Reproducing Coral Reef Community Structure Analyses

Installing Missing Packages

Taken from OA_OW_Physiology_manuscript.Rmd lines 70 - 85

```
## install packages from source code
#install.packages("devtools") # if you need to install this package
#library("devtools")
#install_github("vqv/ggbiplot")




### Used packages that need to be installed to run code (and can be sourced easier from CRAN):
needed_packages <- c("devtools", "knitr", "readr", "broom", "ggplot2", "dplyr", "ggbiplot", "ti

not_installed <- needed_packages[!(needed_packages %in% installed.packages()[ , "Package"])] #
if(length(not_installed)) install.packages(not_installed) # Install not installed packages
```

No issues to report, all packages installed as expected. When run a second time, not_installed list was empty and no packages attempted to install.

Loading Packages

Taken from OA_OW_Physiology_manuscript.Rmd lines 87 - 133

```
## Packages to load
library(knitr)
library(readr)
library(broom)
library(ggplot2)
library(dplyr)
library(ggbiplot)
library(tidyr)
library(corrgram)
library(openxlsx)
library(plotly)
library(tidyverse)
library(vegan)
library(shiny)
library(Rmisc)
library(cowplot)
library(ggfortify)
library(finalfit)
library(kableExtra)
library(readr)
library(lmerTest)
library(vroom)
library(ggpubr)
library(magick)
library(Hmisc)
library(corrplot)
library(gridGraphics)
library(grid)
library(RColorBrewer)
library(wesanderson)
library(performance)
library(MASS)
library(png)
library(car)
library(rcompanion)
library(janitor)
library(ncdf4)
library(raster)
library(xts)
library(ggrepel)

## Source the custom functions necessary for running this script
 source("Code/CustomFunctions.R")
```

All libraries loaded correctly. As this file is not connected to the GitHub repository at the moment, the source for the custom functions did not load. I have two options to contine: I can load the functions in directly, or I can directly source from the GitHub Repository. The latter is preferred, so I will proceed with that method.

First, I cloned the github repository to my local environment with the following line:

git clone https://github.com/seabove7/Bove_CoralPhysiology/

```
## Packages to load
library(knitr)
library(readr)
library(broom)
library(ggplot2)
library(dplyr)
library(ggbiplot)
library(tidyr)
library(corrgram)
library(openxlsx)
library(plotly)
library(tidyverse)
library(vegan)
library(shiny)
library(Rmisc)
library(cowplot)
library(ggfortify)
library(finalfit)
library(kableExtra)
library(readr)
library(lmerTest)
library(vroom)
library(ggpubr)
library(magick)
library(Hmisc)
library(corrplot)
library(gridGraphics)
library(grid)
library(RColorBrewer)
library(wesanderson)
library(performance)
library(MASS)
library(png)
library(car)
library(rcompanion)
library(janitor)
library(ncdf4)
library(raster)
library(xts)
library(ggrepel)

## Source the custom functions necessary for running this script
source("Bove_CoralPhysiology/Code/CustomFunctions.R")
```

After cloning and adding the downloaded respository Bove_CoralPhysiology before the line, I have successfully sourced the code.

Formatting

Taken from OA_OW_Physiology_manuscript.Rmd lines 163 - 229

```
## Set some standards and units

# dodge width
dodge=position_dodge(width=0.3)
dodge2=position_dodge(width = 0.6)
jitter=position_jitter(width=0.1)

# set figure theme
theme_set(theme_pubr())

# set parameter labels
Tlab<-"Temperature (°C)"
alab<-expression(paste(italic("p"),"CO" [2]~ "("*mu,'atm)'))
```

```
dlab<-expression(paste("Cell density (10"^6,"cells cm"^-2,")"))
plab<-expression(paste("Total protein (mg cm"^-2,")"))
calab<-expression(paste("Carbohydrate (mg cm"^-2,")"))
clab<-expression(paste("Chlorophyll a ("*mu,"g cm"^-2,")"))
rlab<-expression(paste("Calcification rate (mg cm"^-2~"day"^-1,")"))
llab<-expression(paste("Total Lipid (mg cm"^-2,")"))
hlab<-expression(paste("Total Host (mg cm"^-2,")"))

# set the current date
date <- Sys.Date()
```

