Temperature conditions during the experiment

Ana Palacio

July 26, 2017

Contents

1. Libraries and settings

```
# Libraries
    library(plyr)
    library(tidyverse)
    library(reshape2)
    library(ggthemes)
    library(lubridate)

# Default ggplot settings

    ggthe_bw<-theme(plot.background=element_blank(),
          panel.grid.major.y = element_blank(),
          panel.grid.major.x = element_blank(),
          panel.grid.minor.x = element_blank(),
          panel.grid.minor.y = element_blank(),
          legend.box.background = element_rect(),
          panel.background =element_rect(fill = NA, color = "black")
          )+
    theme_bw()
```

2. Import the temperature data

```
Temperature<-read.csv("Day_Tem.csv", header = T)

Temperature$Period<-factor((Temperature$Period),levels=c
              ("Acer","Ofav-Ssid","Nutrients", "Ramping up",
                "Bleaching","Ramping down", "Recovery2"))

# Temperature data available from each treatment-replicate-day
Periods.days <- ddply (Temperature, .(Period, Treatment),summarise,
```

```
                Dmin = min (Day, na.rm = F),
                Dmax = max (Day, na.rm = F))
Periods.days
```

```
##          Period Treatment Dmin Dmax
## 1          Acer      Acer -127  -55
## 2     Ofav-Ssid Ofav-Ssid  -92   -9
## 3     Nutrients   Ambient   -5   78
## 4     Nutrients         N   -5   78
## 5     Nutrients       N+P   34   78
## 6    Ramping up   Ambient   79   90
## 7    Ramping up         N   79   90
## 8    Ramping up       N+P   79   90
## 9     Bleaching   Ambient   91  113
## 10    Bleaching         N   91   99
## 11    Bleaching       N+P   91  113
## 12 Ramping down   Ambient  114  118
## 13 Ramping down       N+P  114  118
## 14    Recovery2   Ambient  119  145
## 15    Recovery2       N+P  119  145
```

```
Temperature.82<-subset(Temperature, Day=="82")
Temperature.111<-subset(Temperature, Day=="111")

Periods.82 <- ddply (Temperature.82, .(Treatment),summarise,
                Tmean = mean (Temperature, na.rm = F),
                Tmin = min (Temperature, na.rm = F),
                Tmax = max (Temperature, na.rm = F))
Periods.82
```

```
##   Treatment    Tmean   Tmin   Tmax
## 1   Ambient 27.18789 25.319 27.961
## 2         N 27.44747 26.292 28.357
## 3       N+P 27.12858 25.222 27.862
```

```
Periods.82 <- ddply (Temperature.82, .(Day),summarise,
                Tmean = mean (Temperature, na.rm = F),
                Tmin = min (Temperature, na.rm = F),
                Tmax = max (Temperature, na.rm = F))
Periods.82
```

```
##   Day    Tmean   Tmin   Tmax
## 1  82 27.25465 25.222 28.357
```

```
Periods.82 <- ddply (Temperature.82, .(Day),summarise,
                Tmean = mean (Temperature, na.rm = F),
                Tsd = sd (Temperature, na.rm = F))
Periods.82
```

```
##   Day    Tmean       Tsd
## 1  82 27.25465 0.4853957
```

