocosm
print(zero_count_species <- SUBSET %>%
        group_by(species) %>%
        summarise(zero_count = sum(copies_g == 0)) %>%
        ungroup())
```

```
## # A tibble: 4 x 2
##   species    zero_count
##   <chr>           <int>
## 1 Bromus              6
## 2 Festuca             9
## 3 Gaillardia          1
## 4 Taraxacum           4
```

```
#The number of zeroes varies between groups: Gaillardia (n=1), Bromus (n=3),
#Taraxacum (n=4) and Festuca (n=9).

#Count the number of zeroes between treatments
print(zero_count_treatment <- SUBSET %>%
        group_by(treatment) %>%
        summarise(zero_count = sum(copies_g == 0)) %>%
        ungroup())
```

```
## # A tibble: 3 x 2
##   treatment zero_count
##   <fct>          <int>
## 1 197198             6
## 2 240448             0
## 3 240720            14
```

```
#Zeroes present in 197198 (n=6) and 240720 (n=14), but not in 240448 (n=0).

#Visually check for possible correlation between mesocosm_id and zero count
ggplot(SUBSET, aes(x = mesocosm_id, y = copies_g)) +
  geom_boxplot(outlier.shape = NA) +
  geom_jitter(width = 0.2, aes(color = factor(copies_g == 0)), size = 2,
              alpha = 0.6) +
  labs(x = "Mesocosm ID", y = "Copies of Mat-loci gene / gram of host root",
       color = "Mesocosm ID") +
  theme_minimal()
```



```
#Count the number of zeros per mesocosm
zero_count_mesocosm <- SUBSET %>%
  group_by(mesocosm_id) %>%
  summarise(zero_count = sum(copies_g == 0)) %>%
  ungroup()
print(zero_count_mesocosm, n=Inf)
```

```
## # A tibble: 21 x 2
##    mesocosm_id zero_count
##    <chr>            <int>
##  1 10                   1
##  2 11                   0
##  3 12                   0
##  4 13                   3
##  5 14                   1
##  6 15                   0
##  7 16                   0
##  8 17                   0
##  9 18                   0
## 10 19                   0
## 11 20                   0
## 12 21                   0
## 13 22                   2
## 14 23                   3
## 15 24                   1
## 16 25                   0
## 17 26                   3
## 18 27                   2
## 19 28                   3
## 20 8                    1
## 21 9                    0
```

```
#All mesocosms have at least 1 non-zero observation (isolate established in meso)
```

Summary of zero inflation and heterogeneity of variance

Zero inflation present within the Festuca species group (9/21 observations) and within the treatment group 240720 (14/21 observations). However other subgroups have few or no zeroes present (gaillardia = 1 zero; 240448 = no zeroes). As zero inflated models require some zeroes present to estimate maximum likelihood, low zero counts in both species and treatment subgroups suggests a zero inflated model is not an appropriate fit.

Summary of heterogeneity of variance:

240448 had larger variance spread than 240720 and 197198. 197198 and 240448 possess a outliers that may be challenge model fit, however, the outliers are well within biological range and were retained.

Continue model construction

Try a tweedie model to test if a tweedie dsitibution (power parameter between 1 and 2) reflects the variation in dsitibrution shapes. Tweedie models incorporate zeroes by assuming that zeroes eoccur within the distribrution (i.e. they do not arise as a seperate process).

```
#Estimate power parameter of the interaction to check tweedie distribution
#is approporiate (p.p. should be between 1 == poisson-like and 2 == gamma-like)
#Estimate power parameter of the interaction
p_est_interaction <- tweedie::tweedie.profile(copies_g ~ treatment*species,
                                              p.vec = seq(1.1, 1.9, by = 0.1),
                                              method = "optim",
                                              data=SUBSET)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
## ignored
```

```
## 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9
## .........
```

```
## Warning in tweedie::tweedie.profile(copies_g ~ treatment * species, p.vec = seq(1.1, :   Problem nea
##       Error in glm.fit(x = model.x, y = ydata, weights = weights, offset = offset,  :
##   NA/NaN/Inf in 'x'
##   Examine the data and function inputs carefully.
```

```
## Done.
```

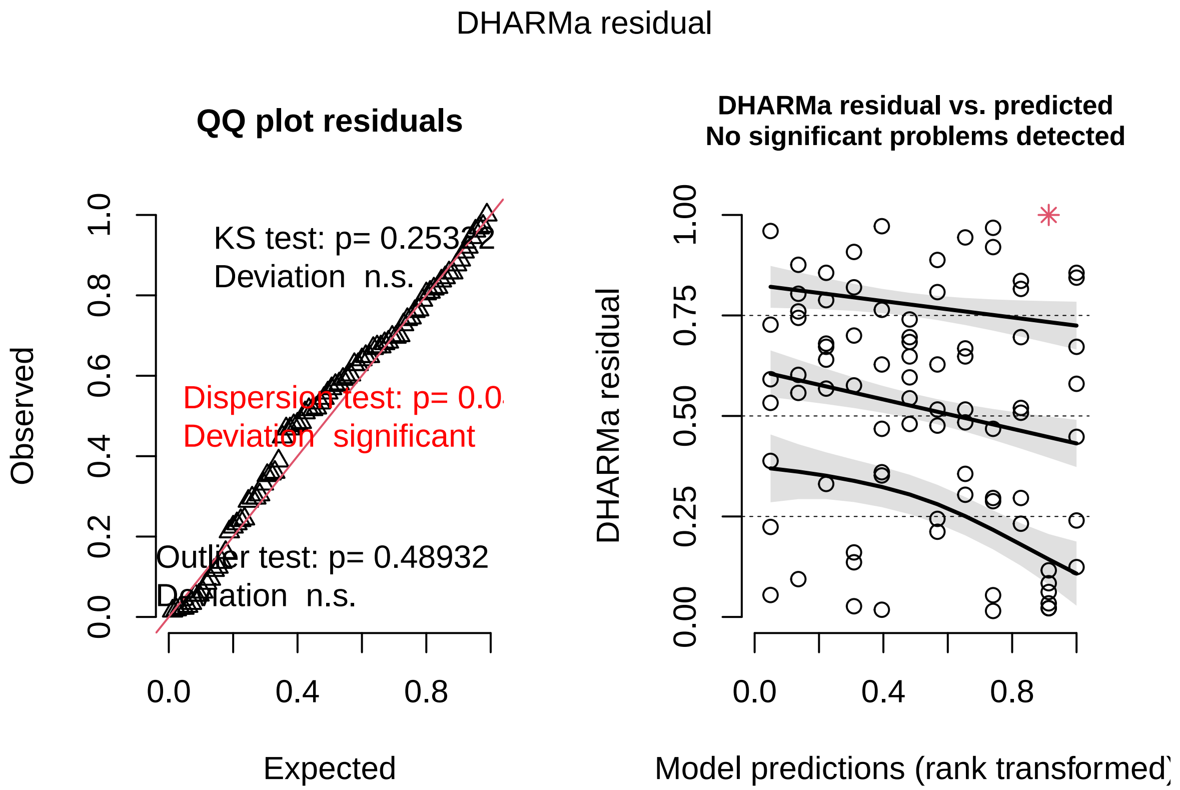
```
#Extract the estimated power parameter
p_est_interaction$p.max #1.5, in tweedie range
```

```
## [1] 1.542857
```

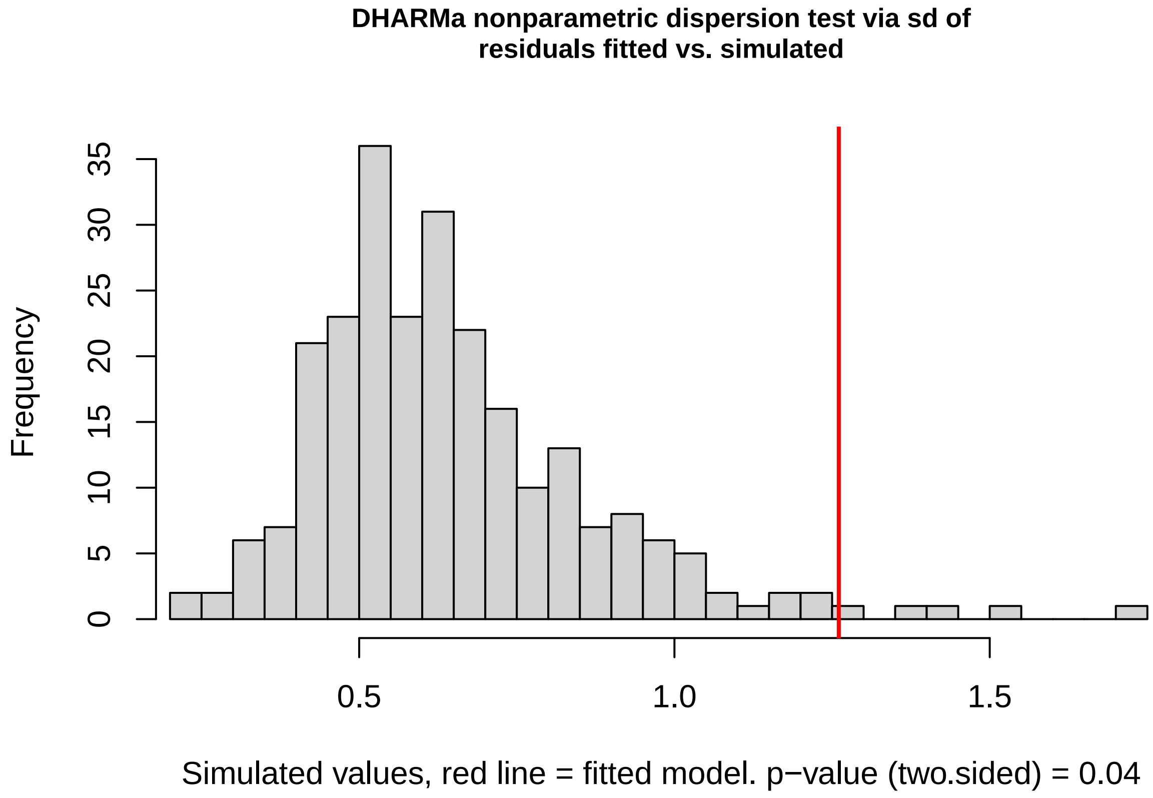
```
#Tweedie model with interaction
glmm_interaction<-glmmTMB(copies_g~species*treatment,
                          family=tweedie,
                          data=SUBSET)
summary(glmm_interaction)
```

