Data Analysis

Joan Moreaux and Meredith Miller

22/11/2021

Loading library and data

```r
library(performance)
library(DHARMa)
library(mgcv)
library(fitdistrplus)
library(goft)
library(gamlss)
library(FSA)
library(fGarch)
library(LambertW)
library(ordinal)
library(cowplot)
library(here)
library(plotrix)
library(patchwork)
library(tidyverse)
```

```r
health_data <- read.csv("Big_Data_Green.csv")
heat_data <- read.csv("Heat_Data.csv")
```

Health Data

Includes feeding time, PAM measurements, base measurements, and symbiont density.

```r
# Organizing the data, adding average PAM, average base diameter, and symbiont density
health_data <- health_data %>%
  mutate(Fv_Fm_1 = as.numeric(Fv_Fm_1), PAM_avg = ((Fv_Fm_1 + Fv_Fm_2 + Fv_Fm_3)/3)) %>%
  mutate(Base_Diameter_mm = (Base_Width + Base_Length + Base_Diagonal)/3) %>%
  mutate(Weight_Tentacle_mg = Weight_Tentacle_g*1000) %>%
  mutate(Green_Density = (Green_Cells/1)*(1/0.1)*(1/0.001)*(0.5/Weight_Tentacle_mg)) %>%
  mutate(Dino_Density = (Dino_Cells/1)*(1/0.1)*(1/0.001)*(0.5/Weight_Tentacle_mg)) %>%
  mutate(MI_Green = (Div_Green/Green_Cells)*100,
         MI_Dino = (Div_Dino/Dino_Cells)*100) %>% # Bates et al. 2010
  mutate(Bucket = as.factor(Bucket), Treatment = as.factor(Treatment),
         Date = as.factor(Date), Event = as.factor(Event),
         Species_ID = as.factor(Species_ID), Field_Site = as.factor(Field_Site),
         Event_True = as.factor(Event_True)) %>%
  filter(Species_ID != "G32B" & Species_ID != "G33B") %>%
  mutate(Treatment = fct_relevel(Treatment, "Control", "25C", "30C"),
         Event = fct_relevel("Acclimation", "Pre-heat", "Post-heat", "Recovery"))
cols_for_heat = c("25C" = "sienna2", "30C" = "brown3", "Control" = "azure4")
```

Symbiont Data

PAM Threshold

Determining a mimum threshold for aposymbiotic anemones that we will remove from the PAM data. From the data bellow, we decide that the threshold for PAM will be >1000 cells per mg (symbiont density).

```
Green_Threshold <- health_data %>%
  select(-c(Event, Field_Site, Acclimation_Period, Base_Width, Base_Length,
            Base_Diagonal, Base_Diameter_mm, Nb_Tentacles, Weight_Total_g,
            Weight_Tube_g, Weight_Tentacle_g, Div_Dino, Dino_Cells, Feeding_Time_Min,
            Feeding_Time, Feeding_Start, Feeding_Stop, Photo_ID, Removed, MI_Dino,
            Fv_Fm_1, Fv_Fm_2, Fv_Fm_3, PAM_avg, Event_True, Dino_Density)) %>%
  drop_na(Green_Density) %>%
  drop_na(Time_Point) %>%
  filter(Green_Density != 0)

Dino_Threshold <- health_data %>%
  select(-c(Event, Field_Site, Acclimation_Period, Base_Width, Base_Length,
            Base_Diagonal, Base_Diameter_mm, Nb_Tentacles, Weight_Total_g,
            Weight_Tube_g, Weight_Tentacle_g, Green_Cells, Div_Green, Feeding_Time_Min,
            Feeding_Time, Feeding_Start, Feeding_Stop, Photo_ID, Removed, MI_Green,
            Fv_Fm_1, Fv_Fm_2, Fv_Fm_3,
            PAM_avg, Event_True, Green_Density)) %>%
  drop_na(Dino_Density) %>%
  drop_na(Time_Point) %>%
  filter(Dino_Density != 0)
```

Density Data

Data for models of symbiont density for both zoochlorellae and zooxanthellae. It filters out unwanted columns and anemones without symbiont count.

```
# Organizing the data
Green_Symbionts <- health_data %>%
  select(-c(Event, Base_Width, Base_Length, Base_Diagonal, Base_Diameter_mm,
            Nb_Tentacles, Weight_Total_g, Weight_Tube_g, Weight_Tentacle_g,
            Green_Cells, Dino_Cells, Feeding_Time_Min, Feeding_Time, Feeding_Start,
            Feeding_Stop, Photo_ID, Removed, Fv_Fm_1, Fv_Fm_2, Fv_Fm_3, PAM_avg,
            Event_True)) %>%
  filter(Date == "11/5/2021" | Date == "11/9/2021" | Date == "11/13/2021") %>%
  mutate(Date = fct_relevel(Date, "11/5/2021", "11/9/2021", "11/13/2021")) %>%
  mutate(Green_Density_Log = log(Green_Density)) %>%
  drop_na(Green_Density) %>%
  drop_na(Dino_Density) %>%
  drop_na(MI_Green) %>%
  drop_na(MI_Dino) %>%
  filter(Green_Density != 0) %>%
  mutate(Acclimation_Period = as.factor(Acclimation_Period)) %>%
```

```r
  group_by(Treatment, Date) %>%
  mutate(Mean_Green_Log = mean(Green_Density_Log),
         SE_Green_Log = std.error(Green_Density_Log))

Dino_Symbionts <- health_data %>%
  select(-c(Event, Base_Width, Base_Length, Base_Diagonal, Base_Diameter_mm,
            Nb_Tentacles, Weight_Total_g, Weight_Tube_g, Weight_Tentacle_g,
            Green_Cells, Dino_Cells, Feeding_Time_Min, Feeding_Time,
            Feeding_Start, Feeding_Stop, Photo_ID, Removed, Fv_Fm_1, Fv_Fm_2,
            Fv_Fm_3, PAM_avg, Event_True)) %>%
  filter(Date == "11/5/2021" | Date == "11/9/2021" | Date == "11/13/2021") %>%
  mutate(Date = fct_relevel(Date, "11/5/2021", "11/9/2021", "11/13/2021")) %>%
  drop_na(Green_Density) %>%
  drop_na(Dino_Density) %>%
  drop_na(MI_Green) %>%
  drop_na(MI_Dino) %>%
  filter(Dino_Density != 0) %>%
  mutate(Acclimation_Period = as.factor(Acclimation_Period))
```

Checking Normality and Equal Variance:

```r
# Green
shapiro.test(Green_Symbionts$Green_Density)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Green_Symbionts$Green_Density
## W = 0.64647, p-value = 3.82e-10
```

```r
bartlett.test(Green_Density ~ Treatment, Green_Symbionts)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Green_Density by Treatment
## Bartlett's K-squared = 23.091, df = 2, p-value = 9.681e-06
```

```r
# Dino
shapiro.test(Dino_Symbionts$Dino_Density)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Dino_Symbionts$Dino_Density
## W = 0.80442, p-value = 5.198e-07
```

```r
bartlett.test(Dino_Density ~ Treatment, Dino_Symbionts)
```

```
##
```

```
##  Bartlett test of homogeneity of variances
##
## data:  Dino_Density by Treatment
## Bartlett's K-squared = 3.8169, df = 2, p-value = 0.1483
```

All p-values are <0.05 and therefore the distributions do not meet assumptions of normality and equal variance.

Checking Distributions of Symbiont Data: Determining if the data fits other distributions.

```
#Greens
exp_test(Green_Symbionts$Green_Density)          #p-value < 0.05
```

```
##
##  Test for exponentiality based on a transformation to uniformity
##
## data:  Green_Symbionts$Green_Density
## T = 4.719, p-value < 2.2e-16
```

```
gamma_test(Green_Symbionts$Green_Density)        #p = 0.123
```

```
##
##  Test of fit for the Gamma distribution
##
## data:  Green_Symbionts$Green_Density
## V = -0.12047, p-value = 0.9321
```

```
lnorm_test(Green_Symbionts$Green_Density)        #p-value < 0.05
```

```
##
##  Test for the lognormal distribution based on a transformation to
##  normality
##
## data:  Green_Symbionts$Green_Density
## p-value = 0.006495
```

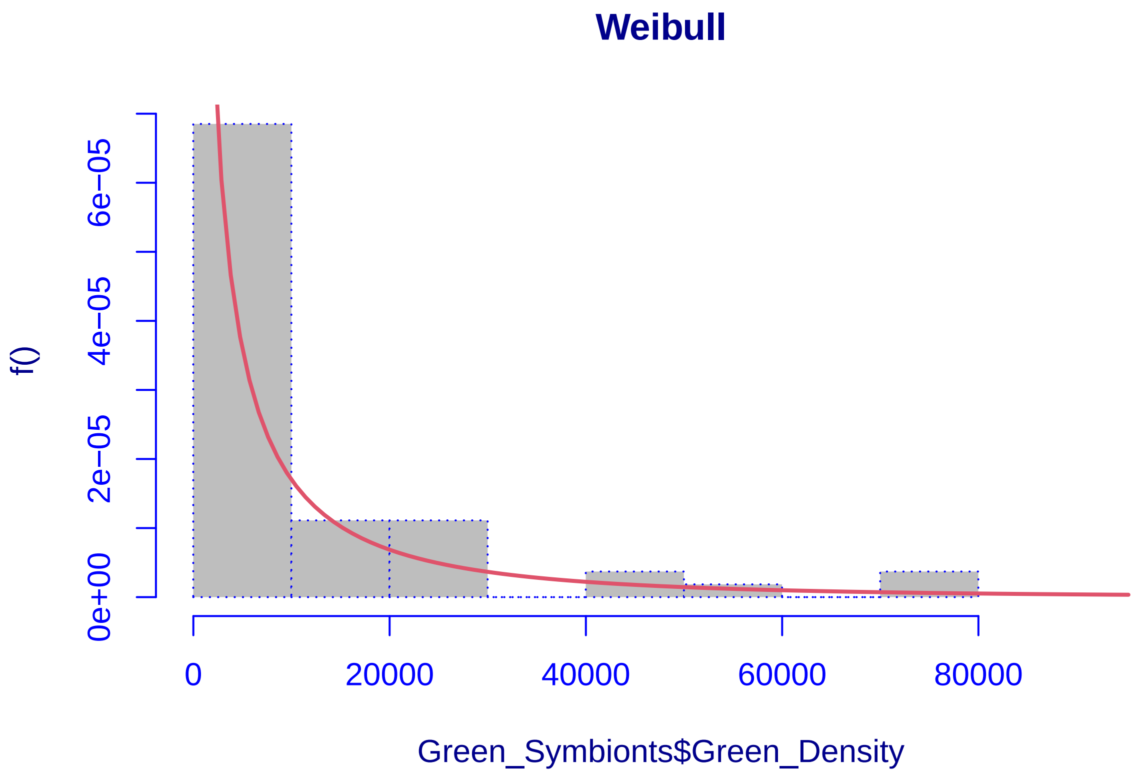
```
normal_test(Green_Symbionts$Green_Density)       #p-value < 0.05
```

```
##
##  Correlation test for normality
##
## data:  Green_Symbionts$Green_Density
## R = 0.89717, p-value = 1.019e-06
## alternative hypothesis: Green_Symbionts$Green_Density does not follow a normal distribution.
```

```
weibull_test(Green_Symbionts$Green_Density)      #p = 0.44
```

```
##
##  Test for the Weibull distribution
##
## data:  Green_Symbionts$Green_Density
## p-value = 0.048
```

```
#Use Weibull since p value is larger than gamma
Green_Weibull <- histDist(Green_Symbionts$Green_Density, "WEI",
                          density = F, main = "Weibull")
```



```
#Dinos
exp_test(Dino_Symbionts$Dino_Density)           #p-value > 0.05
```

```
##
##  Test for exponentiality based on a transformation to uniformity
##
## data:  Dino_Symbionts$Dino_Density
## T = 2.6262, p-value < 2.2e-16
```

```
gamma_test(Dino_Symbionts$Dino_Density)         #p = 0.6709
```

```
##
##  Test of fit for the Gamma distribution
##
## data:  Dino_Symbionts$Dino_Density
## V = -1.131, p-value = 0.4239
```

```
lnorm_test(Dino_Symbionts$Dino_Density)         #p-value < 0.05
```

```
##
##  Test for the lognormal distribution based on a transformation to
##  normality
##
## data:  Dino_Symbionts$Dino_Density
## p-value = 0.003319
```