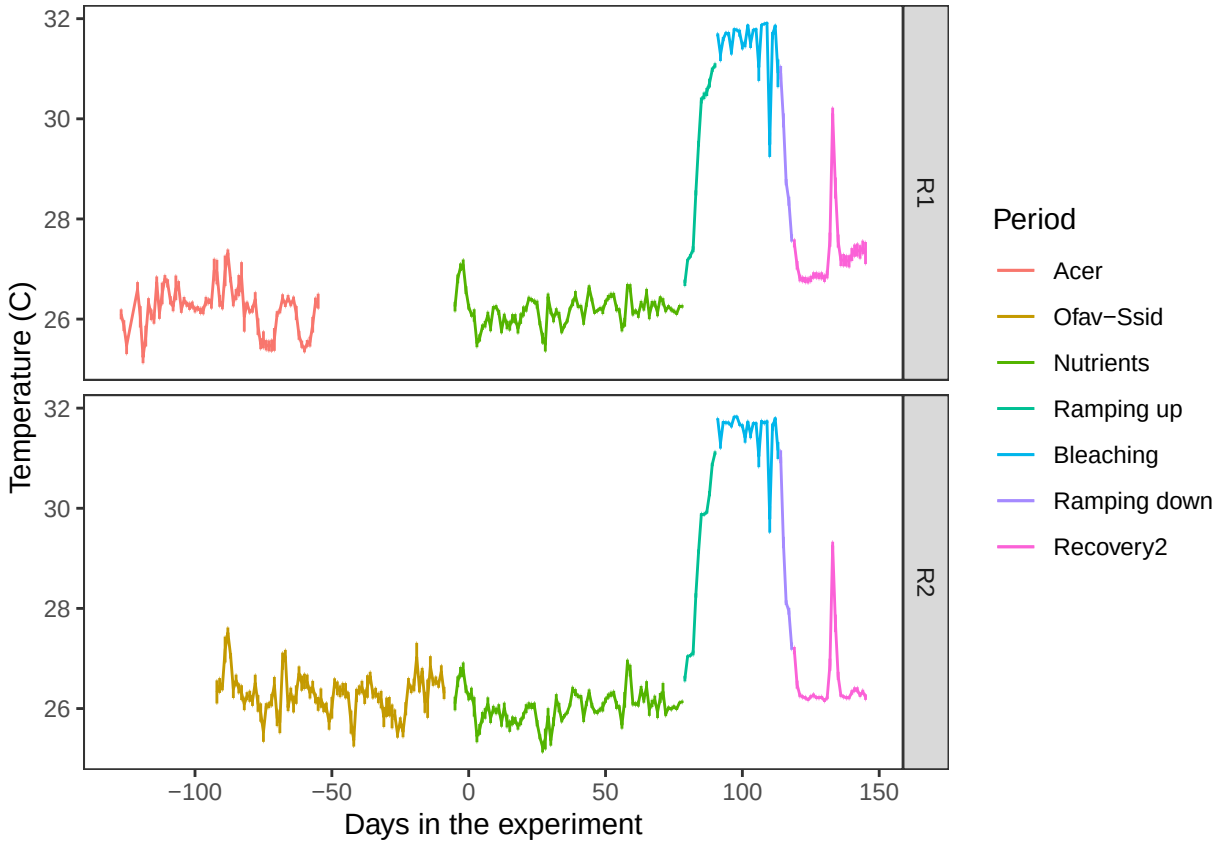
```
Periods.110 <- ddply (Temperature.111, .(Day),summarise,
                Tmean = mean (Temperature, na.rm = F),
                Tsd = sd (Temperature, na.rm = F))
Periods.110
```

```
##   Day     Tmean       Tsd
## 1 111 31.63097 0.4660265
```