```
##  Family: tweedie  ( log )
## Formula:          copies_g ~ species * treatment
## Data: SUBSET
##
##      AIC      BIC   logLik deviance df.resid
##   1592.3   1626.3   -782.1   1564.3       70
##
##
## Dispersion parameter for tweedie family ():  155
##
## Conditional model:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       10.7385     0.3930  27.323  < 2e-16 ***
## speciesFestuca                    -1.5224     0.6832  -2.228  0.02586 *
## speciesGaillardia                 -0.1519     0.5658  -0.268  0.78838
## speciesTaraxacum                  -0.3998     0.5834  -0.685  0.49313
## treatment240448                   -0.1957     0.5688  -0.344  0.73084
## treatment240720                    0.6592     0.5180   1.273  0.20318
## speciesFestuca:treatment240448     1.3549     0.9048   1.498  0.13426
## speciesGaillardia:treatment240448   0.9750     0.7777   1.254  0.20999
## speciesTaraxacum:treatment240448    1.6004     0.7788   2.055  0.03987 *
## speciesFestuca:treatment240720    -3.3663     1.2934  -2.603  0.00925 **
## speciesGaillardia:treatment240720  -2.2500     0.8831  -2.548  0.01084 *
## speciesTaraxacum:treatment240720   -3.3114     1.0429  -3.175  0.00150 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
sim_inter<-simulateResiduals(glmm_interaction)
plot(sim_inter) #Dispersion is significant
```

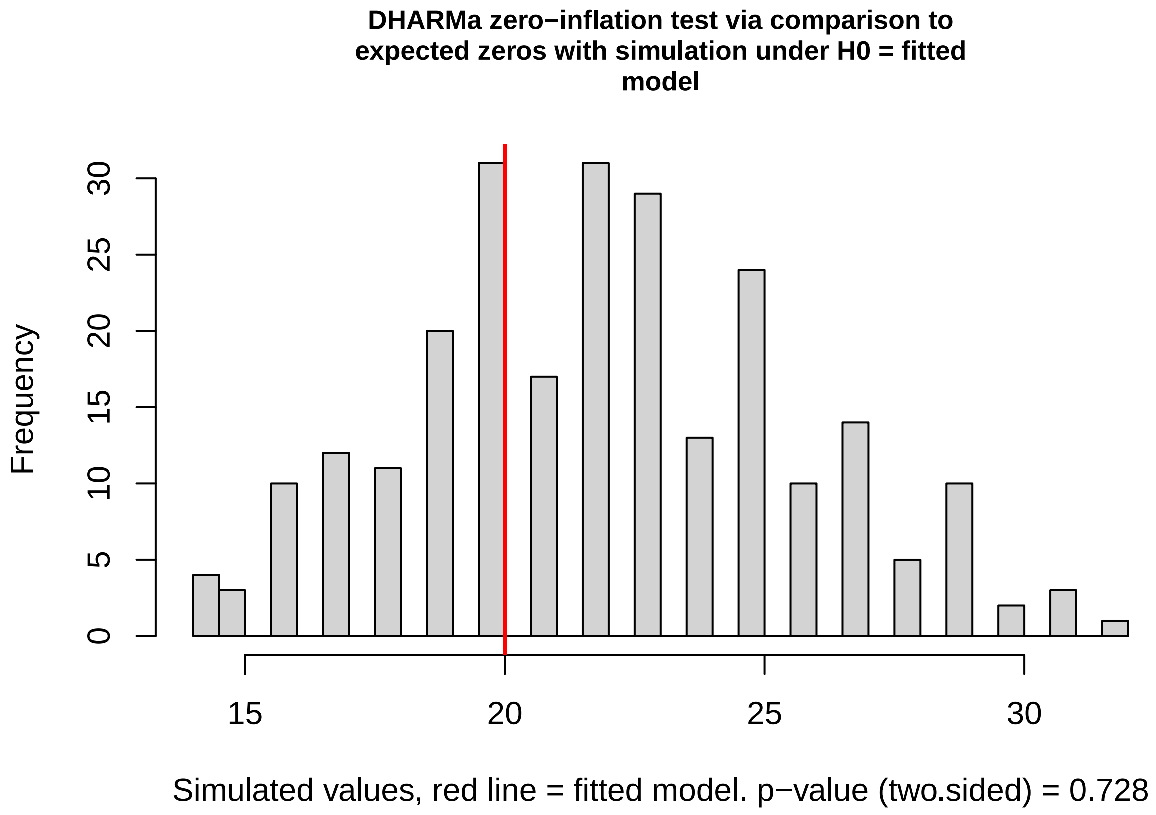


```r
testDispersion(sim_inter) #Most of the data is below the expected median
```



```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 1.952, p-value = 0.04
## alternative hypothesis: two.sided
```

```
#Slightly over dispsersed
testZeroInflation(sim_inter)
```



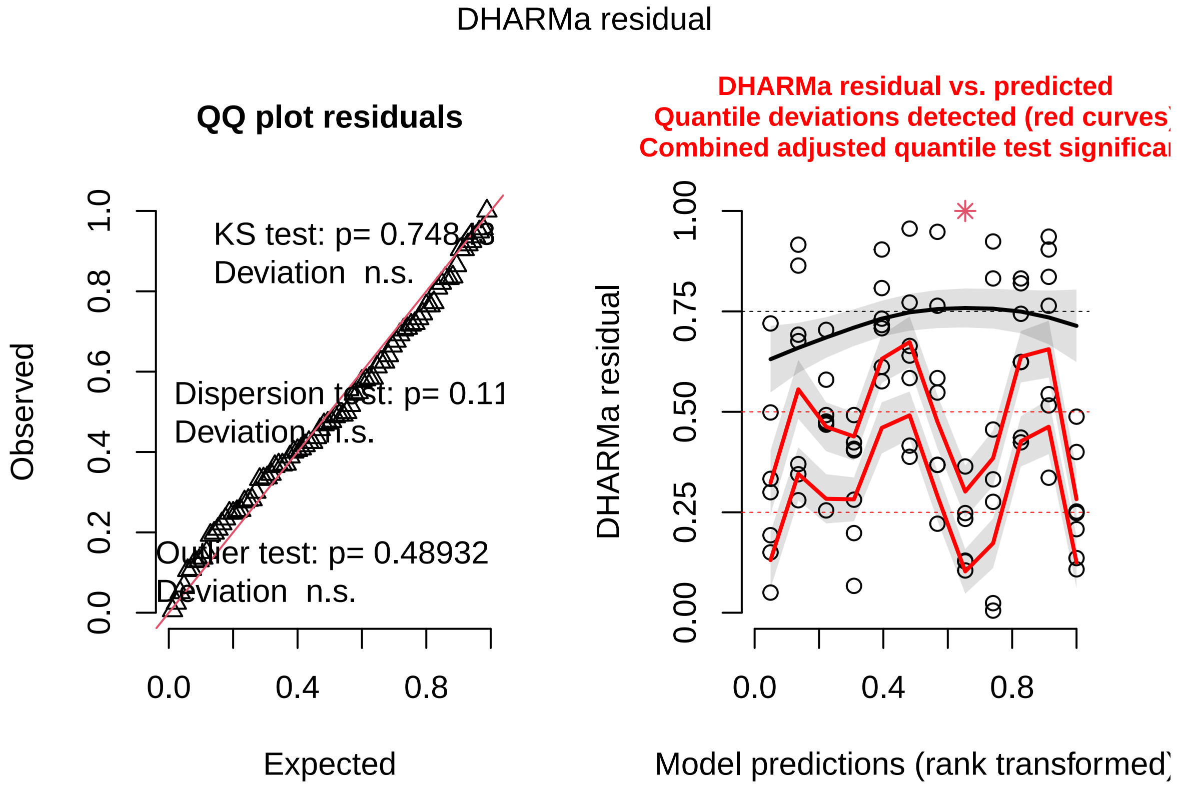
```
##
##  DHARMa zero-inflation test via comparison to expected zeros with
##  simulation under H0 = fitted model
##
## data:  simulationOutput
## ratioObsSim = 0.90351, p-value = 0.728
## alternative hypothesis: two.sided
```

```
#Tweedie model, additive
glmm_additive<-glmmTMB(copies_g~species+treatment,
                       family=tweedie,
                       data=SUBSET)
summary(glmm_additive)
```

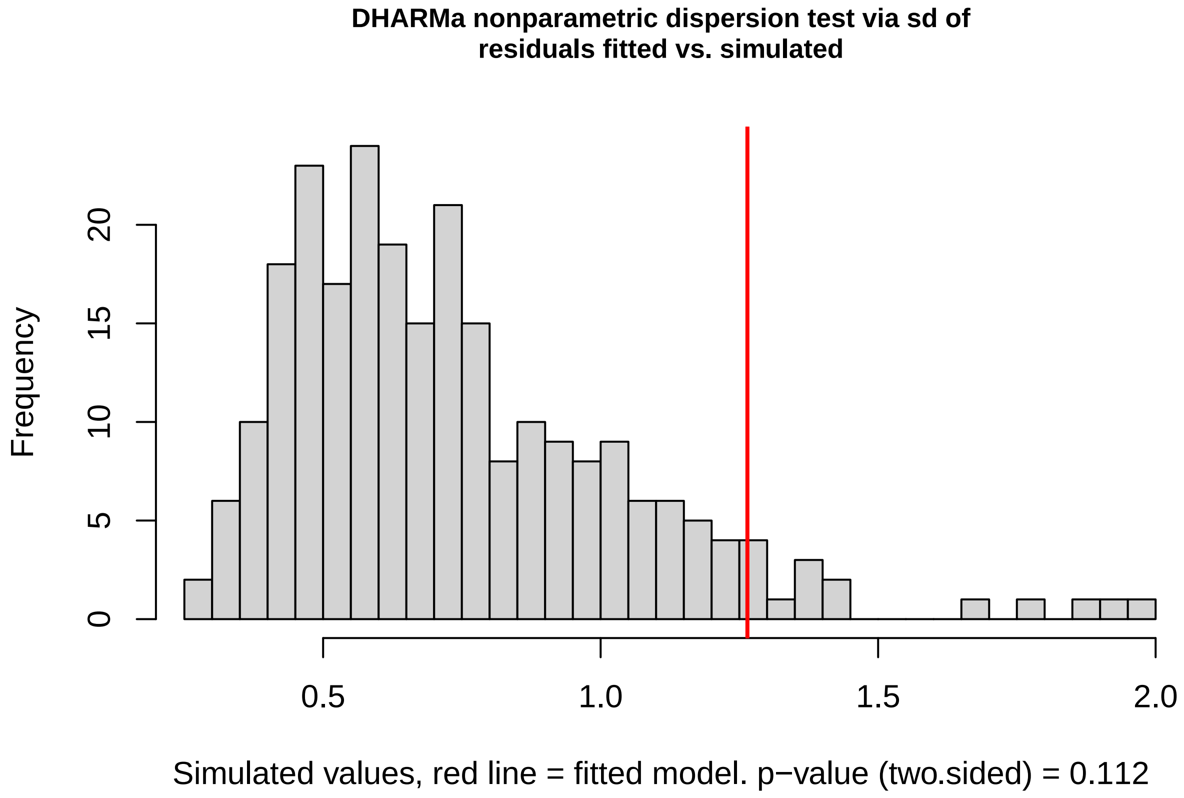
```
##  Family: tweedie  ( log )
## Formula:          copies_g ~ species + treatment
## Data: SUBSET
##
##      AIC      BIC   logLik deviance df.resid
##   1611.5   1631.0   -797.8   1595.5       76
##
##
## Dispersion parameter for tweedie family ():  122
##
## Conditional model:
```