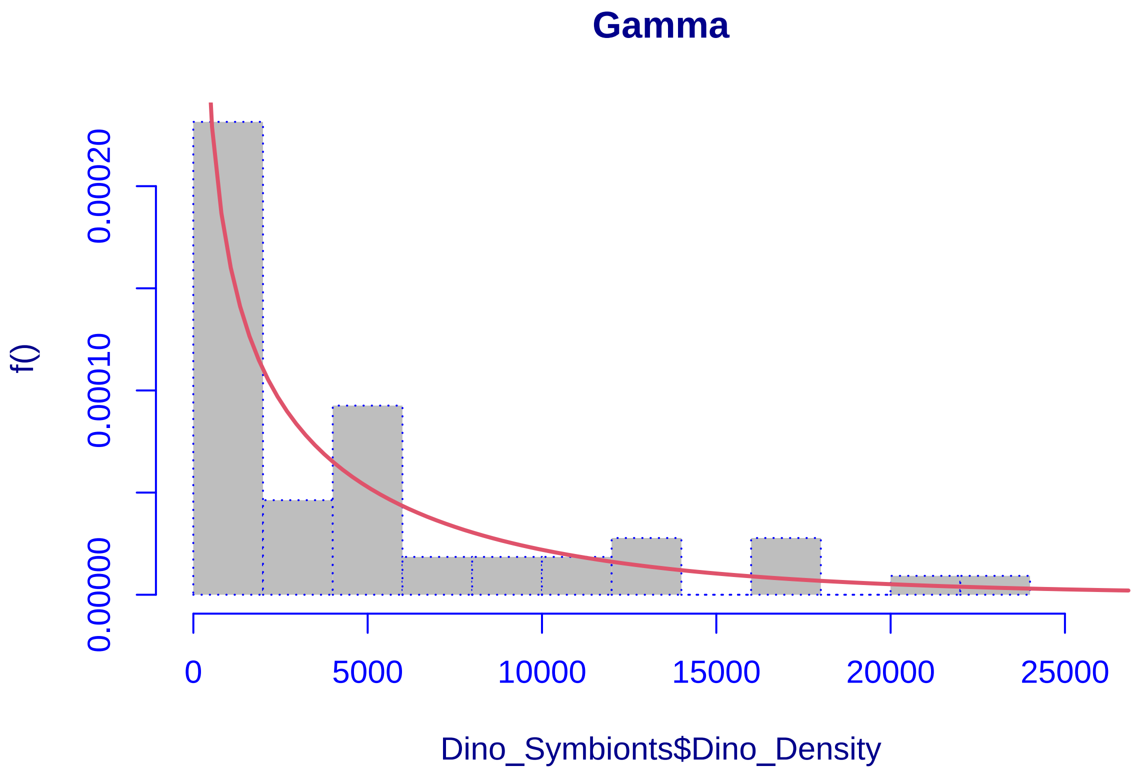
```r
normal_test(Dino_Symbionts$Dino_Density)        #p-value < 0.05
```

```
##
##  Correlation test for normality
##
## data:  Dino_Symbionts$Dino_Density
## R = 0.95707, p-value = 7.874e-05
## alternative hypothesis: Dino_Symbionts$Dino_Density does not follow a normal distribution.
```

```r
weibull_test(Dino_Symbionts$Dino_Density)       #p = 0.774
```

```
##
##  Test for the Weibull distribution
##
## data:  Dino_Symbionts$Dino_Density
## p-value = 0.946
```

```r
#Use Gamma since p value is larger than weibull
Dino_Gamma <- histDist(Dino_Symbionts$Dino_Density, "GA",
                       density = F, main = "Gamma")
```



Running Models on Symbiont Data: We chose to use our Weibull distribution for green density and Gamma distribution for dino density. Here we run our models.

```
#Green
Green_Symbionts$orderTreatment = ordered(Green_Symbionts$Treatment,
                                         levels = c("Control", "25C", "30C"))

Green_Weibull_model <- gamlss(formula = Green_Density ~ Date*Treatment +
                                random(Species_ID) + random(Field_Site) +
                                random(Acclimation_Period),
                              family = WEI(), data = Green_Symbionts,
                              control = gamlss.control(n.cyc = 10))
```

```
## GAMLSS-RS iteration 1: Global Deviance = 927.0726
## GAMLSS-RS iteration 2: Global Deviance = 910.9055
## GAMLSS-RS iteration 3: Global Deviance = 908.0624
## GAMLSS-RS iteration 4: Global Deviance = 907.6404
## GAMLSS-RS iteration 5: Global Deviance = 907.5682
## GAMLSS-RS iteration 6: Global Deviance = 907.5528
## GAMLSS-RS iteration 7: Global Deviance = 907.549
## GAMLSS-RS iteration 8: Global Deviance = 907.5483
```

```
summary(Green_Weibull_model) # significance in the 30 degree treatment
```

```
## ******************************************************************
## Family:  c("WEI", "Weibull") 
## 
## Call:  gamlss(formula = Green_Density ~ Date * Treatment +  
##     random(Species_ID) + random(Field_Site) + random(Acclimation_Period),  
##     family = WEI(), data = Green_Symbionts, control = gamlss.control(n.cyc = 10)) 
## 
## 
## Fitting method: RS() 
## 
## ------------------------------------------------------------------
## Mu link function:  log
## Mu Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                   7.02648    0.20520  34.242  < 2e-16 ***
## Date11/9/2021                -0.17523    0.27804  -0.630   0.5354    
## Date11/13/2021                0.28886    0.28856   1.001   0.3283    
## Treatment25C                 -0.12147    0.32291  -0.376   0.7106    
## Treatment30C                  1.47524    0.26344   5.600 1.53e-05 ***
## Date11/9/2021:Treatment25C    0.05102    0.45057   0.113   0.9109    
## Date11/13/2021:Treatment25C -0.83396    0.44295  -1.883   0.0738 .  
## Date11/9/2021:Treatment30C    0.13530    0.38302   0.353   0.7275    
## Date11/13/2021:Treatment30C  0.32660    0.38314   0.852   0.4037    
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## 
## ------------------------------------------------------------------
## Sigma link function:  log
## Sigma Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   0.6939     0.1161   5.977 6.48e-06 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## 
## ------------------------------------------------------------------
## NOTE: Additive smoothing terms exist in the formulas: 
##  i) Std. Error for smoothers are for the linear effect only. 
## ii) Std. Error for the linear terms maybe are not accurate. 
## ------------------------------------------------------------------
## No. of observations in the fit:  54 
## Degrees of Freedom for the fit:  33.20121
##       Residual Deg. of Freedom:  20.79879 
##                       at cycle:  8 
##  
## Global Deviance:     907.5483 
##             AIC:     973.9507 
##             SBC:     1039.987 
## ******************************************************************
```

```
#Dino
Dino_Symbionts$orderTreatment = ordered(Dino_Symbionts$Treatment,
                                        levels = c("Control", "25C", "30C"))

Dino_Gamma_model <- gamlss(formula = Dino_Density ~ Date*Treatment +
                             random(Species_ID) + random(Field_Site) +
                             random(Acclimation_Period),
                           family = GA(), data = Dino_Symbionts,
                           control = gamlss.control(n.cyc = 3))
```

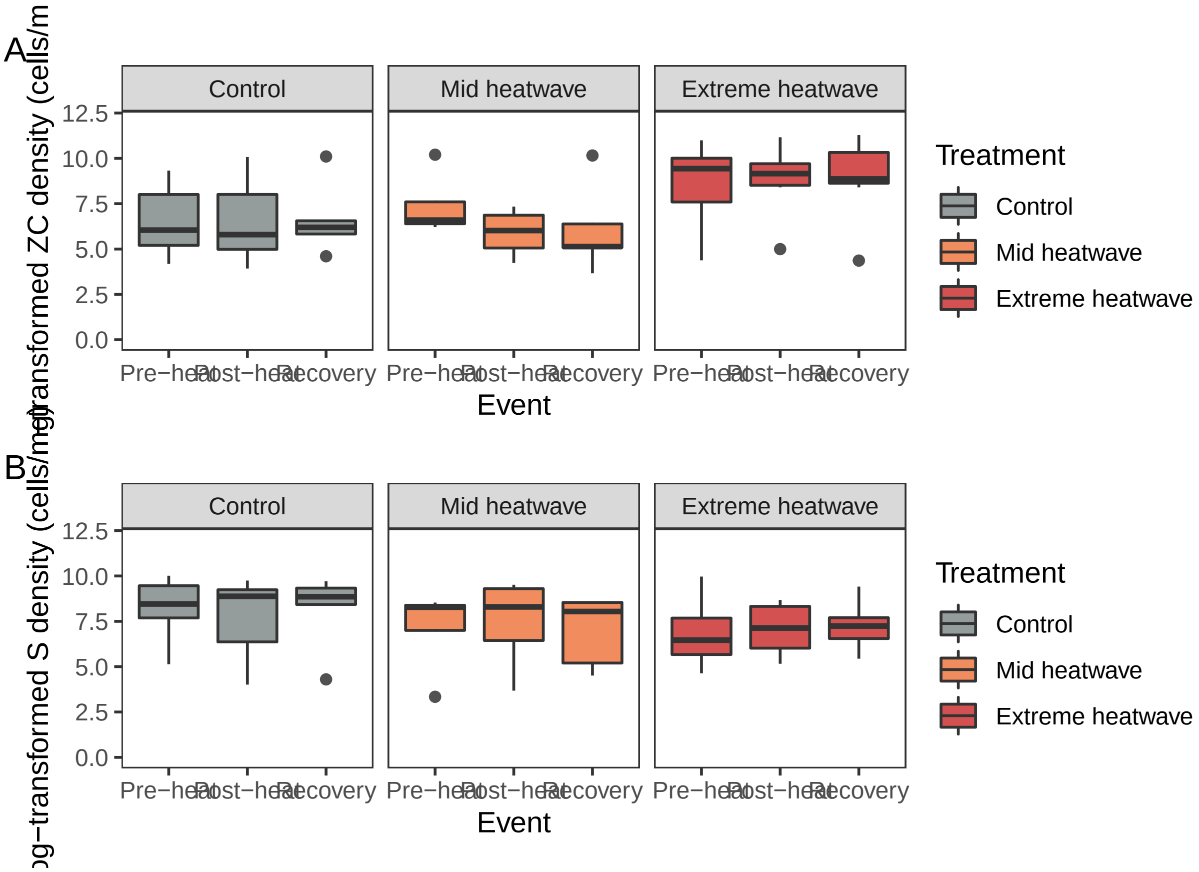
```
## GAMLSS-RS iteration 1: Global Deviance = 908.9065
## GAMLSS-RS iteration 2: Global Deviance = 908.8828
## GAMLSS-RS iteration 3: Global Deviance = 908.8819
```

```
summary(Dino_Gamma_model) # significance in both the 25 and 30 degree treatment
```

```
## ******************************************************************
## Family:  c("GA", "Gamma")
##
## Call:  gamlss(formula = Dino_Density ~ Date * Treatment +
##     random(Species_ID) + random(Field_Site) + random(Acclimation_Period),
##     family = GA(), data = Dino_Symbionts, control = gamlss.control(n.cyc = 3))
##
##
## Fitting method: RS()
##
## ------------------------------------------------------------------
## Mu link function:  log
## Mu Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  8.06079    0.25027  32.208  < 2e-16 ***
## Date11/9/2021               -0.16934    0.34106  -0.496  0.62449
## Date11/13/2021              -0.23477    0.35394  -0.663  0.51405
## Treatment25C                -1.32556    0.39574  -3.350  0.00291 **
## Treatment30C                -0.74465    0.32308  -2.305  0.03104 *
## Date11/9/2021:Treatment25C   0.43494    0.55161   0.788  0.43886
## Date11/13/2021:Treatment25C  1.03874    0.54261   1.914  0.06874 .
## Date11/9/2021:Treatment30C  -0.23333    0.46976  -0.497  0.62435
## Date11/13/2021:Treatment30C  0.03936    0.46977   0.084  0.93399
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ------------------------------------------------------------------
## Sigma link function:  log
## Sigma Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.48968    0.09084   -5.39 2.09e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ------------------------------------------------------------------
## NOTE: Additive smoothing terms exist in the formulas:
##  i) Std. Error for smoothers are for the linear effect only.
## ii) Std. Error for the linear terms maybe are not accurate.
## ------------------------------------------------------------------
## No. of observations in the fit:  54
## Degrees of Freedom for the fit:  32.10774
##       Residual Deg. of Freedom:  21.89226
##                        at cycle:  3
##
## Global Deviance:     908.8819
##             AIC:     973.0974
##             SBC:     1036.959
## ******************************************************************
```