3. Glance at temperature conditions during each period of the experiment:

- during recovery from collection and fragmentation and
- during the experiment (nutrient addition, ramping up, and bleaching)
- during recovery from bleaching

```
Temperature_Periods_R<- ggplot(Temperature, aes (Day, Temperature,
                                                 colour=factor(Period))) +
  stat_summary(fun.data = "mean_cl_boot",geom = "errorbar", width = 0.2 )+
  stat_summary(fun.y=mean, geom="line") +   ggthe_bw +
  xlab("Days in the experiment")  + guides(colour=guide_legend("Period")) +
  ylab("Temperature (C)") + facet_grid(Replicate~.)
Temperature_Periods_R
```




```
Summary_Period <- ddply (Temperature, .(Period),summarise,
                                Temp = mean (Temperature, na.rm = T),
                                Tsd  = sd (Temperature, na.rm = T),
                                Tmax = max (Temperature, na.rm = T),
                                Tmin = min (Temperature, na.rm = T))
Summary_Period
```

```
##         Period     Temp       Tsd   Tmax   Tmin
```

```
## 1         Acer 26.19668 0.6386856 29.252 24.545
## 2    Ofav-Ssid 26.29499 0.6429512 28.853 24.545
## 3    Nutrients 26.14369 0.4279447 28.655 24.545
## 4   Ramping up 29.01168 1.6423175 31.983 25.222
## 5    Bleaching 31.51851 0.7630519 31.983 26.879
## 6 Ramping down 28.81703 1.3859299 31.983 26.488
## 7    Recovery2 26.88718 0.9863522 31.370 25.416
```

```
Summary_Tanks <- ddply (Temperature, .(Period, Treatment, Replicate),summarise,
                            Temp = mean (Temperature, na.rm = T),
                            Tsd  = sd (Temperature, na.rm = T),
                            Tmax = max (Temperature, na.rm = T),
                            Tmin = min (Temperature, na.rm = T))
Summary_Tanks
```

```
##          Period Treatment Replicate     Temp       Tsd   Tmax   Tmin
## 1          Acer      Acer        R1 26.19668 0.6386856 29.252 24.545
## 2     Ofav-Ssid Ofav-Ssid        R2 26.29499 0.6429512 28.853 24.545
## 3     Nutrients   Ambient        R1 26.19199 0.4543236 28.159 24.545
## 4     Nutrients   Ambient        R2 26.01456 0.4414380 28.555 24.545
## 5     Nutrients         N        R1 26.25277 0.4449238 28.655 24.545
## 6     Nutrients         N        R2 26.13733 0.2567759 27.173 25.513
## 7     Nutrients       N+P        R1 26.11464 0.3395627 27.665 24.738
## 8     Nutrients       N+P        R2 26.12493 0.3780865 28.159 24.545
## 9    Ramping up   Ambient        R1 29.08071 1.6334266 31.676 25.902
## 10   Ramping up   Ambient        R2 28.92042 1.6011701 31.778 25.319
## 11   Ramping up         N        R1 29.28274 1.8410539 31.983 26.195
## 12   Ramping up         N        R2 29.01401 1.6025163 31.880 26.000
## 13   Ramping up       N+P        R1 28.97214 1.5821620 31.676 25.319
## 14   Ramping up       N+P        R2 28.85468 1.5949345 31.778 25.222
## 15    Bleaching   Ambient        R1 31.44172 0.7988322 31.983 27.173
## 16    Bleaching   Ambient        R2 31.49137 0.8092811 31.983 26.879
## 17    Bleaching         N        R1 31.53680 0.9977335 31.983 29.752
## 18    Bleaching         N        R2 31.72451 0.3850988 31.983 29.352
## 19    Bleaching       N+P        R1 31.50756 0.7800192 31.983 27.272
## 20    Bleaching       N+P        R2 31.52616 0.7684407 31.983 26.879
## 21 Ramping down   Ambient        R1 29.23688 1.1429751 31.983 27.370
## 22 Ramping down   Ambient        R2 28.70526 1.4715134 31.880 26.488
## 23 Ramping down       N+P        R1 28.60326 1.3065527 31.983 26.585
## 24 Ramping down       N+P        R2 28.75518 1.4866731 31.983 26.488
## 25    Recovery2   Ambient        R1 27.88521 0.8777922 31.370 26.585
## 26    Recovery2   Ambient        R2 26.51795 0.7668564 29.953 25.416
## 27    Recovery2       N+P        R1 26.66063 0.7658123 29.953 25.902
## 28    Recovery2       N+P        R2 26.48461 0.7753585 29.853 25.708
```

Figure 1: Experimental conditions

```
Figure1<- ggplot(Temperature, aes (Day, Temperature, colour=Replicate)) + ggthe_bw +

  scale_y_continuous(limits = c(24,33.4),
                       name=(expression("Temperature"~(degree*C))),
                       breaks = seq(24, 32, 1),
                       expand = c(0, 0)) +
```