Everything performs as expected

Reading in data

Taken from OA_OW_Physiology_manuscript.Rmd lines 135 - 161

```
 df <- read_csv("Data/Raw_data/phys_all_23March2021.csv")[-1] # read in full dataframe

# rename a couple columns
names(df)[38]<-"carb"
names(df)[39]<-"lipid"
names(df)[17]<-"sum"
names(df)[34]<-"red"

# set columns as factors
df$ftemp <- as.factor(df$ftemp)
df$fpco2 <- as.factor(df$fpco2)
df$colony <- as.factor(df$colony)
df$species <- factor(df$species, levels = c("S", "P", "A", "T")) # and reorder these

# replace some pCO2 values with ones we will use moving forward
df$fpco2 <- gsub("2800", "3290", df$fpco2)
df$fpco2 <- gsub("280", "300", df$fpco2)
df$fpco2 <- gsub("400", "420", df$fpco2)
df$fpco2 <- gsub("700", "680", df$fpco2)

# reorder factors for pCO2 and temp
df$fpco2 <- factor(df$fpco2, c("420", "T0", "300", "680", "3290"))
df$ftemp <- factor(df$ftemp, c("28", "T0", "31"))

# calculate phys parameters
df$den <- (df$den / 1000000) # adjust symbiont density to display 10^6 cells
df$chla <- (df$chla / df$SA) # standardize chla to surface area
df$host <- df$pro + df$carb + df$lipid # calculate total host energy reserves (sum of carb, pro
df$treat[df$T0_T90 == "T0"] <- "T0" # replace T0 'treat' with T0 text
df$treat <- factor(df$treat, levels = c("T0", "288_28", "311_31", "447_28", "405_31", "673_28",
                   labels = c("T0", "300_28", "300_31", "420_28", "420_31", "680_28", "680_31",
# add a new treat column for plotting (replacing actual treatment with number for better plotti
df$treat2 <- df$treat
df$treat <- gsub("T0", 2, df$treat)
df$treat <- gsub("300_28", 4, df$treat)
df$treat <- gsub("300_31", 4, df$treat)
df$treat <- gsub("420_28", 6, df$treat)
df$treat <- gsub("420_31", 6, df$treat)
df$treat <- gsub("680_28", 8, df$treat)
df$treat <- gsub("680_31", 8, df$treat)
df$treat <- gsub("3290_28", 10, df$treat)
df$treat <- gsub("3290_31", 10, df$treat)
df$treat <- as.numeric(df$treat) # convert 'treat' column to numerics (again, this is for plott

# inverse of colors (so lower color depicts more bleached coral)
df$sum <- (df$sum * -1)
df$red <- (df$red * -1)
df$blue_bw5 <- (df$blue_bw5 * -1)
df$green_bw5 <- (df$green_bw5 * -1)

## modify dataframe for simplified version
df2 <- df[,c(1:13, 41, 14:16, 37:40, 42:43, 17)] # select columns of interest only
df2 <- gather(df2, param, value, den:sum) # make column of parameter and value
df2$param <- factor(df2$param, levels = c("pro", "carb", "lipid", "den", "chla", "host", "count
df2$species <- revalue(x = df2$species, c("S" = "SSID", "P" = "PSTR", "A" = "PAST", "T" = "UTEN
df2 <- subset(df2, species != "UTEN") # remove UTEN (omitted due to high mortality in some trea

## remote T0 samples from dataframe for models
df2_T90 <- subset(df2, T0_T90 == "T90")

## add treatment column
df2_T90$treat2 <- paste0(df2_T90$fpco2, df2_T90$temp)
```