Plots for symbiont data:

```
treatment_labels = c("Control", "Mid heatwave", "Extreme heatwave")
names(treatment_labels) = (c("Control", "25C", "30C"))
green_plot <- ggplot(Green_Symbionts, aes(x = Date, y = log(Green_Density),
                                          fill = Treatment)) +
  geom_boxplot(alpha = 0.85) +
  xlab("Event") +
  ylab("Log-transformed ZC density (cells/mg)") +
  facet_grid(. ~ Treatment,
             labeller = labeller(Treatment = treatment_labels)) +
  scale_fill_manual(limits = c("Control", "25C", "30C"),
                    labels = c("Control", "Mid heatwave", "Extreme heatwave"),
                    values = cols_for_heat) +
  theme_test()

green_plot <- green_plot + ylim(0,12) +
  scale_x_discrete(labels=c("11/5/2021" = "Pre-heat", "11/9/2021" = "Post-heat",
                            "11/13/2021" = "Recovery"))

dino_plot <- ggplot(Dino_Symbionts, aes(x = Date, y = log(Dino_Density),
                                        fill = Treatment)) +
  geom_boxplot(alpha = 0.85) +
  xlab("Event") +
  ylab("Log-transformed S density (cells/mg)") +
  facet_grid(. ~ Treatment,
             labeller = labeller(Treatment = treatment_labels)) +
  scale_fill_manual(limits = c("Control", "25C", "30C"),
                    labels = c("Control", "Mid heatwave", "Extreme heatwave"),
                    values = cols_for_heat) +
  theme_test()
dino_plot <- dino_plot + ylim(0,12) +
  scale_x_discrete(labels=c("11/5/2021" = "Pre-heat", "11/9/2021" = "Post-heat",
                            "11/13/2021" = "Recovery"))

density_plot = (green_plot + dino_plot) + plot_layout(ncol=1)
density_plot <- density_plot + plot_annotation(tag_levels = ’A’)
density_plot
```



```r
ggsave(here("./images/density_plot.png"), density_plot, width = 10, height = 9)
```

Mitotic Index Data

MI is the amount of cell division in the symbionts, calculated as a %.

```r
# Organizing the data
MI_Data <- health_data %>%
  select(-c(Event, Base_Width, Base_Length, Base_Diagonal, Base_Diameter_mm,
            Nb_Tentacles, Weight_Total_g, Weight_Tube_g, Weight_Tentacle_g,
            Green_Cells, Dino_Cells, Feeding_Time_Min, Feeding_Time, Feeding_Start,
            Feeding_Stop, Photo_ID, Removed, Fv_Fm_1, Fv_Fm_2, Fv_Fm_3, PAM_avg,
            Event_True)) %>%
  filter(Date == "11/5/2021" | Date == "11/9/2021" | Date == "11/13/2021") %>%
  mutate(Date = fct_relevel(Date, "11/5/2021", "11/9/2021", "11/13/2021")) %>%
  drop_na(Green_Density) %>%
  drop_na(Dino_Density) %>%
  drop_na(MI_Green) %>%
  drop_na(MI_Dino) %>%
  mutate(Acclimation_Period = as.factor(Acclimation_Period)) %>%
  mutate(MI_Dino_scaled = MI_Dino + 0.01) %>%
  mutate(MI_Green_scaled = MI_Green + 0.01)
```

Checking Normality and Equal Variance:

```r
#Green
shapiro.test(MI_Data$Green_Density)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  MI_Data$Green_Density
## W = 0.64647, p-value = 3.82e-10
```

```r
bartlett.test(Green_Density ~ Treatment, MI_Data)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Green_Density by Treatment
## Bartlett's K-squared = 23.091, df = 2, p-value = 9.681e-06
```

```r
#Dino
shapiro.test(MI_Data$Dino_Density)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  MI_Data$Dino_Density
## W = 0.80442, p-value = 5.198e-07
```

```r
bartlett.test(Dino_Density ~ Treatment, MI_Data) # p-value = 0.1483
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Dino_Density by Treatment
## Bartlett's K-squared = 3.8169, df = 2, p-value = 0.1483
```

ZC (green) symbiont density does not meet assumptions of normality or equal variance, while ZC (dino) does not meet assumption of normality but meets the assumption of equal variance. We will run Dunn and Kruskal Wallis tests on the MI data to test for significant differences.

Dunn and Kruskal Wallis Test on MI Data:

```r
kruskal.test(MI_Green ~ Treatment, data = MI_Data) # p = 0.009
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  MI_Green by Treatment
## Kruskal-Wallis chi-squared = 9.492, df = 2, p-value = 0.008686
```

```r
dunnTest(MI_Green ~ Treatment, data = MI_Data) # sig. difference between 25-30
```

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

##   p-values adjusted with the Holm method.

##      Comparison         Z     P.unadj       P.adj
## 1     25C - 30C -3.036394 0.002394262 0.007182786
## 2 25C - Control -1.441840 0.149347449 0.149347449
## 3 30C - Control  1.734539 0.082822412 0.165644824
```

```r
kruskal.test(MI_Dino ~ Treatment, data = MI_Data) # not significant
```

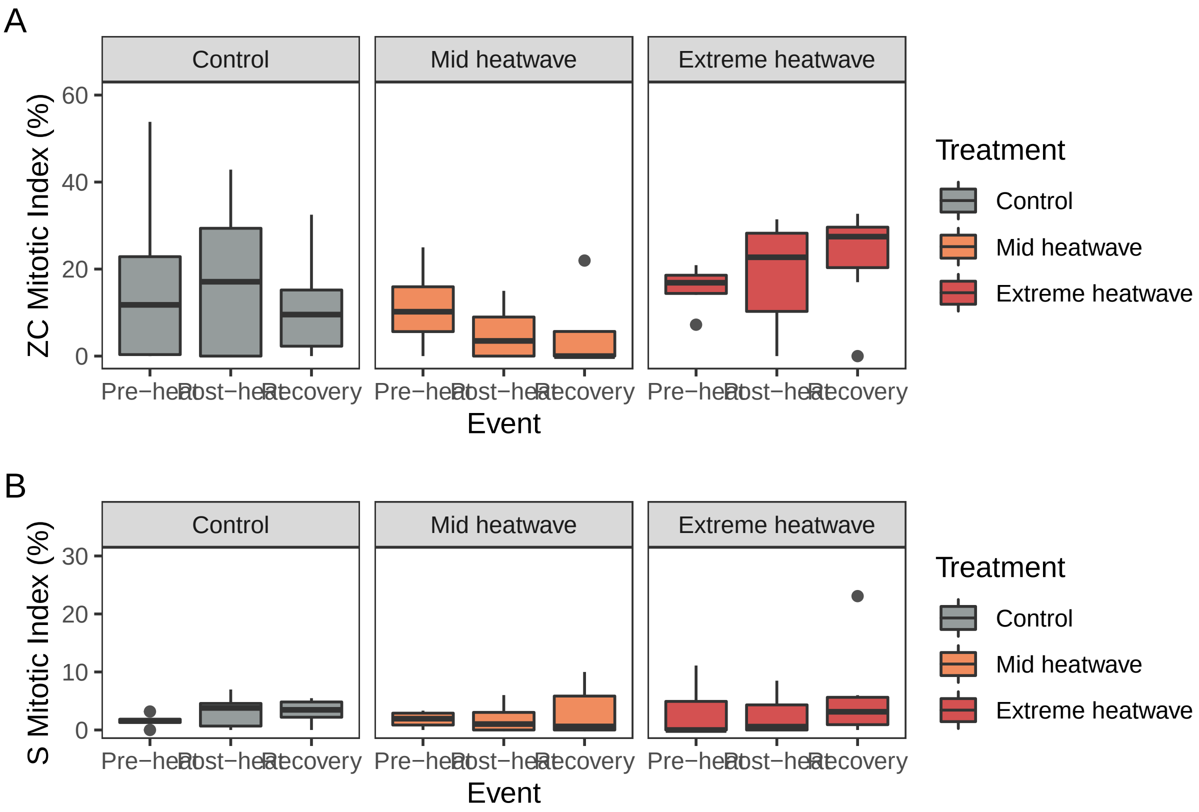
```
## 
##  Kruskal-Wallis rank sum test
## 
## data:  MI_Dino by Treatment
## Kruskal-Wallis chi-squared = 0.51372, df = 2, p-value = 0.7735
```

```r
dunnTest(MI_Dino ~ Treatment, data = MI_Data) # not significant
```

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

##      Comparison           Z   P.unadj     P.adj
## 1     25C - 30C -0.07275603 0.9420003 0.9420003
## 2 25C - Control -0.60896106 0.5425503 1.0000000
## 3 30C - Control -0.61859436 0.5361836 1.0000000
```