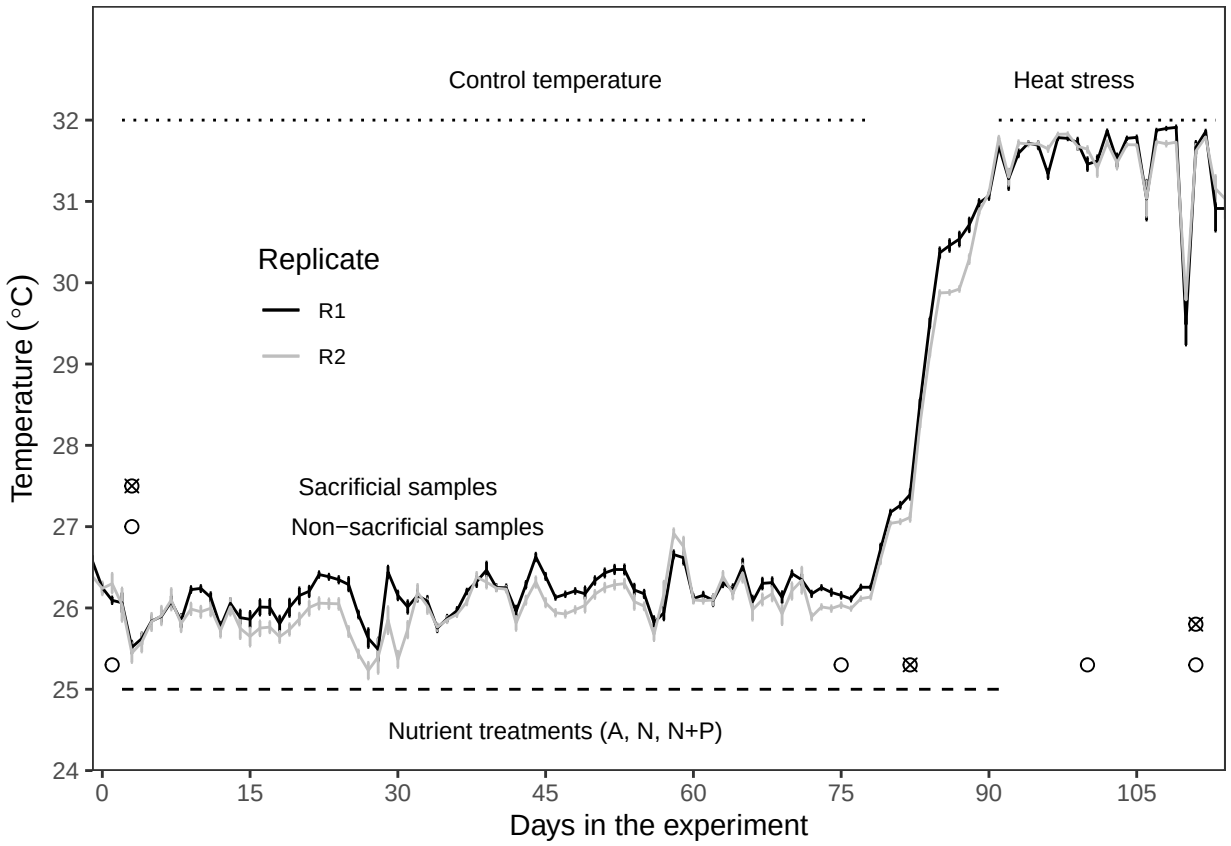
```
  scale_x_continuous(name="Days in the experiment",
                     limits = c(-1,114),
                     breaks = seq(0, 113, 15),
                     expand = c(0, 0))+

  annotate("segment", x = 2, xend = 91, y = 25, yend = 25,
             colour = "black", linetype=2)+
  annotate("text", x= c(46, 84, 101), y = c(24.5,25.5,24.5),
            label=c("Nutrient treatments (A, N, N+P)", " ", " "), size=3) +
  annotate("text", x= c(46, 99), y = c(32.5,32.5),
            label=c("Control temperature",
                    "Heat stress "), size=3) +
  annotate("segment", x = 2, xend = 78, y = 32, yend = 32,
             colour = "black", linetype=3)+
  annotate("segment", x = 91, xend = 113, y = 32, yend = 32,
             colour = "black", linetype=3)+

  theme(legend.position=c(0.2, 0.6),
        legend.text=element_text(size=8)) +
    scale_colour_manual(values = c("black","gray"))+
    stat_summary(fun.data = "mean_cl_boot",geom = "errorbar", width = 0.2 )+
    stat_summary(fun.y=mean, geom="line") +

  annotate("point",x=c(1, 75, 100, 111), y=c(25.3),
           shape=21, size=2, fill="white") +
  annotate("point",x=c(82, 111), y=c(25.3, 25.8),
           shape=21, size=2, fill="white")+
  annotate("point",x=c(82, 111), y=c(25.3, 25.8),
           shape=4, size=2)+
# Labels
  annotate("point",x=c(3, 3), y=c(27, 27.5),
           shape=21, size=2, fill="white")+
  annotate("point",x=c(3), y=c(27.5),
           shape=4, size=2)+
  annotate("text", x= c(32,30), y = c(27,27.5),
             label=c("Non-sacrificial samples",
                     "Sacrificial samples"), size=3)
```