```
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)           10.9673     0.3440   31.88  < 2e-16 ***
## speciesFestuca        -1.8452     0.4551   -4.05 5.03e-05 ***
## speciesGaillardia     -0.6225     0.3888   -1.60  0.10935
## speciesTaraxacum      -0.5590     0.3956   -1.41  0.15766
## treatment240448        0.8993     0.3315    2.71  0.00667 **
## treatment240720       -0.5097     0.3962   -1.29  0.19835
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
sim_inter<-simulateResiduals(glmm_additive)
plot(sim_inter) #Quantil test significant
```

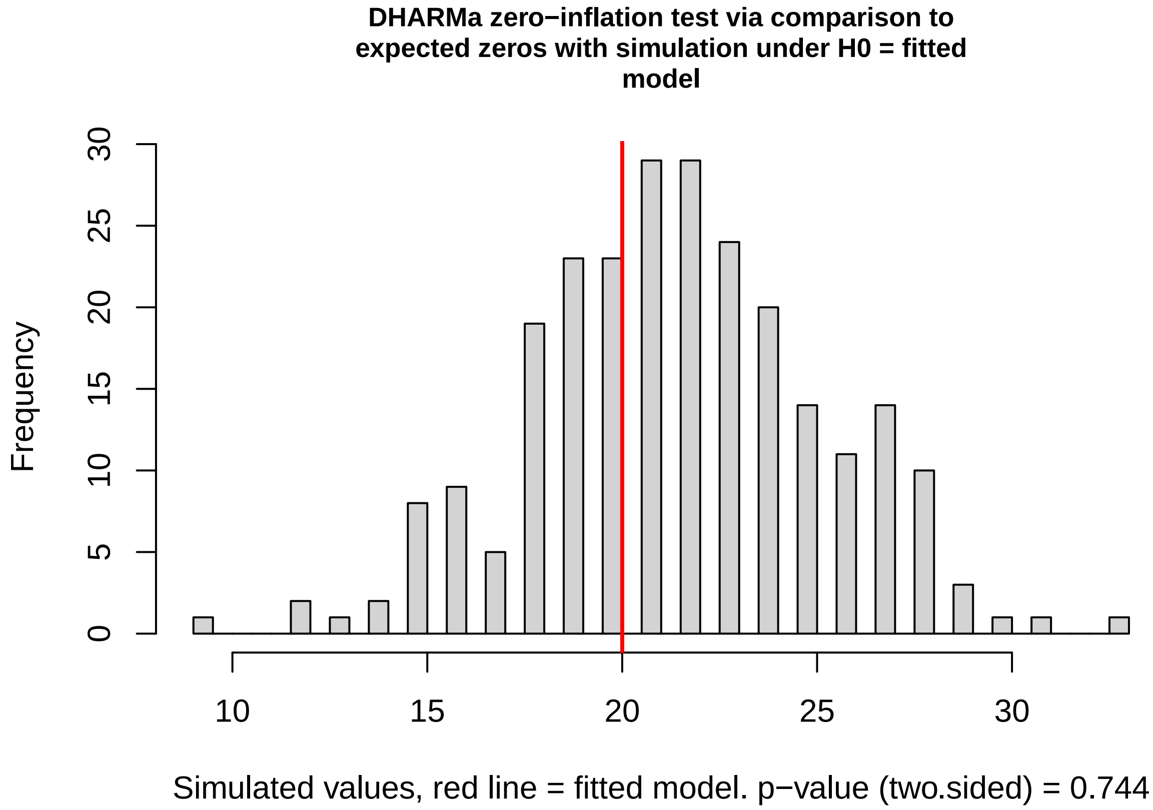


```r
testDispersion(sim_inter) #Most of the data is under expected median
```



```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 1.7252, p-value = 0.112
## alternative hypothesis: two.sided
```

```
testZeroInflation(sim_inter) #Handles zeroes well
```



```
##
##  DHARMa zero-inflation test via comparison to expected zeros with
##  simulation under H0 = fitted model
##
## data:  simulationOutput
## ratioObsSim = 0.92319, p-value = 0.744
## alternative hypothesis: two.sided
```

```
#Tweedie model did not fit well for interaction or additive models
#Test tweedie models on individual species
#GAILARDIA
#Estimate power parameter for gaillardia
p_est_gai <- tweedie::tweedie.profile(copies_g ~ treatment,
  p.vec = seq(1.1, 1.9, by = 0.1),
  method = "optim",
  data=subset_gaillardia)
```

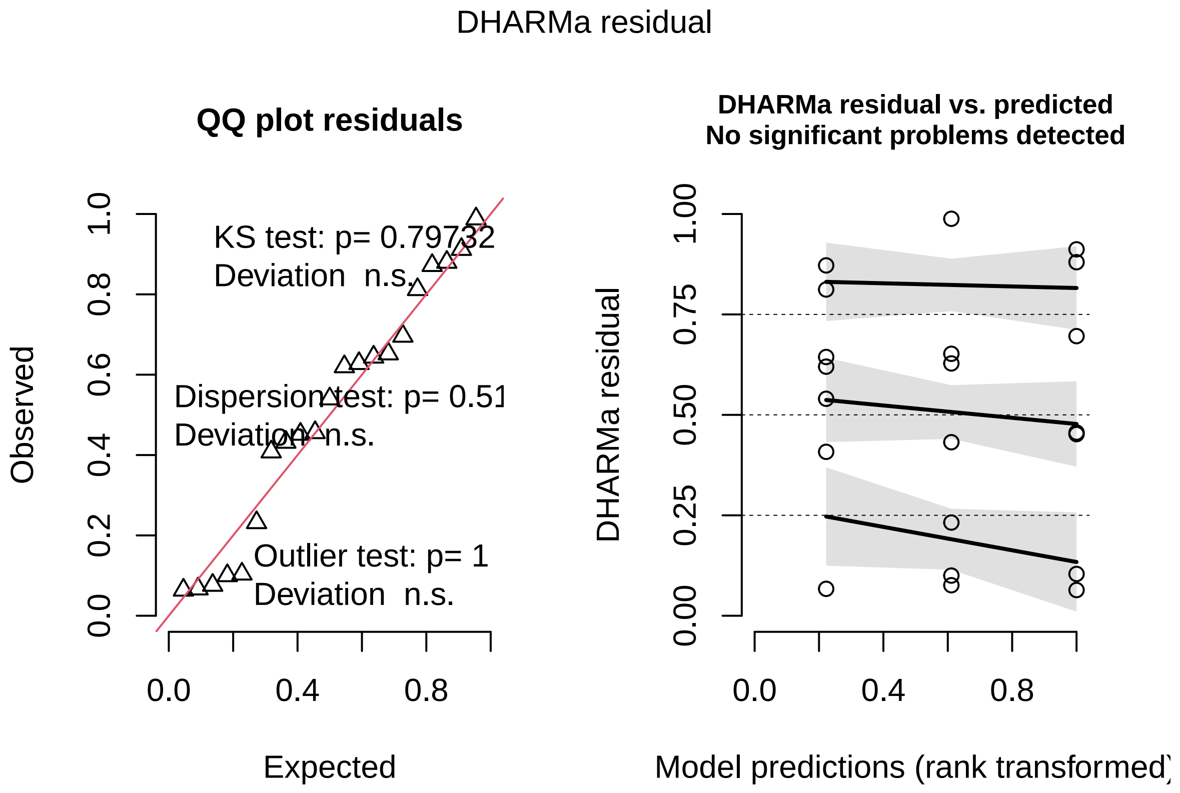
```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
## ignored
```

```
## 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9
## .........Done.
```

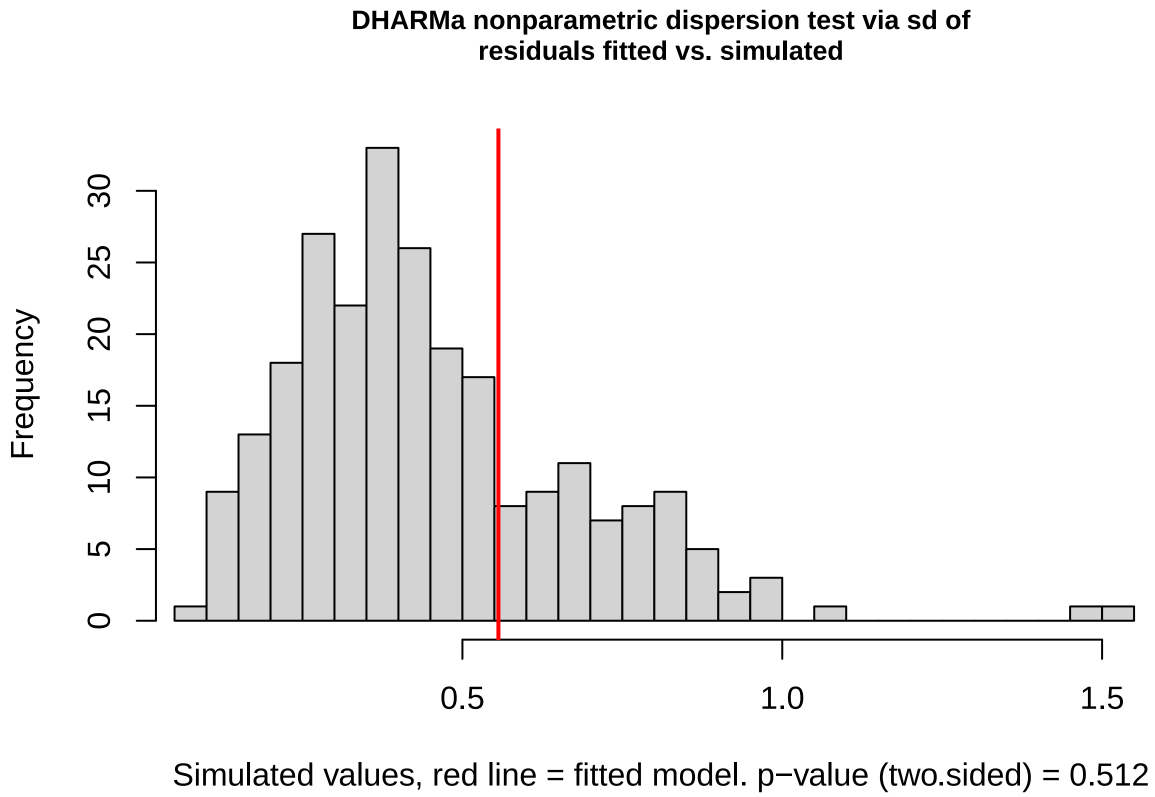
```
## Warning in tweedie::tweedie.profile(copies_g ~ treatment, p.vec = seq(1.1, : Confidence interval can
#Extract the estimated power parameter
p_est_gai$p.max #1.4, good
```

```
## [1] 1.442857
```

```
#Tweedie model with glmmTMB
glmmTMB_gai <- glmmTMB(copies_g ~ treatment,
                      family = tweedie(link="log"),
                      data = subset_gaillardia)

plot(sim_gai<-simulateResiduals(glmmTMB_gai))
```

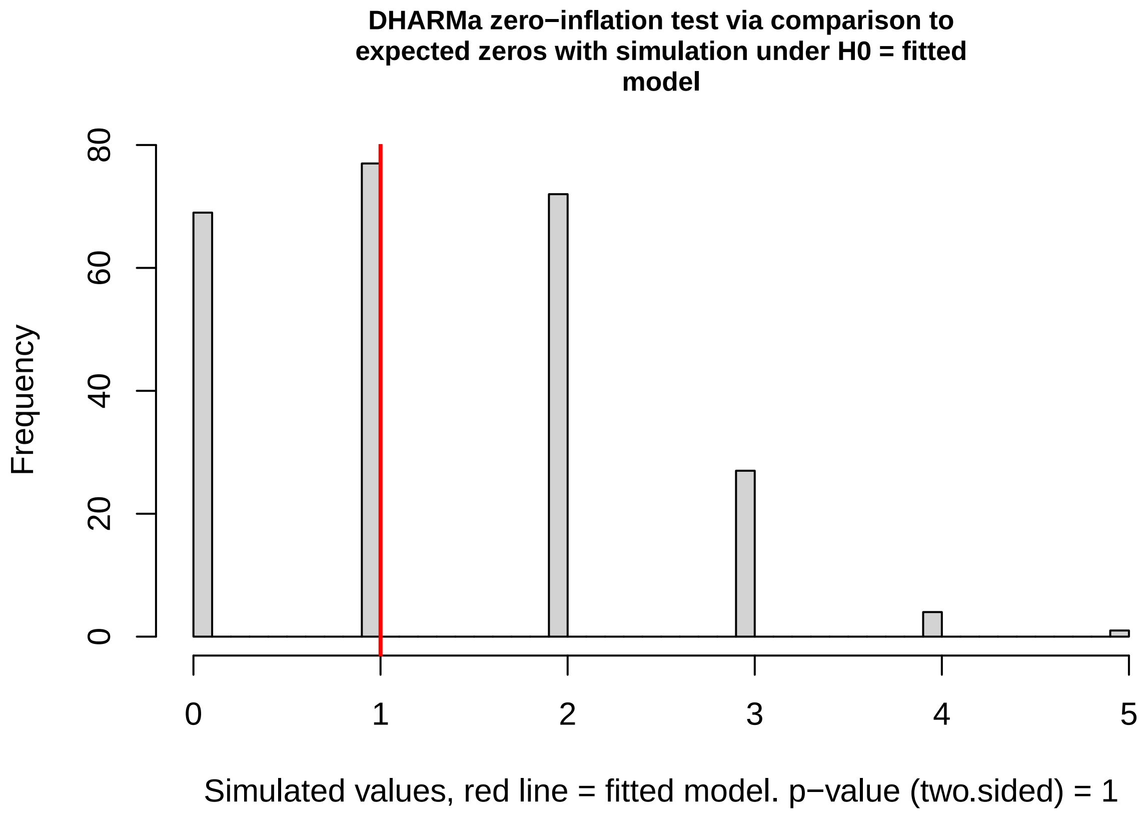