Plot for MI data:

```r
green_MI_plot <- ggplot(Green_Symbionts, aes(x = Date, y = MI_Green,
                                             fill = Treatment)) +
  geom_boxplot(alpha = 0.85) +
  xlab("Event") +
  ylab("ZC Mitotic Index (%)") +
  facet_grid(. ~ Treatment,
             labeller = labeller(Treatment = treatment_labels)) +
  scale_fill_manual(limits = c("Control", "25C", "30C"),
                    labels = c("Control", "Mid heatwave", "Extreme heatwave"),
                    values = cols_for_heat) +
  theme_test()
green_MI_plot <- green_MI_plot + ylim(0,60) +
  scale_x_discrete(labels=c("11/5/2021" = "Pre-heat", "11/9/2021" = "Post-heat",
                            "11/13/2021" = "Recovery"))

dino_MI_plot <- ggplot(Dino_Symbionts, aes(x = Date, y = MI_Dino,
                                           fill = Treatment)) +
  geom_boxplot(alpha = 0.85) +
  xlab("Event") +
  ylab("S Mitotic Index (%)") +
  facet_grid(. ~ Treatment,
             labeller = labeller(Treatment = treatment_labels)) +
  scale_fill_manual(limits = c("Control", "25C", "30C"),
                    labels = c("Control", "Mid heatwave", "Extreme heatwave"),
                    values = cols_for_heat) +
  theme_test()
dino_MI_plot <- dino_MI_plot + ylim(0,30) +
  scale_x_discrete(labels=c("11/5/2021" = "Pre-heat", "11/9/2021" = "Post-heat",
                            "11/13/2021" = "Recovery"))

MI_plot = (green_MI_plot + dino_MI_plot) + plot_layout(ncol=1, heights=c(1.5,1))
MI_plot <- MI_plot + plot_annotation(tag_levels = 'A')
MI_plot
```



```
ggsave(here("./images/MI_plot.png"), MI_plot, width = 10, height = 9)
```

PAM data

We filter out aposymbiotic anemones.

```
# Organizing the data
PAM_individuals <- health_data %>%
  filter(Green_Density > 1000 | Dino_Density > 1000) %>%
```

```r
  distinct(Species_ID)

PAM_data <- health_data %>%
  select(-c(Base_Width, Base_Length, Base_Diagonal, Base_Diameter_mm, Nb_Tentacles,
            Weight_Total_g, Weight_Tube_g, Weight_Tentacle_g, Green_Cells, Dino_Cells,
            Feeding_Time_Min, Feeding_Time, Feeding_Start, Feeding_Stop, Photo_ID,
            Removed, Div_Green, Div_Dino, MI_Green, MI_Dino, Acclimation_Period,
            Green_Density, Dino_Density)) %>%
  drop_na(Time_Point) %>%
  drop_na(PAM_avg) %>%
  group_by(Treatment, Date) %>%
  mutate(Mean_PAM = mean(PAM_avg), SE_PAM = std.error(PAM_avg))

filtered_PAM_data = PAM_individuals %>%
  left_join(PAM_data) %>%
  filter(Date == "11/6/2021" | Date == "11/9/2021" | Date == "11/13/2021") %>%
  filter(Time_Point != 4) %>%
  mutate(Date = fct_relevel(Date, "11/6/2021", "11/9/2021", "11/13/2021"))
```

Checking Normality and Equal Variance:

```r
shapiro.test(filtered_PAM_data$PAM_avg)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  filtered_PAM_data$PAM_avg
## W = 0.84753, p-value = 3.465e-08
```

```r
bartlett.test(PAM_avg ~ Treatment, data = filtered_PAM_data)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  PAM_avg by Treatment
## Bartlett's K-squared = 21.408, df = 2, p-value = 2.245e-05
```

Our data does not meet the assumptions of normality and equal variance ($p < 0.05$) and therefore we will use Dunn and Kruskal Wallis tests to test for significant differences.

```r
kruskal.test(PAM_avg ~ Treatment, data = filtered_PAM_data) # significance
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  PAM_avg by Treatment
## Kruskal-Wallis chi-squared = 8.0406, df = 2, p-value = 0.01795
```

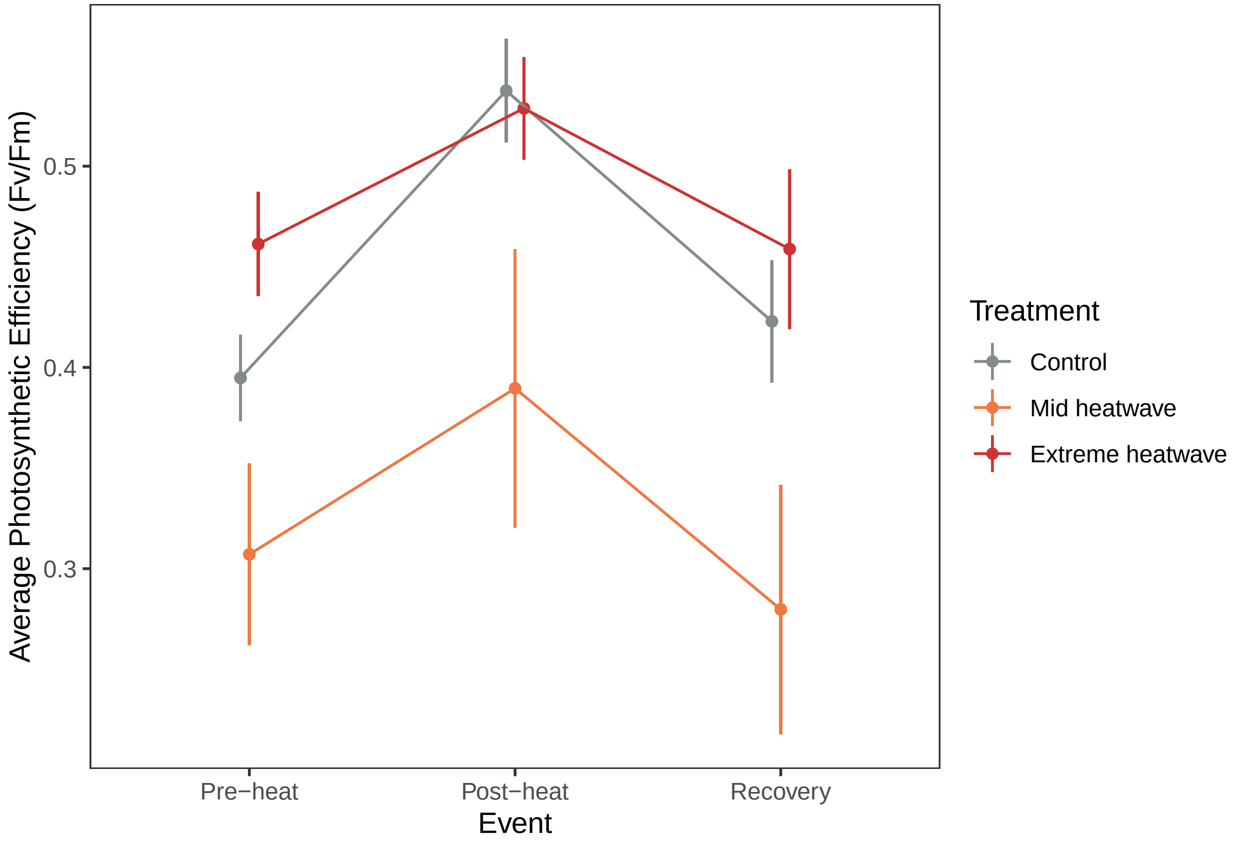
```r
dunnTest(PAM_avg ~ Treatment, data = filtered_PAM_data)
```

```
##      Comparison          Z     P.unadj      P.adj
## 1     25C - 30C -2.6353649 0.008404688 0.02521406
## 2 25C - Control -0.6723531 0.501358906 0.50135891
## 3 30C - Control  2.1391425 0.032424133 0.06484827
```

```
# significance only between 25 and 30
```

Plot for PAM:

```
PAM_over_time <- ggplot(data = filtered_PAM_data, aes(x=Date, y = Mean_PAM,
                                                      colour = Treatment,
                                                      group = Treatment)) +
  geom_point(position = position_dodge(width = 0.1)) +
  geom_linerange(aes(ymin = Mean_PAM - SE_PAM,
                     ymax = Mean_PAM + SE_PAM), position = position_dodge(width = 0.1)) +
  geom_line(position = position_dodge(width = 0.1)) +
  labs(x="Event",
       y = "Average Photosynthetic Efficiency (Fv/Fm)") +
  scale_colour_manual(limits = c("Control", "25C", "30C"),
                    labels = c("Control", "Mid heatwave", "Extreme heatwave"),
                    values = cols_for_heat) +
  theme_test()
PAM_over_time <- PAM_over_time +
  scale_x_discrete(labels=c("11/6/2021" = "Pre-heat", "11/9/2021" = "Post-heat",
                            "11/13/2021" = "Recovery"))
PAM_over_time
```



```
ggsave(here("./images/PAM_over_time.png"), PAM_over_time)
```

Size Data

Measuring the size of the base as a proxy for weight.

```
# Organizing the data
Size_Data <- health_data %>%
  select(-c(Base_Width, Base_Length, Base_Diagonal, Nb_Tentacles, Weight_Total_g,
            Weight_Tube_g, Weight_Tentacle_g, Weight_Tentacle_mg, Green_Cells,
            Dino_Cells, Feeding_Time_Min, Feeding_Time, Feeding_Start, Feeding_Stop,
            Photo_ID, Removed, Fv_Fm_1, Fv_Fm_2, Fv_Fm_3, PAM_avg, Green_Density,
            Dino_Density, MI_Green, MI_Dino, Div_Green, Div_Dino)) %>%
  filter(Date == "11/5/2021" | Date == "11/9/2021" | Date == "11/13/2021") %>%
  mutate(Date = fct_relevel(Date, "11/5/2021", "11/9/2021", "11/13/2021"),
        (Treatment = fct_relevel(Treatment, "Control", "25C", "30C"))) %>%
  drop_na(Base_Diameter_mm) %>%
  mutate(Acclimation_Period = as.factor(Acclimation_Period)) %>%
  group_by(Treatment, Date) %>%
  mutate(Mean_Size = mean(Base_Diameter_mm), SE_Size = std.error(Base_Diameter_mm))
```

Checking for Normality and Equal Variance:

```
shapiro.test(Size_Data$Base_Diameter_mm)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Size_Data$Base_Diameter_mm
## W = 0.94517, p-value = 0.0002429
```

```
bartlett.test(Base_Diameter_mm ~ Treatment, data = Size_Data)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Base_Diameter_mm by Treatment
## Bartlett's K-squared = 3.8061, df = 2, p-value = 0.1491
```

The distributions do not meet assumptions of normality but does meet the assumption of equal variance.