Figure1




```
# ggsave(file="Outputs/Figure_1.svg", plot=Figure1, dpi = 300, width=4, height=2.5)
```

Packages used

```
# Creates bibliography
#knitr::write_bib(c(.packages()), "packages.bib")
```


Arnold, Jeffrey B. 2019. *Ggthemes: Extra Themes, Scales and Geoms for 'Ggplot2'*. https://CRAN.R-project.org/package=ggthemes.

Gohel, David, Hadley Wickham, Lionel Henry, and Jeroen Ooms. 2019. *Gdtools: Utilities for Graphical Rendering*. https://CRAN.R-project.org/package=gdtools.

Henry, Lionel, and Hadley Wickham. 2019. *Purrr: Functional Programming Tools*. https://CRAN.R-project.org/package=purrr.

Müller, Kirill, and Hadley Wickham. 2019. *Tibble: Simple Data Frames*. https://CRAN.R-project.org/package=tibble.

R Core Team. 2020. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Spinu, Vitalie, Garrett Grolemund, and Hadley Wickham. 2018. *Lubridate: Make Dealing with Dates a Little Easier*. https://CRAN.R-project.org/package=lubridate.

Wickham, Hadley. 2016. *Plyr: Tools for Splitting, Applying and Combining Data*. https://CRAN.R-project.

org/package=plyr.

———. 2017a. *Reshape2: Flexibly Reshape Data: A Reboot of the Reshape Package*. https://CRAN.R-project.org/package=reshape2.

———. 2017b. *Tidyverse: Easily Install and Load the 'Tidyverse'*. https://CRAN.R-project.org/package=tidyverse.

———. 2019a. *Forcats: Tools for Working with Categorical Variables (Factors)*. https://CRAN.R-project.org/package=forcats.

———. 2019b. *Stringr: Simple, Consistent Wrappers for Common String Operations*. https://CRAN.R-project.org/package=stringr.

Wickham, Hadley, and Lionel Henry. 2020. *Tidyr: Tidy Messy Data*. https://CRAN.R-project.org/package=tidyr.

Wickham, Hadley, Winston Chang, Lionel Henry, Thomas Lin Pedersen, Kohske Takahashi, Claus Wilke, Kara Woo, and Hiroaki Yutani. 2019. *Ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics*. https://CRAN.R-project.org/package=ggplot2.

Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2019. *Dplyr: A Grammar of Data Manipulation*. https://CRAN.R-project.org/package=dplyr.

Wickham, Hadley, Jim Hester, and Romain Francois. 2018. *Readr: Read Rectangular Text Data*. https://CRAN.R-project.org/package=readr.