```
testDispersion(sim_gai) #many of the simulated residuals are below the
```



```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 1.2251, p-value = 0.512
## alternative hypothesis: two.sided
```

```
#predicted range, however, dispersion is in the expected range (~1)
testZeroInflation(sim_gai)
```



```
##
##  DHARMa zero-inflation test via comparison to expected zeros with
##  simulation under H0 = fitted model
##
## data:  simulationOutput
## ratioObsSim = 0.77399, p-value = 1
## alternative hypothesis: two.sided
```

```
#FESTUCA
#Estimate power parameter
p_est_fest <- tweedie::tweedie.profile(copies_g ~ treatment,
                                       p.vec = seq(1.1, 1.9, by = 0.01),
                                       method = "optim",
                                       data=subset_festuca)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
## ignored
```

```
## 1.1 1.11 1.12 1.13 1.14 1.15 1.16 1.17 1.18 1.19 1.2 1.21 1.22 1.23 1.24 1.25 1.26 1.27 1.28 1.29 1.3
## ........................................................................
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## .
## Warning: glm.fit: algorithm did not converge

## .

## Warning: glm.fit: algorithm did not converge

## .

## Warning: glm.fit: algorithm did not converge

## .

## Warning: glm.fit: algorithm did not converge

## .

## Warning: glm.fit: algorithm did not converge

## .

## Warning: glm.fit: algorithm did not converge

## .

## Warning: glm.fit: algorithm did not converge

## .

## Warning: glm.fit: algorithm did not converge

## .

## Warning in tweedie::tweedie.profile(copies_g ~ treatment, p.vec = seq(1.1, :   Problem near  p  =  1
##       Error in glm.fit(x = model.x, y = ydata, weights = weights, offset = offset,  :
##   NA/NaN/Inf in 'x'
##   Examine the data and function inputs carefully.

## Done.

## Warning in tweedie::tweedie.profile(copies_g ~ treatment, p.vec = seq(1.1, :
## True maximum possibly not detected.

## Warning in tweedie::tweedie.profile(copies_g ~ treatment, p.vec = seq(1.1, : Confidence interval cann
```

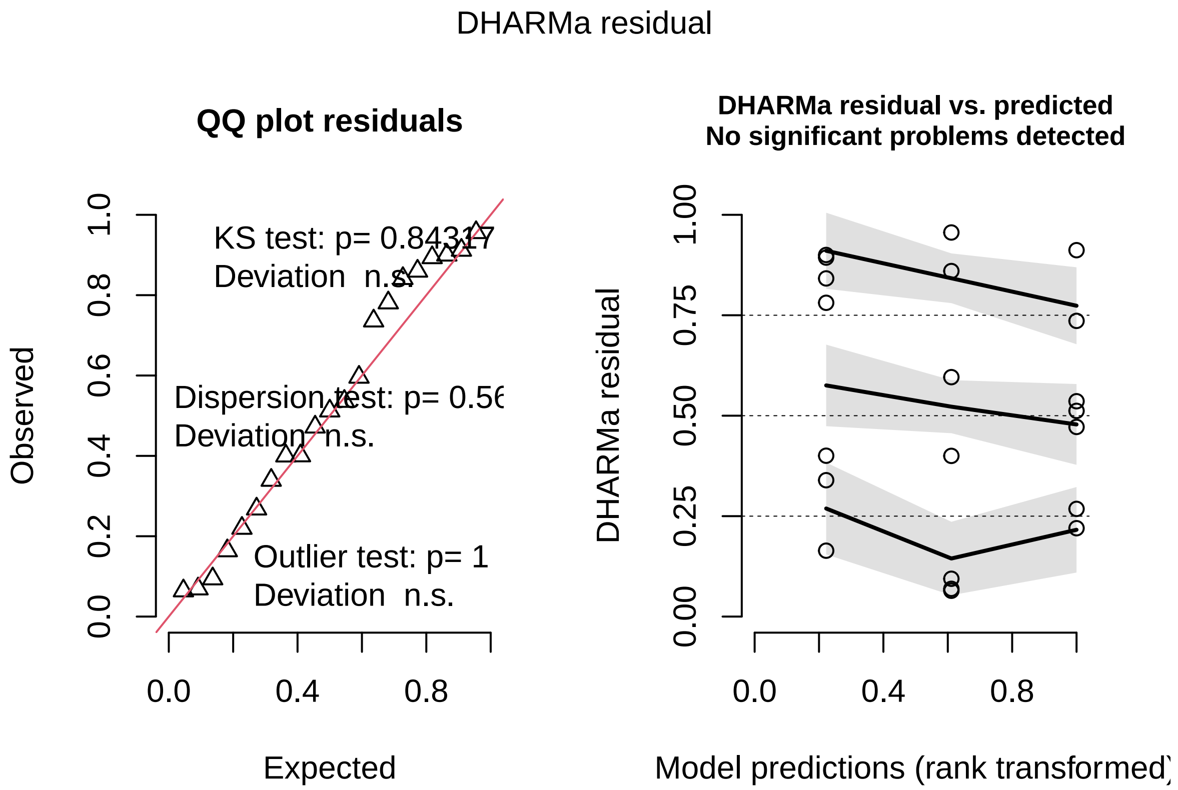
```
#Extract the estimated power parameter
p_est_fest$p.max #COuld not find left CI, tweedie likely not a good fit for
```

```
## [1] 1.1
```

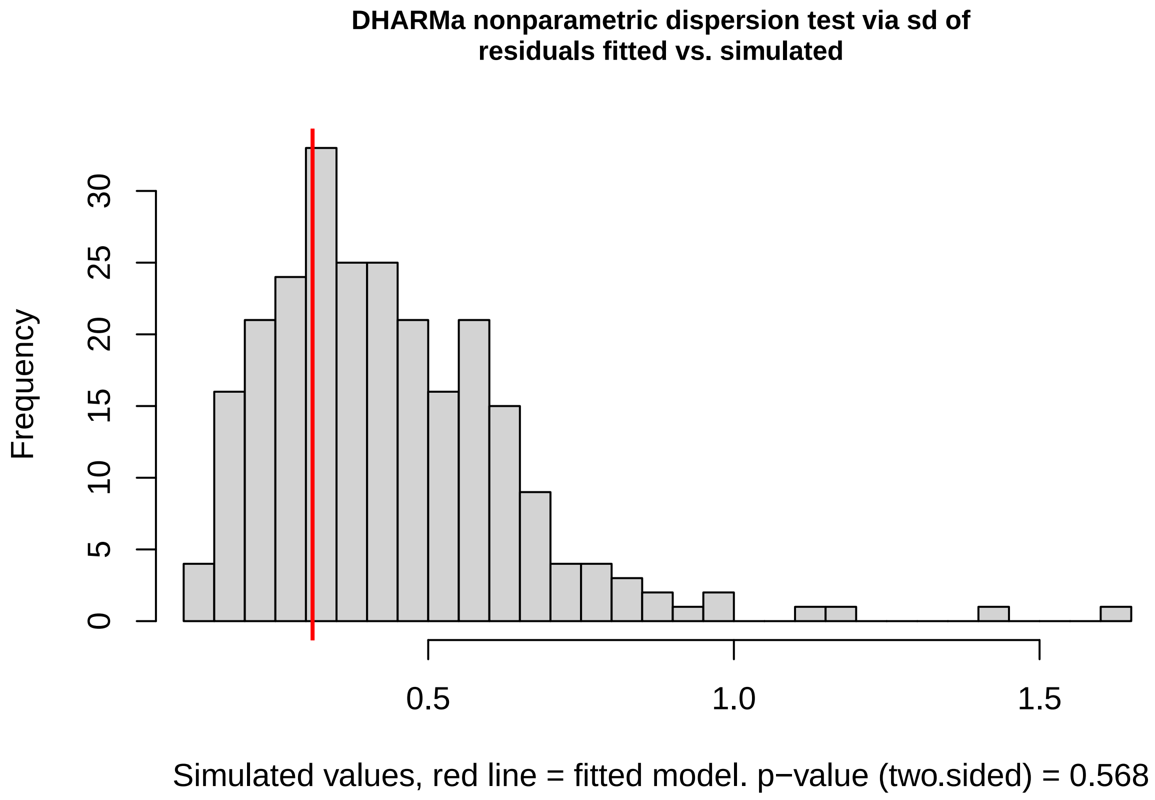
```
#festuca

#Tweedie model with glmmTMB
glmmTMB_fest <- glmmTMB(copies_g~treatment,
                        family = tweedie,
                        data = subset_festuca)
summary(glmmTMB_fest)
```

```
##  Family: tweedie  ( log )
## Formula:          copies_g ~ treatment
## Data: subset_festuca
##
##      AIC      BIC   logLik deviance df.resid
##    280.2    285.4   -135.1    270.2       16
##
##
## Dispersion parameter for tweedie family (): 3.68e+03
##
## Conditional model:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)        9.2160     0.3547  25.981  < 2e-16 ***
## treatment240448    1.1592     0.4122   2.812  0.00492 **
## treatment240720   -2.7071     1.2580  -2.152  0.03141 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#interpret results knowing the estiamted power paraemet was 1.1
plot(sim_fest<-simulateResiduals(glmmTMB_fest))
```



```r
testDispersion(sim_fest) #Looks okay, slightl underdispersed, but not significant
```



```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 0.70902, p-value = 0.568
## alternative hypothesis: two.sided
```

```
#Check predictions
summary(predict(glmmTMB_fest, type = "response"))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   671.1   671.1 10057.1 14261.8 32057.2 32057.2
```

```
mean(subset_festuca$copies_g) #Mean is nearly identical, overfit
```

```
## [1] 14261.79
```

```
#BROMUS
#Estimate power parameter
p_est_brom <- tweedie::tweedie.profile(copies_g ~ treatment,
                                       p.vec = seq(1.1, 1.9, by = 0.1),
                                       method = "optim",
                                       data=subset_bromus)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
## ignored
```