Checking Distributions of Symbiont Data: Determining if the data fits other distributions.

```
exp_test(Size_Data$Base_Diameter_mm)             #p-value < 0.05
```

```
##
##  Test for exponentiality based on a transformation to uniformity
##
## data:  Size_Data$Base_Diameter_mm
## T = -11.277, p-value < 2.2e-16
```

```r
gamma_test(Size_Data$Base_Diameter_mm)          #p = 0.1874
```

```
##
##  Test of fit for the Gamma distribution
##
## data:  Size_Data$Base_Diameter_mm
## V = 1.8642, p-value = 0.1874
```

```r
lnorm_test(Size_Data$Base_Diameter_mm)          #p = 0.4117
```

```
##
##  Test for the lognormal distribution based on a transformation to
##  normality
##
## data:  Size_Data$Base_Diameter_mm
## p-value = 0.4117
```

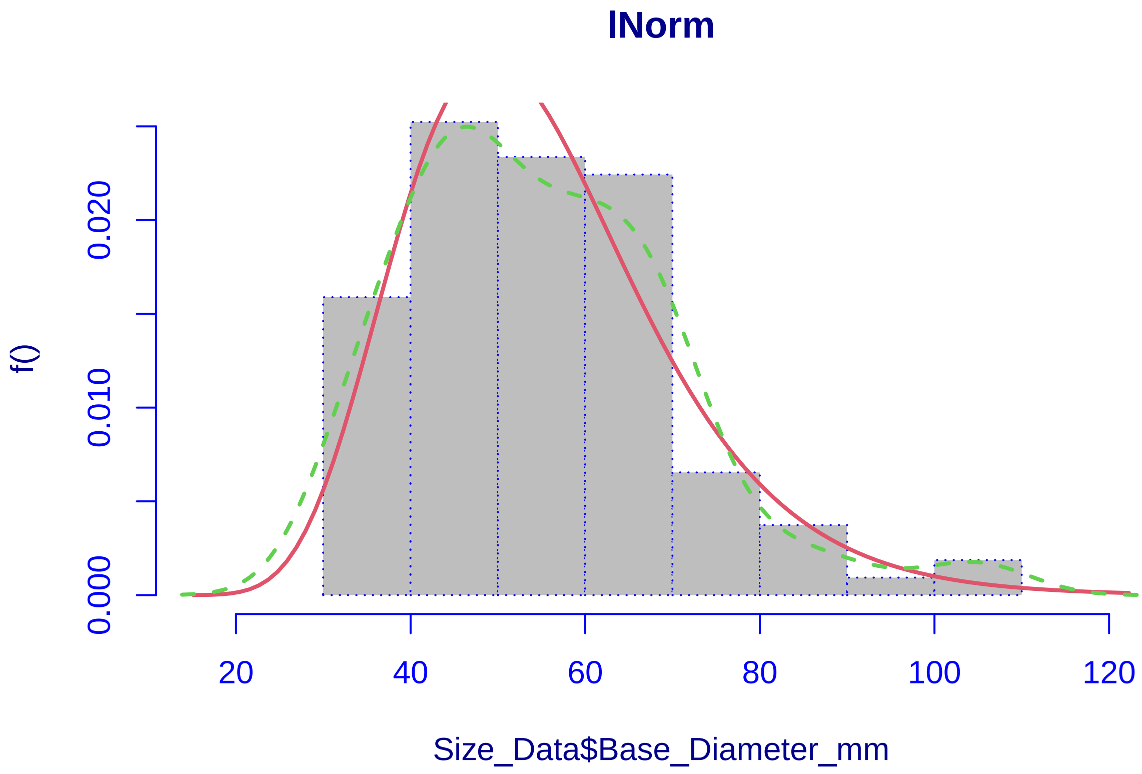
```r
normal_test(Size_Data$Base_Diameter_mm)         #p-value < 0.05
```

```
##
##  Correlation test for normality
##
## data:  Size_Data$Base_Diameter_mm
## R = 0.98889, p-value = 0.001424
## alternative hypothesis: Size_Data$Base_Diameter_mm does not follow a normal distribution.
```

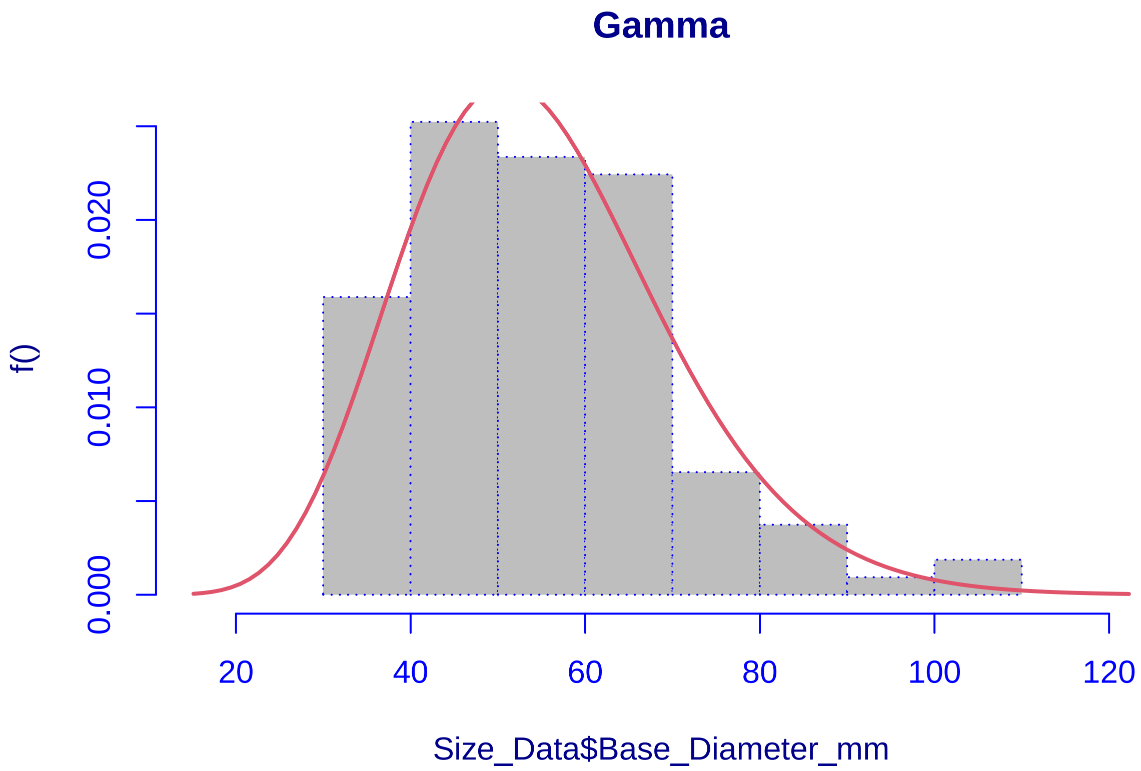
```r
weibull_test(Size_Data$Base_Diameter_mm)        #p-value < 0.05
```

```
##
##  Test for the Weibull distribution
##
## data:  Size_Data$Base_Diameter_mm
## p-value < 2.2e-16
```

```r
# Run lnorm since it has the higest p value
Size_lnorm <- histDist(Size_Data$Base_Diameter_mm, "LOGNO", density = T, main = "lNorm")

# Choosing the best distribution by comparing AIC values
Size_lnorm <- histDist(Size_Data$Base_Diameter_mm, "LOGNO", density = T, main = "lNorm")
```



```
#AIC = 880.625
Feeding_Gamma <- histDist(Size_Data$Base_Diameter_mm, "GA", density = F, main = "Gamma")
```



```
#AIC = 883.161
```

Running Models: We are going to run a lNorm model on our size data because the p value is larger and AIC value is smaller.

```
Size_Data$orderTreatment = ordered(Size_Data$Treatment, levels = c("Control", "25C", "30C"))

Size_lNorm_model <- gamlss(formula = Base_Diameter_mm ~ Date*Treatment +
                             random(Species_ID) + random(Field_Site) +
                             random(Acclimation_Period),
                           family = LOGNO(), data = Size_Data,
                           control = gamlss.control(n.cyc = 4))
```

```
## GAMLSS-RS iteration 1: Global Deviance = 566.2608
## GAMLSS-RS iteration 2: Global Deviance = 566.254
## GAMLSS-RS iteration 3: Global Deviance = 566.2585
## GAMLSS-RS iteration 4: Global Deviance = 566.2585
```

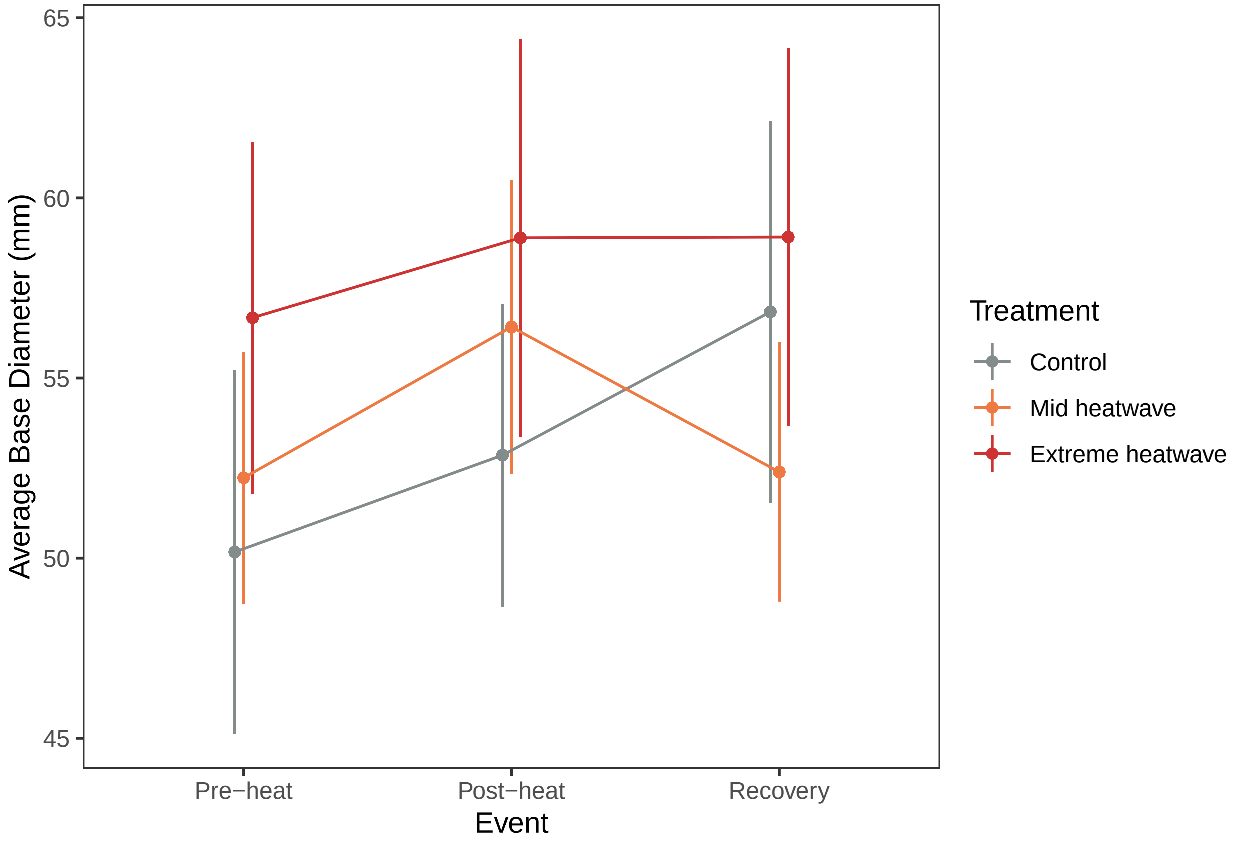
```
summary(Size_lNorm_model) # significant in 30
```

```
## ******************************************************************
## Family:  c("LOGNO", "Log Normal")
##
## Call:  gamlss(formula = Base_Diameter_mm ~ Date * Treatment +
##     random(Species_ID) + random(Field_Site) + random(Acclimation_Period),
##     family = LOGNO(), data = Size_Data, control = gamlss.control(n.cyc = 4))
##
## Fitting method: RS()
##
## -------------------------------------------------------------------
## Mu link function:  identity
## Mu Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  3.891115   0.019396 200.618  < 2e-16 ***
## Date11/9/2021                0.034371   0.026852   1.280 0.205309
## Date11/13/2021               0.093877   0.026852   3.496 0.000878 ***
## Treatment25C                 0.039133   0.026852   1.457 0.150054
## Treatment30C                 0.109203   0.026852   4.067 0.000137 ***
## Date11/9/2021:Treatment25C   0.038217   0.037560   1.018 0.312859
## Date11/13/2021:Treatment25C -0.092427   0.037560  -2.461 0.016657 *
## Date11/9/2021:Treatment30C  -0.004105   0.037560  -0.109 0.913320
## Date11/13/2021:Treatment30C -0.055277   0.037560  -1.472 0.146148
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## -------------------------------------------------------------------
## Sigma link function:  log
## Sigma Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.74376    0.06836  -40.14   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## -------------------------------------------------------------------
## NOTE: Additive smoothing terms exist in the formulas:
##  i) Std. Error for smoothers are for the linear effect only.
## ii) Std. Error for the linear terms maybe are not accurate.
## -------------------------------------------------------------------
## No. of observations in the fit:  107
## Degrees of Freedom for the fit:  44.92919
##       Residual Deg. of Freedom:  62.07081
##                       at cycle:  4
##
## Global Deviance:     566.2585
##             AIC:     656.1169
##             SBC:     776.2049
## *******************************************************************
```

The control and 25 did have significant values, but the 30 did not.