Dataframe was not read in. The initial directory must be added because of the cloning process.

```
df <- read_csv("Bove_CoralPhysiology/Data/Raw_data/phys_all_23March2021.csv")[-1] # read in ful
```

```
New names:
Rows: 310 Columns: 42
── Column specification
──────────────────────────────────────────────────────── Delimiter: "," chr
(10): coral, tank, treat, T0_T90, reef, species, col, fpco2, ftemp, colony dbl
(32): ...1, pco2, temp, SA, den, pro, chla, sum_bw5, red_bw1, green_bw1,...
ℹ Use `spec()` to retrieve the full column specification for this data. ℹ
Specify the column types or set `show_col_types = FALSE` to quiet this message.
• `` -> `...1`
```

```
# rename a couple columns
names(df)[38]<-"carb"
names(df)[39]<-"lipid"
names(df)[39]<-"lipid"
names(df)[17]<-"sum"
names(df)[34]<-"red"

# set columns as factors
df$ftemp <- as.factor(df$ftemp)
df$fpco2 <- as.factor(df$fpco2)
df$colony <- as.factor(df$colony)
df$species <- factor(df$species, levels = c("S", "P", "A", "T")) # and reorder these

# replace some pCO2 values with ones we will use moving forward
df$fpco2 <- gsub("2800", "3290", df$fpco2)
df$fpco2 <- gsub("280", "300", df$fpco2)
df$fpco2 <- gsub("400", "420", df$fpco2)
df$fpco2 <- gsub("700", "680", df$fpco2)

# reorder factors for pCO2 and temp
df$fpco2 <- factor(df$fpco2, c("420", "T0", "300", "680", "3290"))
df$ftemp <- factor(df$ftemp, c("28", "T0", "31"))

# calculate phys parameters
df$den <- (df$den / 1000000) # adjust symbiont density to display 10^6 cells
df$chla <- (df$chla / df$SA) # standardize chla to surface area
df$host <- df$pro + df$carb + df$lipid # calculate total host energy reserves (sum of carb, pro
df$treat[df$T0_T90 == "T0"] <- "T0" # replace T0 'treat' with T0 text
df$treat <- factor(df$treat, levels = c("T0", "288_28", "311_31", "447_28", "405_31", "673_28",
                   labels = c("T0", "300_28", "300_31", "420_28", "420_31", "680_28", "680_31",

# add a new treat column for plotting (replacing actual treatment with number for better plotti
df$treat2 <- df$treat
df$treat <- gsub("T0", 2, df$treat)
df$treat <- gsub("300_28", 4, df$treat)
df$treat <- gsub("300_31", 4, df$treat)
df$treat <- gsub("420_28", 6, df$treat)
df$treat <- gsub("420_31", 6, df$treat)
df$treat <- gsub("680_28", 8, df$treat)
df$treat <- gsub("680_31", 8, df$treat)
df$treat <- gsub("3290_28", 10, df$treat)
df$treat <- gsub("3290_31", 10, df$treat)
df$treat <- as.numeric(df$treat) # convert 'treat' column to numerics (again, this is for plott

# inverse of colors (so lower color depicts more bleached coral)
df$sum <- (df$sum * -1)
df$red <- (df$red * -1)
df$blue_bw5 <- (df$blue_bw5 * -1)
df$green_bw5 <- (df$green_bw5 * -1)

## modify dataframe for simplified version
df2 <- df[,c(1:13, 41, 14:16, 37:40, 42:43, 17)] # select columns of interest only
df2 <- gather(df2, param, value, den:sum) # make column of parameter and value
```

```
df2$param <- factor(df2$param, levels = c("pro", "carb", "lipid", "den", "chla", "host", "count
df2$species <- revalue(x = df2$species, c("S" = "SSID", "P" = "PSTR", "A" = "PAST", "T" = "UTEN
df2 <- subset(df2, species != "UTEN") # remove UTEN (omitted due to high mortality in some trea

## remote T0 samples from dataframe for models
df2_T90 <- subset(df2, T0_T90 == "T90")

## add treatment column
df2_T90$treat2 <- paste0(df2_T90$fpco2, df2_T90$temp)
```

With that adjustment, all errors are solved.

Bootstrap model setup

Taken from OA_OW_Physiology_manuscript.Rmd lines 231 - 238

```
## Performing the parametric bootstrapping of the model:
bootnum = 1500 # set number of iterations (we used 2000) between 999 and 9999
seed = 30 # seed to make results replicatable (our seed was 3)
set.seed(30)
```

Unclear on whether seeds and bootnumbers need to be changed. Based on the information given, I will redefine them to what the notes suggest and edit again if results are not identical.

```
## Performing the parametric bootstrapping of the model:
bootnum = 2000 # set number of iterations (we used 2000) between 999 and 9999
seed = 3 # seed to make results replicatable (our seed was 3)
set.seed(3)
```

Separating data by parameter

Taken from OA_OW_Physiology_manuscript.Rmd lines 231 - 238

```
## Create a forloop to subset the data by each parameter and save as individual dataframes

# make a list of parameter names
param_list <- levels(df2_T90$param)

# forloop for dataframes
for (p in 1:length(param_list)) {
  param_select <- param_list[p]
  df_subset <- subset(df2_T90, param == param_select) # subset the dataframe for only one param
  df_subset <- completeFun(df_subset, "value") # run function to remove any missing values
  df_subset$value <- as.numeric(df_subset$value)
  df_subset$ftemp <- droplevels(df_subset$ftemp)
  df_subset$fpco2 <- droplevels(df_subset$fpco2)
  df_subset$colony <- droplevels(df_subset$colony)
  df_subset$species <- droplevels(df_subset$species)
  assign(paste(param_select, "mod_df", sep = "_"), df_subset) # assign the data to dataframe na
}
```

Everything runs as expected.

Creating Dataframes (for Sample Size Calculations)

Taken from OA_OW_Physiology_manuscript.Rmd lines 262 - 278

```
df_withT <- df %>% filter(T0_T90 == "T90") %>% droplevels()
df_90 <- subset(df_withT, species != "T")
df_90 <- df_90[,-c(18:33)]
df_90 <- completeFun(df_90, "den")
df_90 <- completeFun(df_90, "host")
df_90 <- completeFun(df_90, "lipid")
df_90 <- completeFun(df_90, "carb")
df_90_l <- gather(df_90, param, value, 14:23)

## specific species dataframes
s_df <- subset(df_90, species == "S")
p_df <- subset(df_90, species == "P")
a_df <- subset(df_90, species == "A")
```