```
## 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9
## .........
```

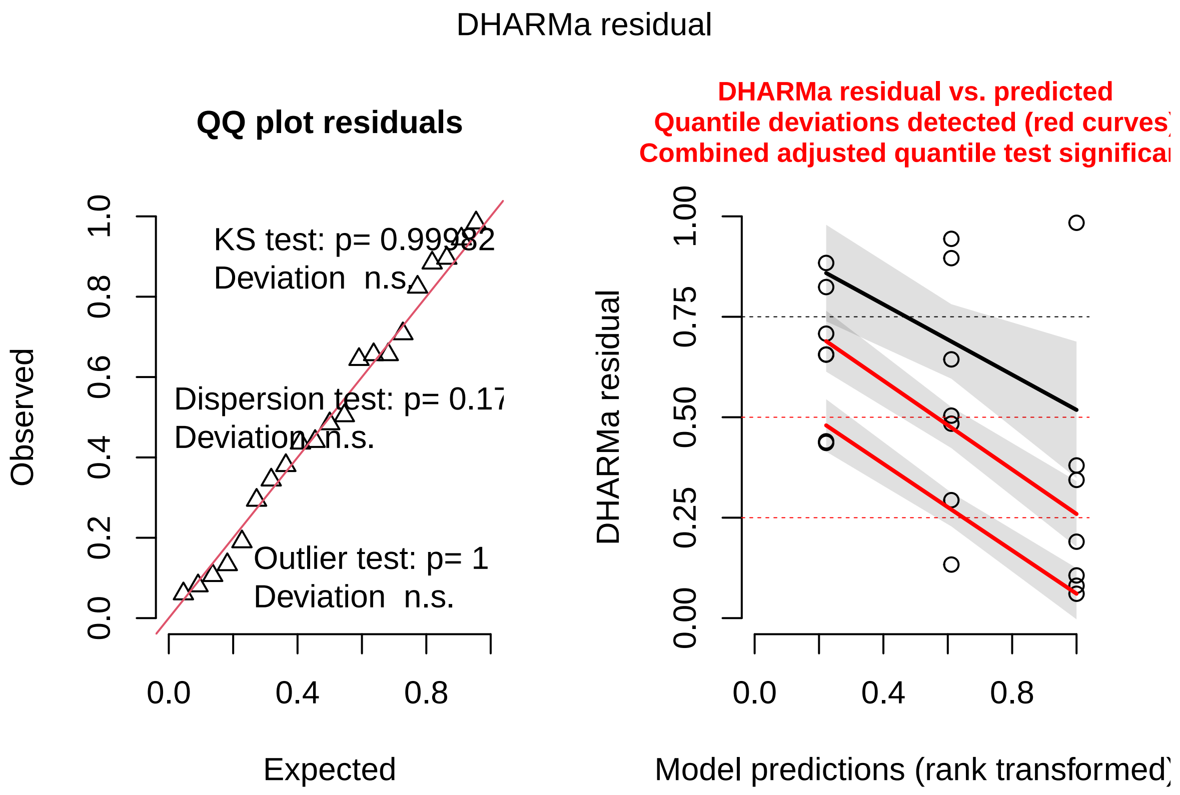
```
## Warning: glm.fit: algorithm did not converge
```

```
## Done.
```

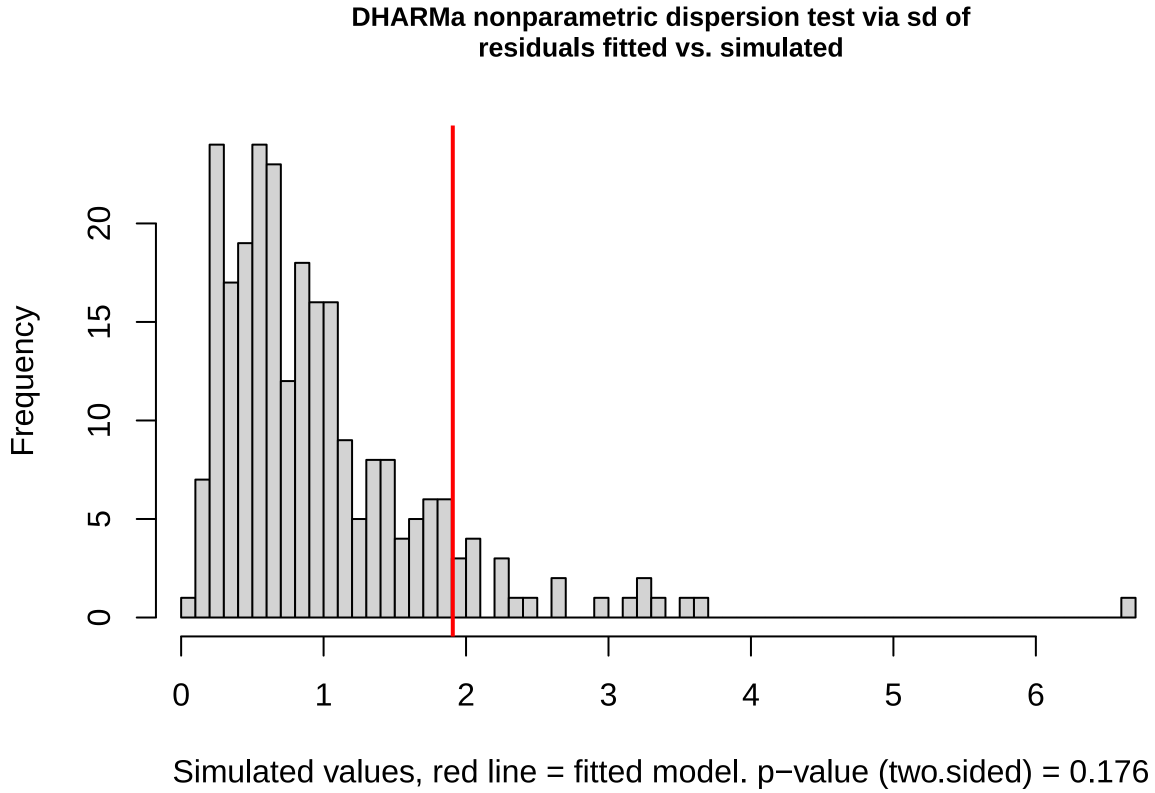
```
#Extract the estimated power parameter
p_est_brom$p.max #1.8, okay
```

```
## [1] 1.769388
```

```
#Tweedie model with glmmTMB #glmmTMB set p = 1.5 for tweedie -- cannot alter
glmmTMB_brom <- glmmTMB(copies_g~treatment,
                        family = tweedie(link="log"),
                        data = subset_bromus)

plot(sim_brom<-simulateResiduals(glmmTMB_brom)) #Significant quantile deviations
```

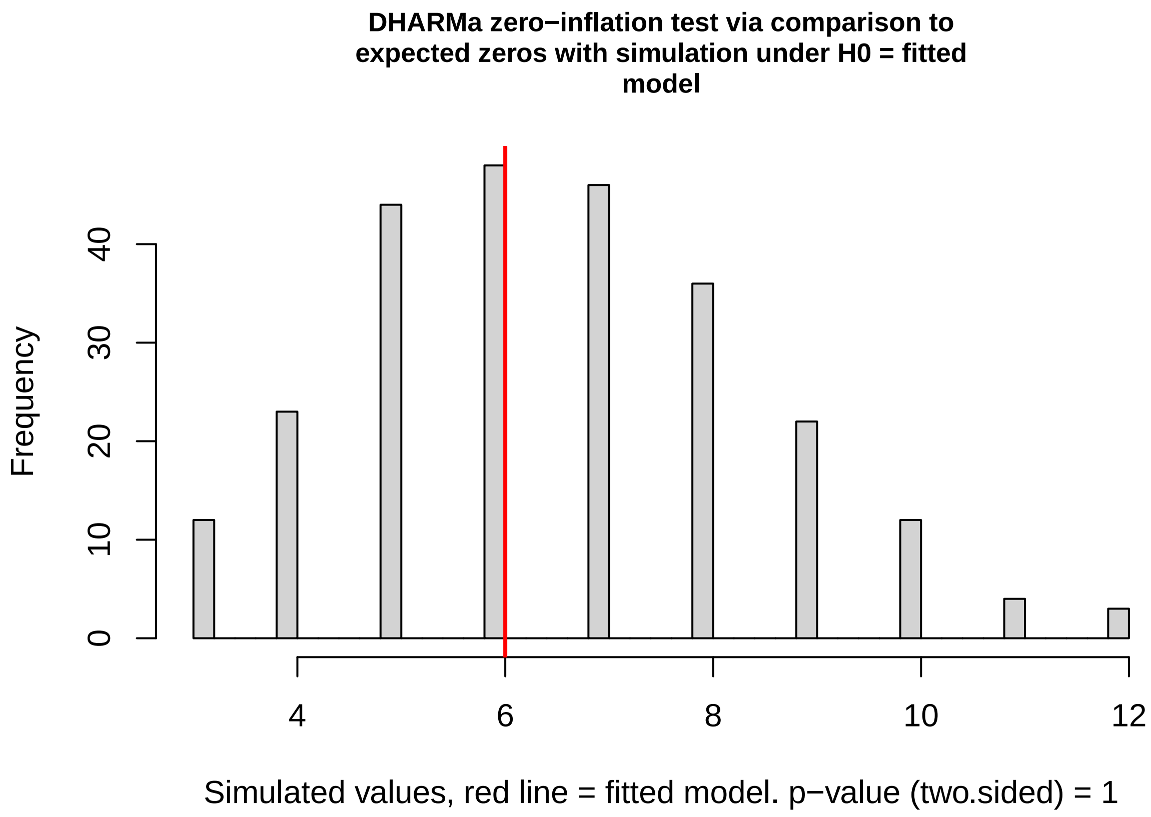


```
testDispersion(sim_brom) #underestimating
```



```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 1.9768, p-value = 0.176
## alternative hypothesis: two.sided
```

```
testZeroInflation(sim_brom) #Okay
```



```
##
##  DHARMa zero-inflation test via comparison to expected zeros with
##  simulation under H0 = fitted model
##
## data:  simulationOutput
## ratioObsSim = 0.91241, p-value = 1
## alternative hypothesis: two.sided
```

```
#TARAXACUM
#Estimate power parameter
p_est_tar <- tweedie::tweedie.profile(copies_g ~ treatment,
                                      p.vec = seq(1.1, 1.9, by = 0.1),
                                      method = "optim",
                                      data=subset_taraxacum)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts argument
## ignored

## 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9
## .........Done.
```

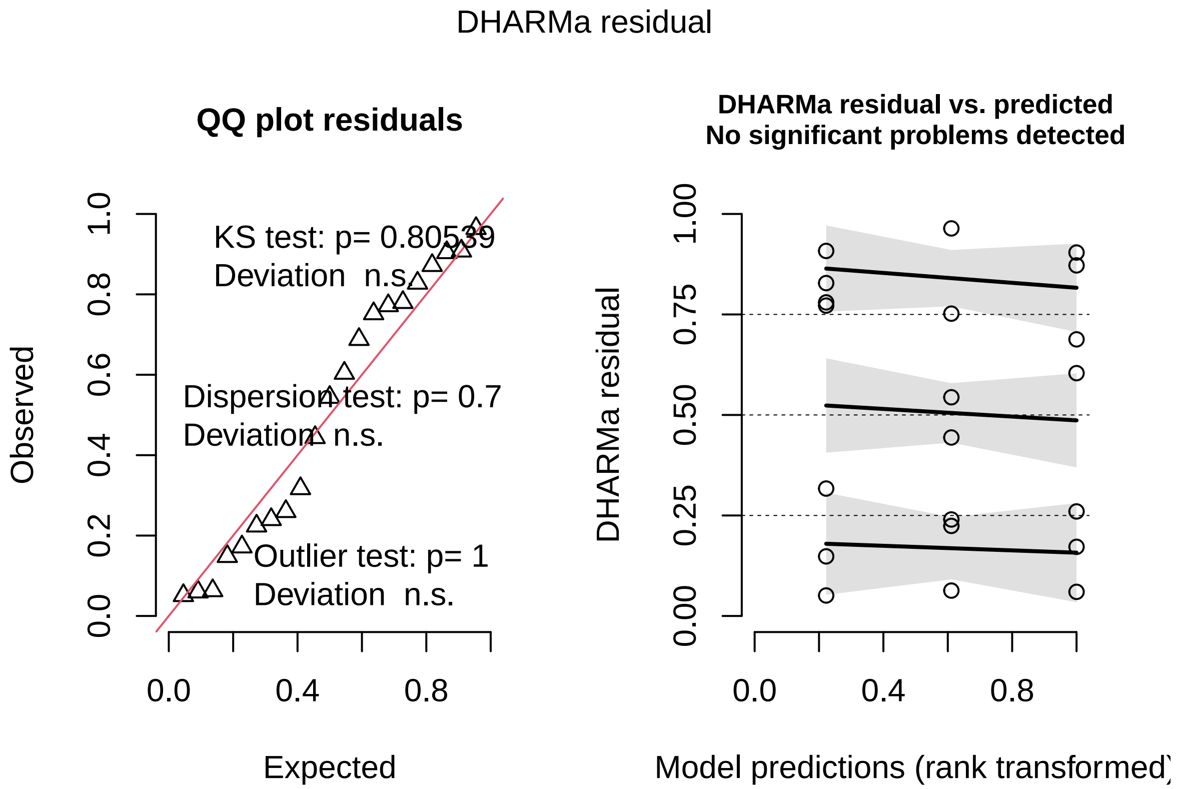
```
#Extract the estimated power parameter
p_est_tar$p.max #1.5, good
```