Plot for Size:

```
size_plot <- ggplot(data = Size_Data, aes(x=Date, y = Mean_Size,
                                          colour = Treatment, group = Treatment)) +
  geom_point(position = position_dodge(width = 0.1)) +
  geom_linerange(aes(ymin = Mean_Size - SE_Size, ymax = Mean_Size + SE_Size),
                 position = position_dodge(width = 0.1)) +
  geom_line(position = position_dodge(width = 0.1)) +
  labs(x="Event",
       y = "Average Base Diameter (mm)") +
  scale_colour_manual(limits = c("Control", "25C", "30C"),
                      labels = c("Control", "Mid heatwave", "Extreme heatwave"),
                      values = cols_for_heat) +
  theme_test()

size_plot <- size_plot +
  scale_x_discrete(labels=c("11/5/2021" = "Pre-heat", "11/9/2021" = "Post-heat",
                            "11/13/2021" = "Recovery"))
size_plot
```



```
ggsave(here("./images/size_plot.png"), size_plot)
```

```
## Saving 6.5 x 4.5 in image
```

Feeding Data

This is the feeding time before and after heatwaves.

```
# Organizing the data
Feeding_Data <- health_data %>%
  select(-c(Base_Width, Base_Length, Base_Diagonal, Nb_Tentacles, Weight_Total_g,
            Weight_Tube_g, Weight_Tentacle_g, Weight_Tentacle_mg, Green_Cells,
            Dino_Cells, Base_Diameter_mm, Photo_ID, Removed, Fv_Fm_1, Fv_Fm_2,
            Fv_Fm_3, PAM_avg, Green_Density, Dino_Density, MI_Green, MI_Dino,
            Div_Green, Div_Dino)) %>%
  filter(Date == "11/5/2021" | Date == "11/9/2021") %>%
  mutate(Date = fct_relevel(Date, "11/5/2021", "11/9/2021")) %>%
  drop_na(Feeding_Time_Min) %>%
  mutate(Acclimation_Period = as.factor(Acclimation_Period))
```

Checking for Normality and Equal Variance:

```
shapiro.test(Feeding_Data$Feeding_Time_Min)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Feeding_Data$Feeding_Time_Min
## W = 0.62812, p-value = 1.973e-05
```

```
bartlett.test(Feeding_Time_Min ~ Treatment, data = Feeding_Data)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Feeding_Time_Min by Treatment
## Bartlett's K-squared = 9.0434, df = 2, p-value = 0.01087
```

Distribution is not normal but meets the assumption of equal variance. We will check if it fits other distributions.

Checking Distributions: Determining if the data fits other distributions.

```
exp_test(Feeding_Data$Feeding_Time_Min)            #p-value < 0.05
```

```
##
##  Test for exponentiality based on a transformation to uniformity
##
## data:  Feeding_Data$Feeding_Time_Min
## T = 2.048, p-value = 0.004
```

```
gamma_test(Feeding_Data$Feeding_Time_Min)          #p = 0.3717
```

```
##
##  Test of fit for the Gamma distribution
##
## data:  Feeding_Data$Feeding_Time_Min
## V = 0.86695, p-value = 0.5399
```

```
lnorm_test(Feeding_Data$Feeding_Time_Min)          #p = 0.6937
```

```
##
##  Test for the lognormal distribution based on a transformation to
##  normality
##
## data:  Feeding_Data$Feeding_Time_Min
## p-value = 0.7746
```

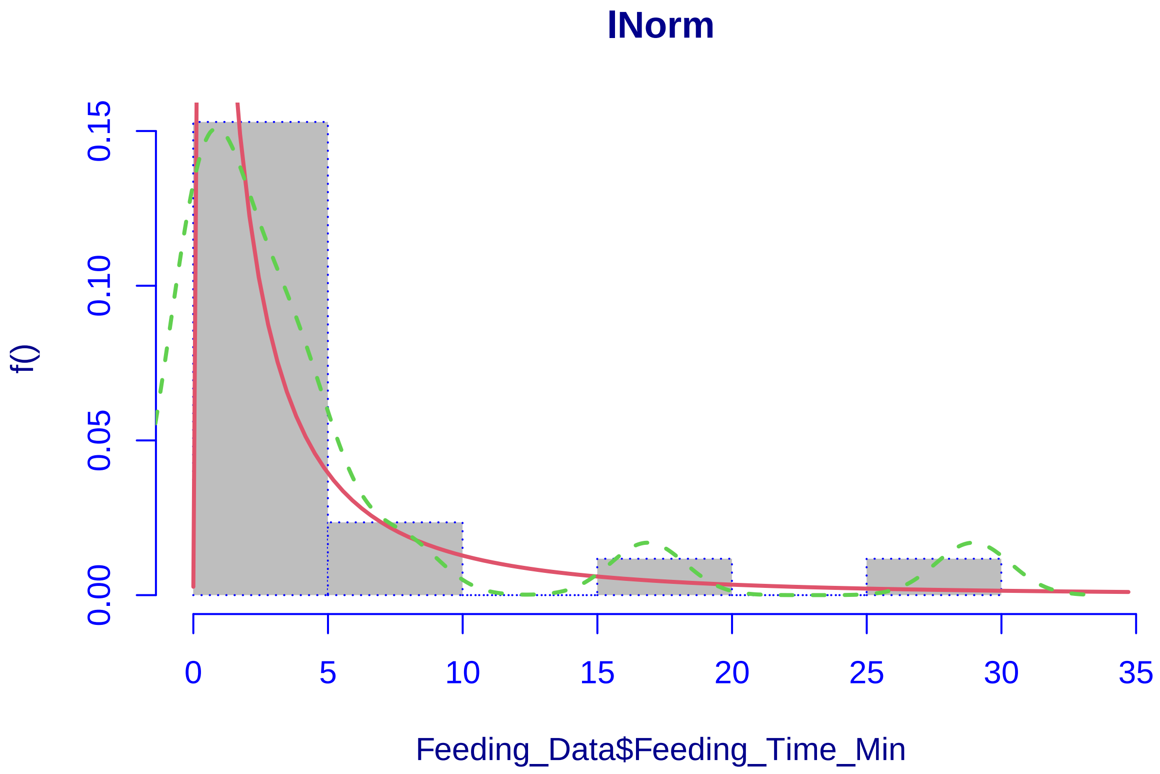
```
normal_test(Feeding_Data$Feeding_Time_Min)         #p-value < 0.05
```

```
##
##  Correlation test for normality
##
## data:  Feeding_Data$Feeding_Time_Min
## R = 0.88533, p-value = 0.0001004
## alternative hypothesis: Feeding_Data$Feeding_Time_Min does not follow a normal distribution.
```

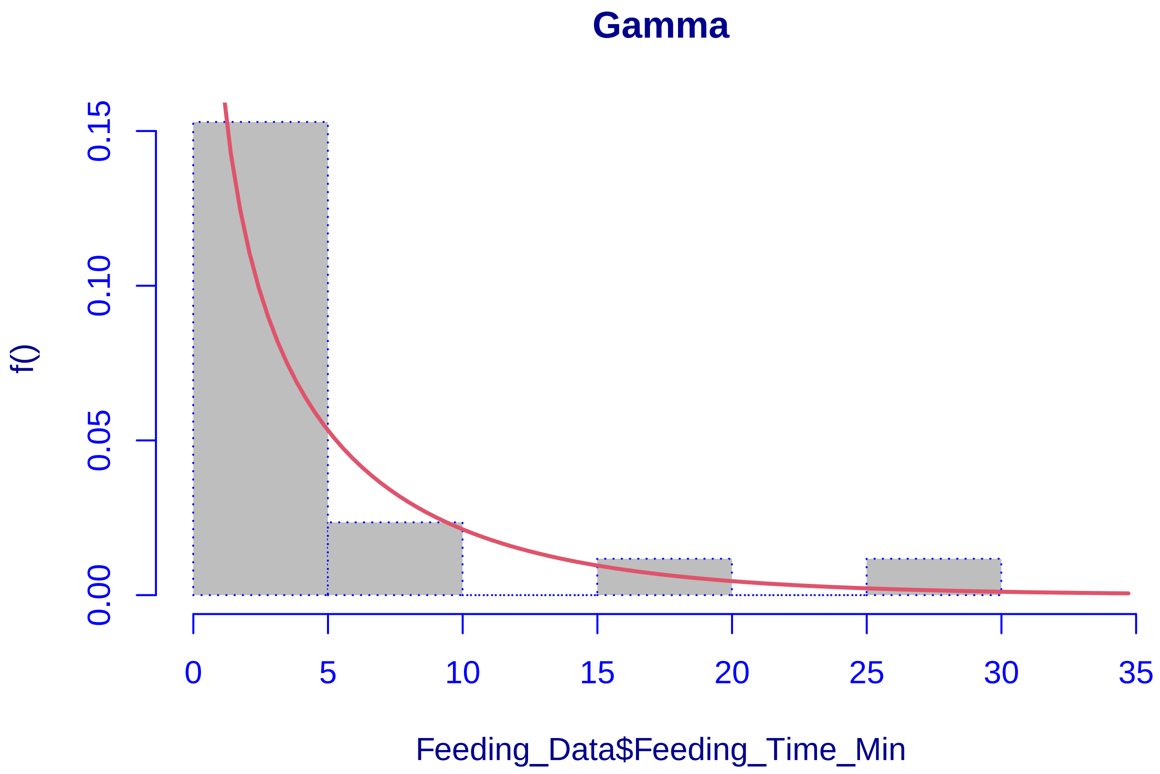
```
weibull_test(Feeding_Data$Feeding_Time_Min)        #p = 0.088
```

```
##
##  Test for the Weibull distribution
##
## data:  Feeding_Data$Feeding_Time_Min
## p-value = 0.138
```

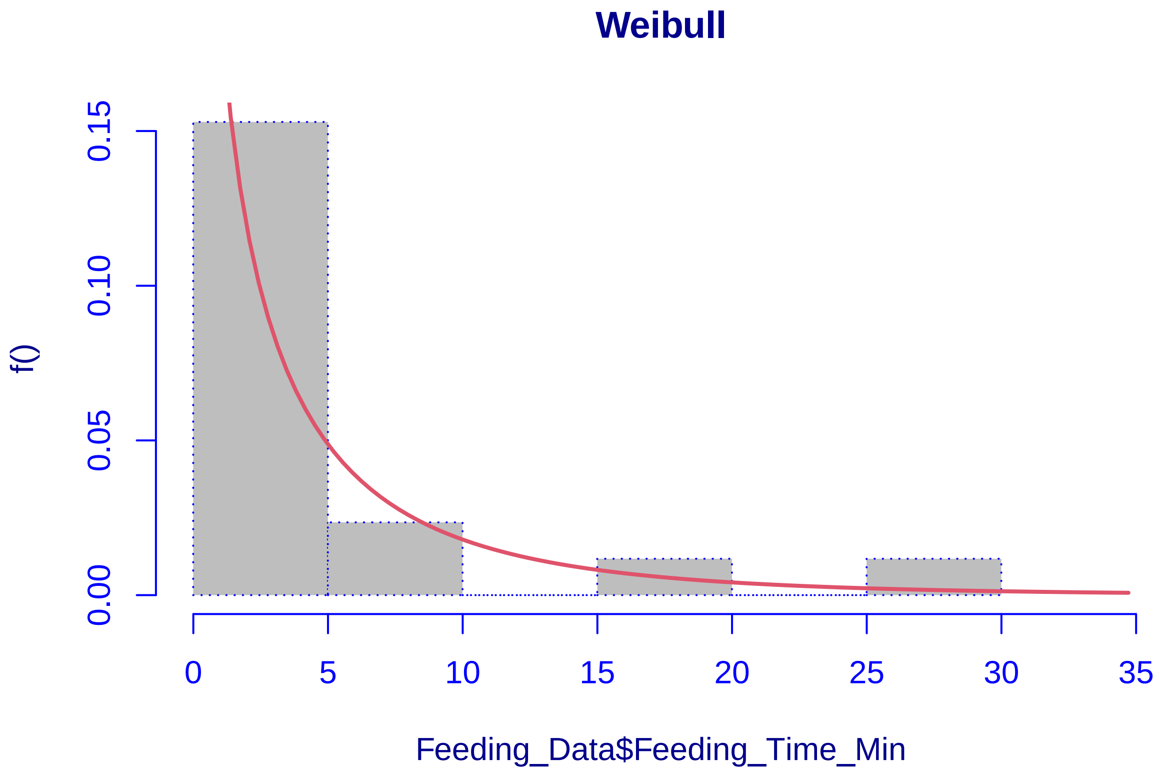
```
# Choosing the best distribution by comparing AIC value
Feeding_lnorm <- histDist(Feeding_Data$Feeding_Time_Min, "LOGNO",
                          density = T, main = "lNorm") #AIC = 112.738
```



```
Feeding_Gamma <- histDist(Feeding_Data$Feeding_Time_Min, "GA",
                          density = F, main = "Gamma") #AIC = 116.789
```



```
Green_Weibull <- histDist(Feeding_Data$Feeding_Time_Min, "WEI",
                          density = F, main = "Weibull") #AIC = 115.143
```



Running model: We are going to use the lnorm distribution because the p value is larger and AIC value is smaller.

```
Feeding_Data$orderTreatment = ordered(Feeding_Data$Treatment,
                                      levels = c("Control", "25C", "30C"))

Feeding_lNorm_model <- gamlss(formula = Feeding_Time_Min ~ Date*Treatment +
                                random(Species_ID) + random(Field_Site) +
                                random(Acclimation_Period),
                              family = LOGNO(), data = Feeding_Data,
                              control = gamlss.control(n.cyc = 3))
```

```
## GAMLSS-RS iteration 1: Global Deviance = 60.3695
## GAMLSS-RS iteration 2: Global Deviance = 60.3691
```

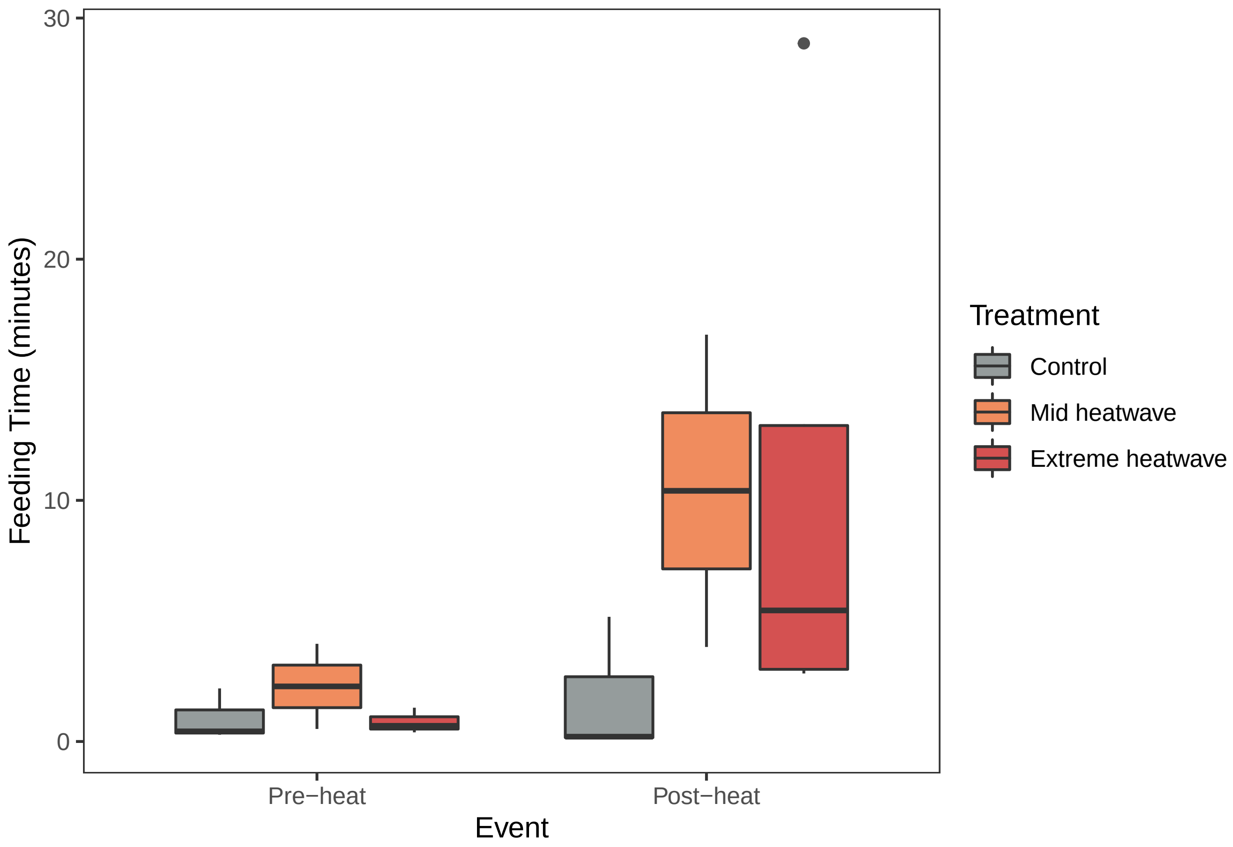
```
summary(Feeding_lNorm_model) # significance is only in post-heat 30C
```

```
## ******************************************************************
## Family:  c("LOGNO", "Log Normal")
##
## Call:  gamlss(formula = Feeding_Time_Min ~ Date * Treatment +
##     random(Species_ID) + random(Field_Site) + random(Acclimation_Period),
##     family = LOGNO(), data = Feeding_Data, control = gamlss.control(n.cyc = 3))
##
##
```

```
## Fitting method: RS()
##
## ------------------------------------------------------------------
## Mu link function:  identity
## Mu Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -0.5298     0.5149  -1.029   0.3330
## Date11/9/2021               -0.2449     0.7281  -0.336   0.7451
## Treatment25C                 0.8233     0.8141   1.011   0.3409
## Treatment30C                 0.3369     0.7281   0.463   0.6557
## Date11/9/2021:Treatment25C   1.9684     1.1513   1.710   0.1249
## Date11/9/2021:Treatment30C   2.3532     0.9970   2.360   0.0453 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ------------------------------------------------------------------
## Sigma link function:  log
## Sigma Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.1145     0.1715  -0.668    0.523
##
## ------------------------------------------------------------------
## NOTE: Additive smoothing terms exist in the formulas:
##  i) Std. Error for smoothers are for the linear effect only.
## ii) Std. Error for the linear terms maybe are not accurate.
## ------------------------------------------------------------------
## No. of observations in the fit:  17
## Degrees of Freedom for the fit:  8.831933
##       Residual Deg. of Freedom:  8.168067
##                       at cycle:  2
##
## Global Deviance:     60.36911
##             AIC:     78.03298
##             SBC:     85.39186
## ******************************************************************
```

Plot for Feeding:

```
feeding_plot <- ggplot(Feeding_Data, aes(x = Date, y = Feeding_Time_Min, fill = Treatment)) +
  geom_boxplot(alpha=0.85) +
  xlab("Event") +
  ylab("Feeding Time (minutes)") +
  scale_fill_manual(limits = c("Control", "25C", "30C"),
                    labels = c("Control", "Mid heatwave", "Extreme heatwave"),
                    values = cols_for_heat) +
  theme_test()
feeding_plot <- feeding_plot + scale_x_discrete(labels=c("11/5/2021" = "Pre-heat", "11/9/2021" = "Post-
feeding_plot
```



```r
ggsave(here("./images/feeding_plot.png"), feeding_plot)
```

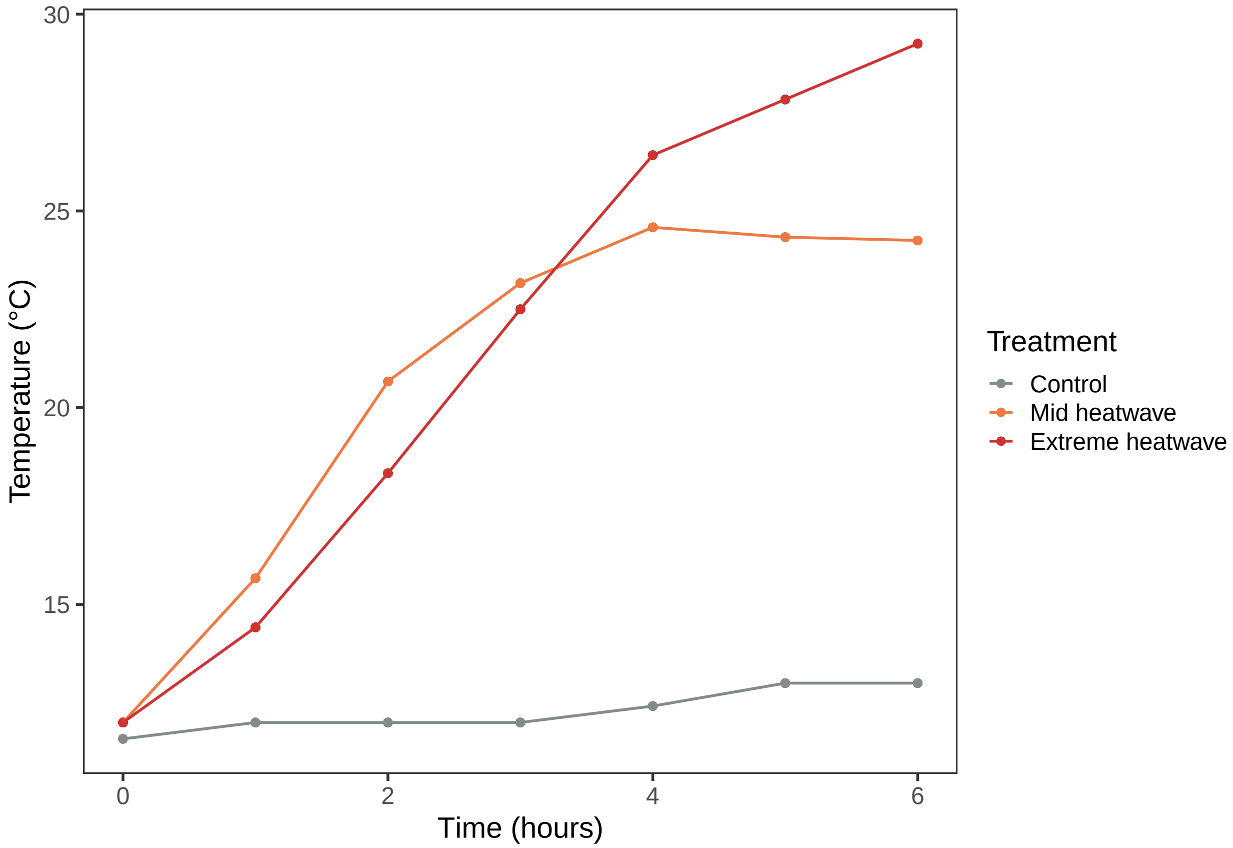
Heat Data

Includes temperature during heatwave and behavioral responses (open vs closed)

```r
heat_data <- heat_data %>%
  mutate(Open_Closed = as.factor(Open_Closed), Bucket = as.factor(Bucket),
         Treatment = as.factor(Treatment), Field_Site = as.factor(Field_Site)) %>%
  mutate(Open_Closed = fct_relevel(Open_Closed, "Open", "Partial", "Closed")) %>%
  mutate(Treatment = fct_relevel(Treatment, "Control", "25C", "30C")) %>%
  group_by(Treatment, Time_Block) %>%
  mutate(Temp_avg = mean(Bucket_Temp))
```