```
## [1] 1.47551
```

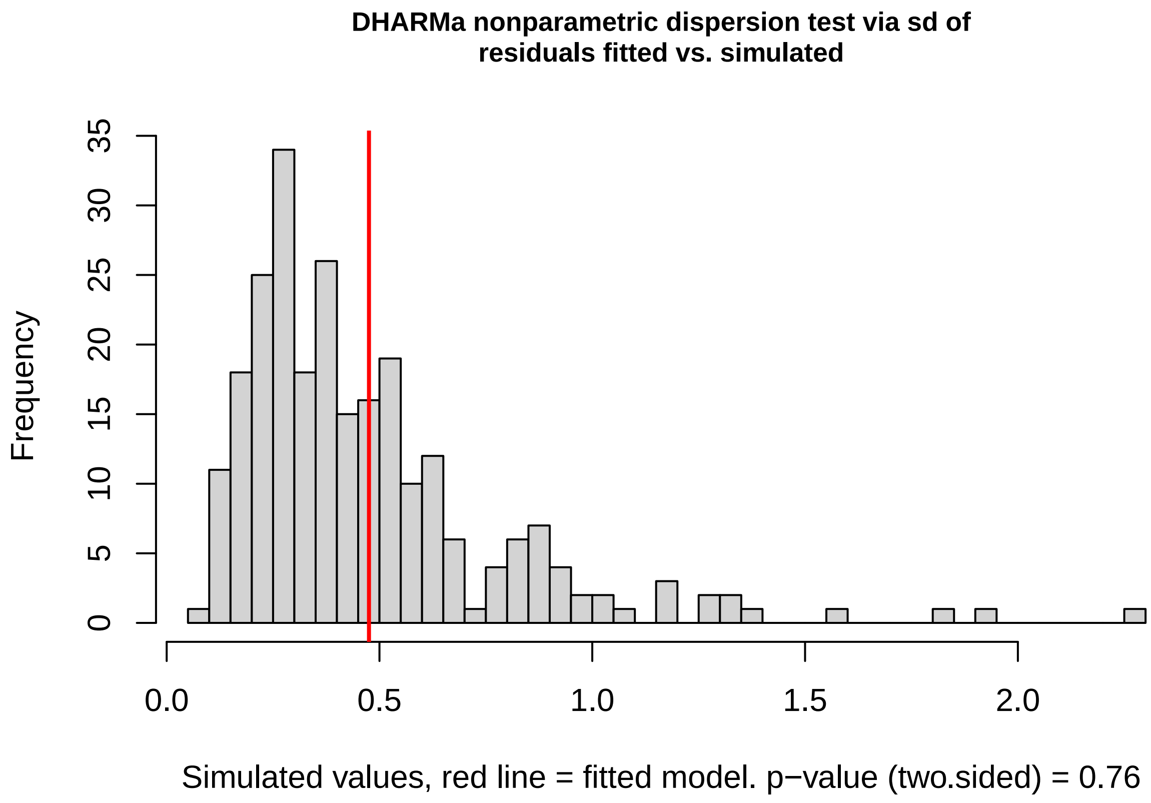
```
#Tweedie model with glmmTMB
glmmTMB_tar <- glmmTMB(copies_g~treatment,
                       family = tweedie,
                       data = subset_taraxacum)
summary(glmmTMB_tar)
```

```
##  Family: tweedie  ( log )
## Formula:          copies_g ~ treatment
## Data: subset_taraxacum
##
##      AIC      BIC   logLik deviance df.resid
##    414.9    420.1   -202.4    404.9       16
##
##
## Dispersion parameter for tweedie family ():  170
##
## Conditional model:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       10.3387     0.3425  30.183  < 2e-16 ***
## treatment240448    1.4048     0.4173   3.366 0.000762 ***
## treatment240720   -2.6522     0.7605  -3.487 0.000488 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(sim_tar<-simulateResiduals(glmmTMB_tar)) #Okay
```

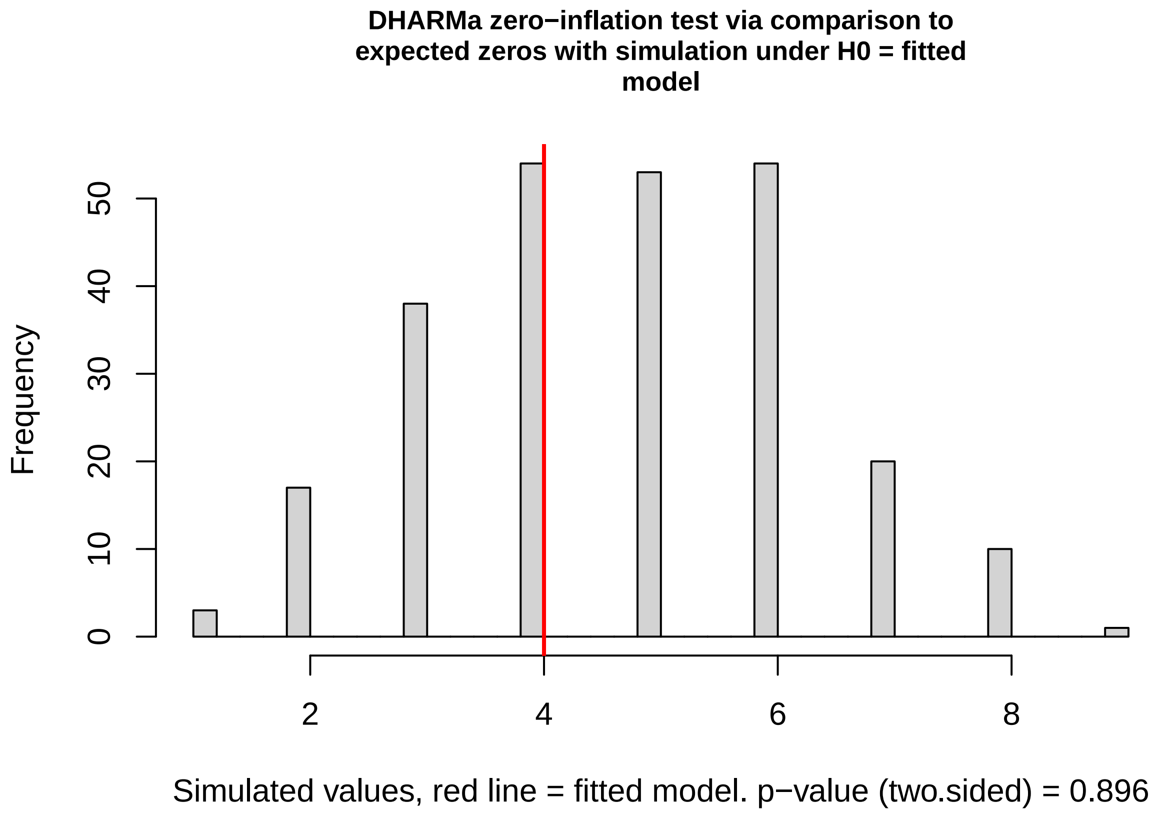


```
testDispersion(sim_tar) #okay
```



```
##
##  DHARMa nonparametric dispersion test via sd of residuals fitted vs.
##  simulated
##
## data:  simulationOutput
## dispersion = 1.0088, p-value = 0.76
## alternative hypothesis: two.sided
```

```
testZeroInflation(sim_tar) #Okay
```



```
##
##  DHARMa zero-inflation test via comparison to expected zeros with
##  simulation under H0 = fitted model
##
## data:  simulationOutput
## ratioObsSim = 0.84388, p-value = 0.896
## alternative hypothesis: two.sided
```

Fit of the tweeide models varied substantially between species groups. Not enough data for Gaillardia and Festuca to robustly estimate power parameter. Tweedie did not fit well for the additive model, interaction model, or Bromus.

Non-parametric approach

Use Kruskal-Wallis to check for treatment effects within each species group. Kruskal-Wallis was selected due to data distributions that do to uneven zero inflation between groups and a distribution shape that does not support a gamma model (due to presence of zeroes), a hurdle model (due to uneven distiribution of zeroes) nor a tweedie model (due to insufficient data and poor model fit). Because Kruskal-Wallis is non-parametric it ranks the data and does not assume a specific distibrution. This makes it ideal for non-normal data with a smaller sample size. However, because it is rank based, Kruskal- Wallis can only tell us whether two groups differ and not the specific magnitude. #absolute abundance

```
#Start with testing main effects
#TREATMENT
kruskal.test(copies_g~treatment, data=SUBSET)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  copies_g by treatment
## Kruskal-Wallis chi-squared = 37.718, df = 2, p-value = 6.452e-09
```

```
#HOST SPECIES
kruskal.test(copies_g~species, data=SUBSET)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  copies_g by species
## Kruskal-Wallis chi-squared = 7.2288, df = 3, p-value = 0.06495
```

```
#By species
#GAILLARDIA
kruskal.test(copies_g~treatment, data=subset_gaillardia) #Significant
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  copies_g by treatment
## Kruskal-Wallis chi-squared = 13.433, df = 2, p-value = 0.001211
```

```
#FESTUCA
kruskal.test(copies_g~treatment, data=subset_festuca) #Significant
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  copies_g by treatment
## Kruskal-Wallis chi-squared = 13.132, df = 2, p-value = 0.001408
```

```
#BROMUS
kruskal.test(copies_g~treatment, data=subset_bromus) #Not significant
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  copies_g by treatment
## Kruskal-Wallis chi-squared = 3.7893, df = 2, p-value = 0.1504
```

```
#TARAXACUM
kruskal.test(copies_g~treatment, data=subset_taraxacum) #Significant
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  copies_g by treatment
## Kruskal-Wallis chi-squared = 13.974, df = 2, p-value = 0.0009239
```

```r
#Kruskal-Wallis does not tell use which groups are different,
#a Dunn test can be used for post-hoc comparisons

#Used benjamin hochbberg adjustment to control for false discovery with
#multiple comparisons. Less conservative than holm or tukey.
#TREATMENT
dunnTest(copies_g ~ treatment, data = SUBSET, method = "holm")
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison

##   p-values adjusted with the Holm method.

##        Comparison         Z      P.unadj        P.adj
## 1 197198 - 240448 -2.819128 4.815428e-03 4.815428e-03
## 2 197198 - 240720  3.315648 9.143096e-04 1.828619e-03
## 3 240448 - 240720  6.134776 8.527922e-10 2.558377e-09
```

```r
#SPECIES
dunnTest(copies_g ~ species, data = SUBSET, method = "holm")
```

```
## Warning: species was coerced to a factor.