Temperature over Time Graph:

```r
heat_plot <- ggplot(data = heat_data, aes(x = Time_Block, y = Temp_avg, color = Treatment)) +
  geom_point(size = 1) +
  geom_line(alpha = 1.5) +
  xlab("Time (hours)") +
  ylab("Temperature (°C)") +
  scale_colour_manual(limits = c("Control", "25C", "30C"),
                      labels = c("Control", "Mid heatwave", "Extreme heatwave"),
                      values = cols_for_heat) +
  theme(axis.title.x = element_text(size=10),
        axis.title.y = element_text(size=10)) +
  theme_test()
heat_plot <- heat_plot + theme(legend.key.size = unit(0.75, 'lines'))
heat_plot
```



```
ggsave(here("./images/heat_plot.png"), heat_plot)
```

Ordinal Regression

Model to test significance on the open/partial/closed data.

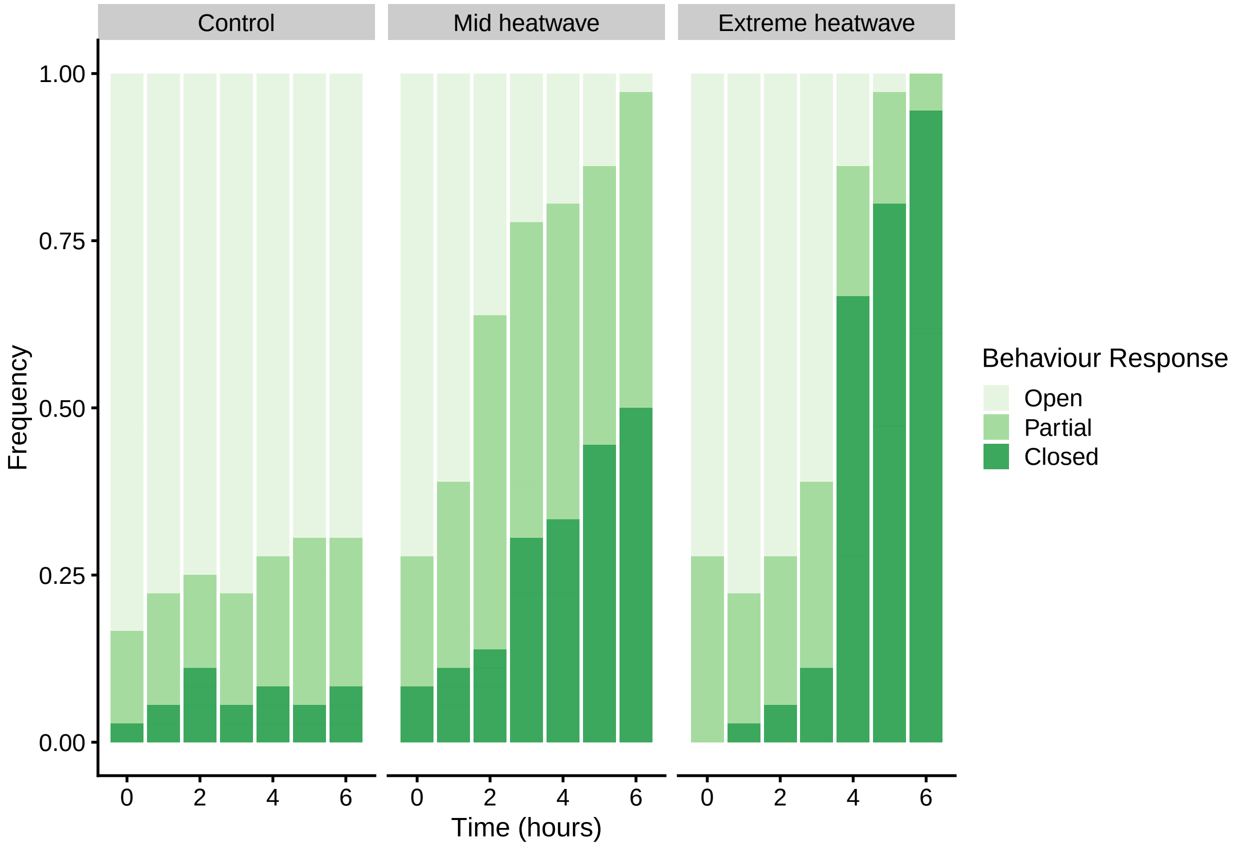
```
ord_model = clmm(Open_Closed ~ Treatment + (1 | Bucket) + (1 | Species_ID),
                 data = heat_data)
summary(ord_model) # significance in both treatments
```

```
## Cumulative Link Mixed Model fitted with the Laplace approximation
##
## formula: Open_Closed ~ Treatment + (1 | Bucket) + (1 | Species_ID)
## data:    heat_data
##
##  link  threshold nobs logLik  AIC     niter    max.grad cond.H
##  logit flexible  756  -682.12 1376.25 220(930) 2.31e-05 1.1e+02
```

```
##
## Random effects:
##  Groups     Name        Variance  Std.Dev.
##  Species_ID (Intercept) 1.069e+00 1.034e+00
##  Bucket     (Intercept) 1.190e-11 3.450e-06
## Number of groups:  Species_ID 36,  Bucket 12
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## Treatment25C   2.0438     0.4768   4.286 1.82e-05 ***
## Treatment30C   1.9778     0.4777   4.140 3.47e-05 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Threshold coefficients:
##                  Estimate Std. Error z value
## Open|Partial     1.3765     0.3492   3.941
## Partial|Closed   2.9049     0.3613   8.040
```

Open/Closed Graph

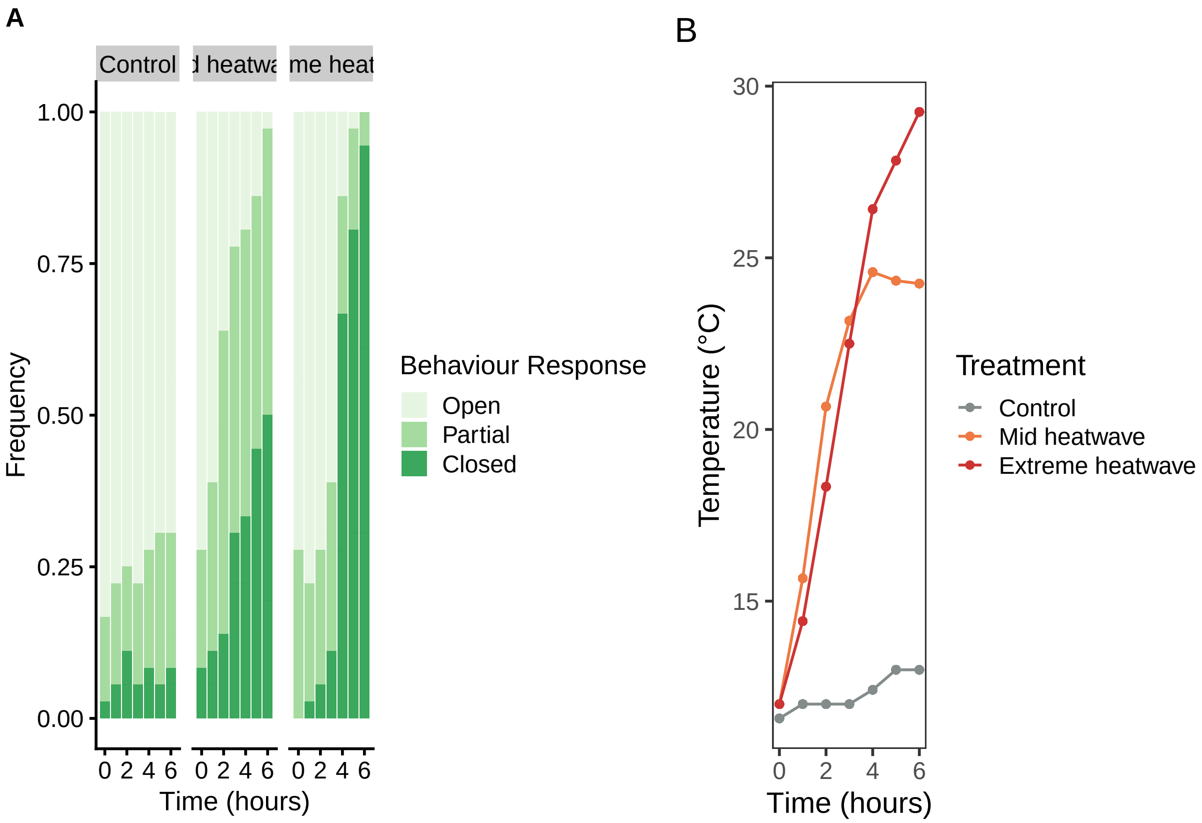
```
cols_for_behaviour = c("Open" = "olivedrab3", "Partial" = "forestgreen", "Closed" = "darkgreen")
behaviour_data <- heat_data %>%
  group_by(Day, Treatment, Time_Block) %>%
  count(Open_Closed)

behaviour_plot <- ggplot(data = behaviour_data, aes(x = Time_Block, y = n, fill = Open_Closed)) +
  geom_bar(alpha = 0.95, position="fill", stat="identity") +
  facet_grid(. ~ Treatment,
             labeller = labeller(Treatment = treatment_labels)) +
  xlab("Time (hours)") +
  ylab("Frequency") +
  scale_fill_brewer(palette = "Greens") +
  theme_cowplot(10)
behaviour_plot <- behaviour_plot + labs(fill = "Behaviour Response")
behaviour_plot
```



```
ggsave(here("./images/behaviour_plot.png"), behaviour_plot, width = 10, height = 6)
```

Combining plots together for comparision:

```
behaviour_heat <- behaviour_plot + heat_plot
behaviour_heat <- behaviour_heat + plot_layout(ncol=2, widths = c(1.8,1))
behaviour_heat <- behaviour_heat + plot_annotation(tag_levels = 'A')
behaviour_heat
```



```
ggsave(here("./images/behaviour_heat.png"), behaviour_heat, width = 10, height = 4)
```