## Dunn (1964) Kruskal-Wallis multiple comparison
##   p-values adjusted with the Holm method.

##               Comparison          Z     P.unadj     P.adj
## 1       Bromus - Festuca  0.9905894 0.321886100 0.9656583
## 2    Bromus - Gaillardia -1.6053282 0.108421594 0.4336864
## 3   Festuca - Gaillardia -2.5959176 0.009433867 0.0566032
## 4     Bromus - Taraxacum -0.6975533 0.485456576 0.4854566
## 5    Festuca - Taraxacum -1.6881428 0.091383827 0.4569191
## 6 Gaillardia - Taraxacum  0.9077749 0.363997172 0.7279943
```

```r
#GAILLARDIA
dunnTest(copies_g ~ treatment, data = subset_gaillardia, method = "bh")
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison

##   p-values adjusted with the Benjamini-Hochberg method.

##        Comparison         Z      P.unadj        P.adj
## 1 197198 - 240448 -1.465436 0.1428021203 0.1428021203
## 2 197198 - 240720  2.176603 0.0295102143 0.0442653215
## 3 240448 - 240720  3.642038 0.0002704877 0.0008114632
```

```r
#FESTUCA
dunnTest(copies_g ~ treatment, data = subset_festuca, method = "bh")
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
##   p-values adjusted with the Benjamini-Hochberg method.
##        Comparison         Z      P.unadj       P.adj
## 1 197198 - 240448 -2.201053 0.0277322968 0.041598445
## 2 197198 - 240720  1.392503 0.1637702255 0.163770226
## 3 240448 - 240720  3.593555 0.0003261963 0.000978589
```

```
#BROMUS
dunnTest(copies_g ~ treatment, data = subset_bromus, method = "bh") #Npt signficiant
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
##   p-values adjusted with the Benjamini-Hochberg method.
```

```
##        Comparison          Z    P.unadj     P.adj
## 1 197198 - 240448 -0.4792812 0.63173858 0.6317386
## 2 197198 - 240720  1.3942726 0.16323529 0.2448529
## 3 240448 - 240720  1.8735539 0.06099194 0.1829758
```

```
#TARAXACUM
dunnTest(copies_g ~ treatment, data = subset_taraxacum, method = "bh")
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
##   p-values adjusted with the Benjamini-Hochberg method.
```

```
##        Comparison         Z      P.unadj        P.adj
## 1 197198 - 240448 -1.836577 0.0662723333 0.0662723333
## 2 197198 - 240720  1.901398 0.0572499551 0.0858749326
## 3 240448 - 240720  3.737975 0.0001855085 0.0005565256
```

```
#Test for whether species identity effects abundance
#197198
kruskal.test(copies_g~species, data=subset_197198) #Not significant
```

```
## 
##  Kruskal-Wallis rank sum test
## 
## data:  copies_g by species
## Kruskal-Wallis chi-squared = 4.4538, df = 3, p-value = 0.2165
```

```
#240448
kruskal.test(copies_g~species, data=subset_240448) #Significant
```

```
## 
##  Kruskal-Wallis rank sum test
## 
## data:  copies_g by species
## Kruskal-Wallis chi-squared = 10.126, df = 3, p-value = 0.01753
```

```
#240720
kruskal.test(copies_g~species, data=subset_240720) #Marginally significant
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  copies_g by species
## Kruskal-Wallis chi-squared = 9.8834, df = 3, p-value = 0.01958
```

```
#Follow up with Dunn test on 240448 and 240720
#Dunn 240448
dunnTest(copies_g ~ species, data = subset_240448, method = "bh")
```

```
## Warning: species was coerced to a factor.
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
##   p-values adjusted with the Benjamini-Hochberg method.
```

```
##               Comparison         Z    P.unadj      P.adj
## 1       Bromus - Festuca  0.048795 0.96108266 0.96108266
## 2    Bromus - Gaillardia -1.984330 0.04721905 0.07082857
## 3   Festuca - Gaillardia -2.033125 0.04203988 0.08407976
## 4     Bromus - Taraxacum -2.423485 0.01537238 0.04611714
## 5    Festuca - Taraxacum -2.472280 0.01342542 0.08055255
## 6 Gaillardia - Taraxacum -0.439155 0.66054921 0.79265905
```

```
#Marginally significant difference between Bromus and Taraxacum
#Dun 240720
dunnTest(copies_g ~ species, data = subset_240720, method = "bh")
```

```
## Warning: species was coerced to a factor.
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
##   p-values adjusted with the Benjamini-Hochberg method.
```

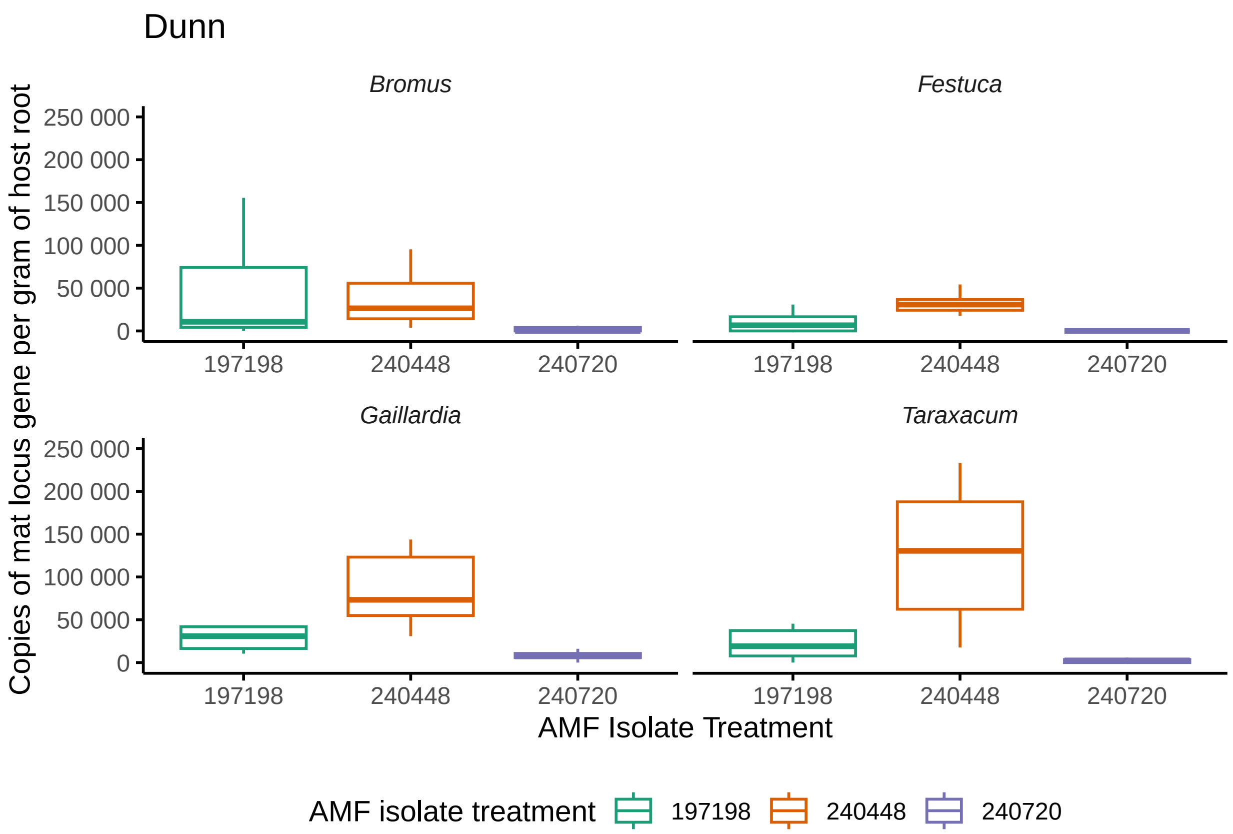
```
##               Comparison           Z     P.unadj      P.adj
## 1       Bromus - Festuca  1.07643407 0.281733171 0.42259976
## 2    Bromus - Gaillardia -1.99661159 0.045867393 0.09173479
## 3   Festuca - Gaillardia -3.07304566 0.002118861 0.01271316
## 4     Bromus - Taraxacum  0.01736184 0.986147952 0.98614795
## 5    Festuca - Taraxacum -1.05907223 0.289566885 0.34748026
## 6 Gaillardia - Taraxacum  2.01397342 0.044012324 0.13203697
```

```
#No differences after pairwise comparisons
```

Create visuals with significance stars

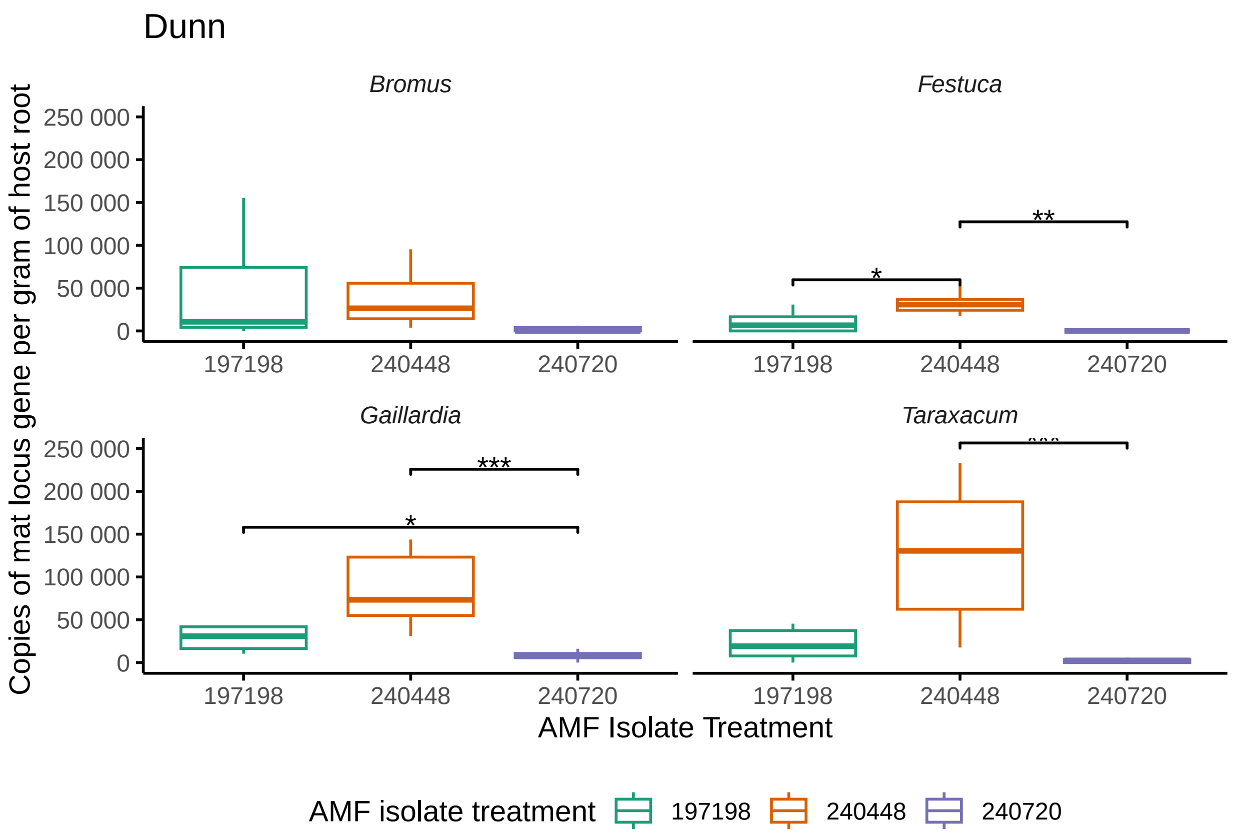
```
#Create visual with significance stars,

#Start by creating species and treatment labels

#Species labels (italicized for strip titles and x-axis)
```

```r
species_labels <- c(
  "Bromus" = expression(italic("B. tectorum")),
  "Gaillardia" = expression(italic("G. aristata")),
  "Festuca" = expression(italic("F. idahoensis")),
  "Taraxacum" = expression(italic("T. officinale"))
)
#Treatment labels for facet strip and legend
treatment_labels <- c(
  "197198" = "DAOM 197198",
  "240448" = "DAOM 240448",
  "240720" = "DAOM 240720"
)

#Create a base plot for treatment within species
plot_copies_sig <- ggplot(SUBSET, aes(x = treatment, y = copies_g, color = treatment)) +
  geom_boxplot(alpha = 0.5, outlier.shape = NA) +
  facet_wrap(~species, scales = "free_x", labeller = labeller(species = species_labels)) +
  labs(
    title = "Dunn",
    x = "AMF Isolate Treatment",
    y = "Copies of mat locus gene per gram of host root",
    color = "AMF isolate treatment"
  ) +

  scale_y_continuous(limits = c(0, 1.1 * max(SUBSET$copies_g, na.rm = FALSE)),
                     labels = scales::label_number()) +# disables scientific notation
  coord_cartesian(ylim = c(0, 250000)) +
  scale_colour_brewer(palette = "Dark2") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_blank(),
    axis.line = element_line(color = "black"),
    axis.ticks = element_line(color = "black"),
    strip.text = element_text(face = "italic"),
    legend.position = "bottom"
  )

print(plot_copies_sig)
```



```
#Calculate y for each facet
y_max <- SUBSET %>%
  group_by(species) %>%
  summarise(y_max = max(copies_g, na.rm = TRUE) * 1.1)

global_max_y <- max(y_max$y_max, na.rm = TRUE)

#Create significance annotation table for treatment within species
sig_annotations <- data.frame(
  species = c(
    "Gaillardia", "Gaillardia", "Gaillardia",
    "Festuca", "Festuca", "Festuca",
    "Bromus", "Bromus", "Bromus",
    "Taraxacum", "Taraxacum", "Taraxacum"
  ),
  group1 = c(
    "197198", "197198", "240448",
    "197198", "197198", "240448",
    "197198", "197198", "240448",
    "197198", "197198", "240448"
  ),
  group2 = c(
    "240448", "240720", "240720",
    "240448", "240720", "240720",
    "240448", "240720", "240720",
    "240448", "240720", "240720"
```

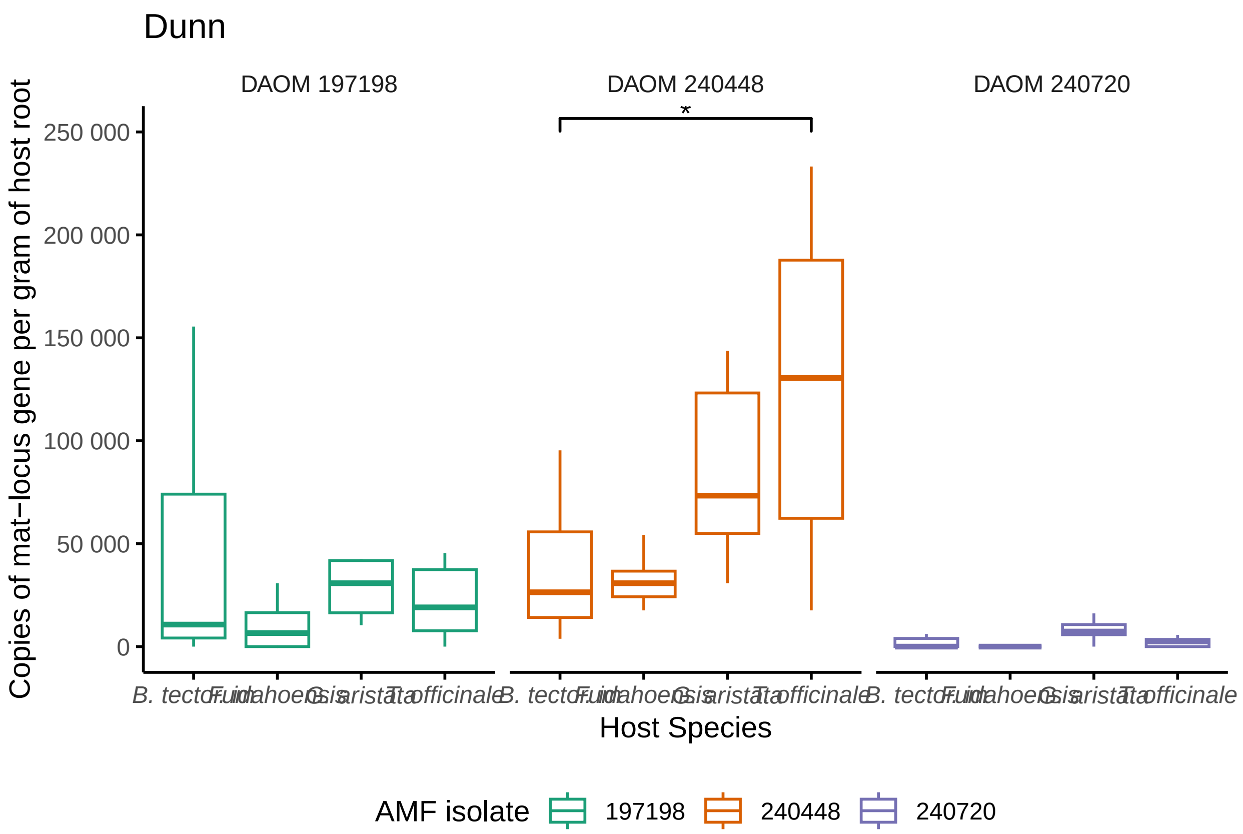
```r
  ),
  p.adj = c(
    0.1678, 0.0282, 0.0006,  # Gaillardia: 197198-240448, 197198-240720, 240448-240720
    0.0416, 0.1638, 0.0010,  # Festuca:    same order
    0.6317, 0.2449, 0.1830,  # Bromus:     all non-significant
    0.0663, 0.0859, 0.0006   # Taraxacum:  marginal + one significant
  )
)

#Add significance labels to plot
sig_annotations <- sig_annotations %>%
  mutate(
    p.label = case_when(
      p.adj < 0.001 ~ "***",
      p.adj < 0.01  ~ "**",
      p.adj < 0.05  ~ "*"
    )
  ) %>%
  filter(!is.na(p.label)) %>%
  left_join(y_max, by = "species") %>%
  group_by(species) %>%
  mutate(offset = row_number() - 1,
         y.position = y_max + offset * (global_max_y * 0.1)) %>%
  ungroup()

plot_copies_sig <- plot_copies_sig +
  stat_pvalue_manual(
    data = sig_annotations,
    label = "p.label",
    xmin = "group1",
    xmax = "group2",
    y.position = "y.position",
    tip.length = 0.01,
    bracket.size = 0.5
  )

print(plot_copies_sig)
```



```r
#HOST SPECIES WITHIN TREATMENT
treatment_labels <- c(
  `197198` = "DAOM 197198",
  `240448` = "DAOM 240448",
  `240720` = "DAOM 240720"
)

facet_wrap(~treatment, scales = "free_x",
           labeller = labeller(treatment = treatment_labels))
```

```
## <ggproto object: Class FacetWrap, Facet, gg>
##     compute_layout: function
##     draw_back: function
##     draw_front: function
##     draw_labels: function
##     draw_panels: function
##     finish_data: function
##     init_scales: function
##     map_data: function
##     params: list
##     setup_data: function
##     setup_params: function
##     shrink: TRUE
##     train_scales: function
##     vars: function
##     super:  <ggproto object: Class FacetWrap, Facet, gg>
```

```r
#CReate visual with significance stars,

#Create significance annotation table for species comparisons
sig_annotations_treat <- data.frame(
  treatment = c(
    rep("240448", 6),
    rep("240720", 6)
  ),
  group1 = c(
    "Bromus", "Bromus", "Festuca", "Bromus", "Festuca", "Gaillardia",
    "Bromus", "Bromus", "Festuca", "Bromus", "Festuca", "Gaillardia"
  ),
  group2 = c(
    "Festuca", "Gaillardia", "Gaillardia", "Taraxacum", "Taraxacum", "Taraxacum",
    "Festuca", "Gaillardia", "Gaillardia", "Taraxacum", "Taraxacum", "Taraxacum"
  ),
  p.adj = c(
    0.9611, 0.0708, 0.0841, 0.0461, 0.0806, 0.7927,    # from dunnTest on 240448
    0.3022, 0.3928, 0.0674, 0.9033, 0.3075, 0.3297     # from dunnTest on 240720
  )
)

#Add significance labels
sig_annotations_treat <- sig_annotations_treat %>%
  mutate(
    p.label = case_when(
      p.adj < 0.001 ~ "***",
      p.adj < 0.01  ~ "**",
      p.adj < 0.05  ~ "*"
    )
  ) %>%
  filter(!is.na(p.label))

#Get max y-value per treatment for y.position calculation
y_max <- SUBSET %>%
  group_by(treatment) %>%
  summarise(y_max = max(copies_g, na.rm = TRUE) * 1.1)

global_max_y <- max(y_max$y_max, na.rm = TRUE)

#Add staggered y positions
sig_annotations_treat <- sig_annotations_treat %>%
  left_join(y_max, by = "treatment") %>%
  group_by(treatment) %>%
  mutate(offset = row_number() - 1) %>%
  mutate(
    y.position = y_max + offset * (global_max_y * 0.1)
  ) %>%
  ungroup()

#Create base plot
plot_copies_sig_treat <- ggplot(SUBSET, aes(x = species, y = copies_g, color = treatment)) +
  geom_boxplot(alpha = 0.5, outlier.shape = NA, outlier.alpha = 1) +
  facet_wrap(~treatment, scales = "free_x", labeller = labeller(treatment = treatment_labels)) +
  labs(
    title = "Dunn",
    x = "Host Species",
    y = "Copies of mat-locus gene per gram of host root",
    color = "AMF isolate"
  ) +
  scale_x_discrete(labels = species_labels) +
  scale_y_continuous(labels = scales::label_number()) +
  coord_cartesian(ylim = c(0, 250000)) +
  scale_color_manual(
    values = RColorBrewer::brewer.pal(4, "Dark2"),
    labels = species_labels,
    guide = guide_legend(override.aes = list(alpha = 1)),
    aesthetics = "color"
  ) +
  stat_pvalue_manual(
    data = sig_annotations_treat,
    label = "p.label",
    xmin = "group1",
    xmax = "group2",
    y.position = "y.position",
    tip.length = 0.01,
    bracket.size = 0.5
  ) +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    panel.grid = element_blank(),
    axis.line = element_line(color = "black"),
    axis.ticks = element_line(color = "black")
  )

print(plot_copies_sig_treat)
```


Dunn
DAOM 197198
DAOM 240448
DAOM 240720
*
Copies of mat−locus gene per gram of host root
250 000
200 000
150 000
100 000
50 000
0
B. tectorum
F. idahoensis
O. aristata
T. officinale
Host Species
AMF isolate
197198
240448
240720