Everything runs as expectede

PCA

PCA for species Siderastrea Siderea

Taken from OA_OW_Physiology_manuscript.Rmd lines 290 - 315

```
# set up the dataframe
s_df <- unique(s_df) # remove any duplicate rows
s_df_l <- gather(s_df, param, value, c(14:17,21:23))
s_df$fpco2 <- factor(s_df$fpco2, levels = c("300", "420", "680", "3290"))
sid_pca_df <- s_df[,c(14:17,21:23)]
sid_pca_df <- rename(sid_pca_df, colour = sum) # renaming the 'sum' column to 'colour'

# run the adonis
s_pca_mod_full <- adonis2(sid_pca_df ~ reef * ftemp * fpco2, data = s_df, method = 'eu', permut
s_pca_mod <- adonis2(sid_pca_df ~ fpco2 + ftemp + reef, data = s_df, method = 'eu', permutation
s_pca_mod # view SSID adonis output

# pull AIC from the full and reduced PERMANOVA models
s_pca_aic_full <- round(AICc.PERMANOVA2(s_pca_mod_full)[[1]], 1)
s_pca_aic_full <- round(AICc.PERMANOVA2(s_pca_mod_full)[[1]], 1)
s_pca_aic_final <- round(AICc.PERMANOVA2(s_pca_mod)[[1]], 1)



# extract pvalues
s_pval <- s_pca_mod["Pr(>F)"]

# perform principal component analysis (PCA)
s_pca <- prcomp(sid_pca_df, center = TRUE, scale= TRUE)
```

Error caught in line 297 of renaming column.

```
# set up the dataframe
s_df <- unique(s_df) # remove any duplicate rows
s_df_l <- gather(s_df, param, value, c(14:17,21:23))
s_df$fpco2 <- factor(s_df$fpco2, levels = c("300", "420", "680", "3290"))
sid_pca_df <- s_df[,c(14:17,21:23)]
sid_pca_df <- dplyr::rename(sid_pca_df, colour = sum) # renaming the 'sum' column to 'colour'

# run the adonis
s_pca_mod_full <- adonis2(sid_pca_df ~ reef * ftemp * fpco2, data = s_df, method = 'eu', permut
s_pca_mod <- adonis2(sid_pca_df ~ fpco2 + ftemp + reef, data = s_df, method = 'eu', permutation
s_pca_mod # view SSID adonis output
```

```
Permutation test for adonis under reduced model
Marginal effects of terms
Permutation: free
Number of permutations: 2000

adonis2(formula = sid_pca_df ~ fpco2 + ftemp + reef, data = s_df, permutations = bootnum,
method = "eu", by = "margin")
         Df SumOfSqs      R2      F    Pr(>F)    
fpco2     3    59423 0.20282 7.9334 0.0004998 ***
ftemp     1     9320 0.03181 3.7329 0.0459770 *  
reef      1    24705 0.08432 9.8948 0.0019990 ** 
Residual 80   199740 0.68174                     
Total    85   292988 1.00000                     
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# pull AIC from the full and reduced PERMANOVA models
s_pca_aic_full <- round(AICc.PERMANOVA2(s_pca_mod_full)[[1]], 1)
s_pca_aic_final <- round(AICc.PERMANOVA2(s_pca_mod)[[1]], 1)



# extract pvalues
s_pval <- s_pca_mod["Pr(>F)"]

# perform principal component analysis (PCA)
s_pca <- prcomp(sid_pca_df, center = TRUE, scale= TRUE)
s_pca <- prcomp(sid_pca_df, center = TRUE, scale= TRUE)
```

Specified dplyr package in case it used a different rename function. This renamed the variable in the dataframe so the change was successful. Everything runs as expected.

PCA Plot

Taken from OA_OW_Physiology_manuscript.Rmd lines 317 - 426

```
# create labels for p values calculated above
s_pco2_pval <-substitute(italic(P[pCO[2]])==p, list(p = format(s_pval[1,1], digits = 1)))
s_temp_pval <-substitute(italic(P[temp])==p, list(p = format(s_pval[2,1], digits = 1)))
s_reef_pval <-substitute(italic(P[reef])==p, list(p = format(s_pval[3,1], digits = 1)))


# temperature = shape; pco2 = colours
s_pca_plot <- autoplot(s_pca, data = s_df,
         colour = "fpco2",
         shape = "ftemp",
         frame = FALSE,
         loadings = TRUE,
         loadings.colour = "grey29",
         loadings.label = TRUE,
         loadings.label.colour = "black",
         loadings.label.size = 4,
         loadings.label.hjust = 1.5,
         loadings.label.vjust = 0.5,
         loadings.label.repel = TRUE) +
    stat_ellipse(type = "t", aes(colour = fpco2)) +
    scale_shape_manual("", labels = c("28 C", "31 C"), values = c(19, 1)) +
    scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme")
    guides(linetype = "none", shape = guide_legend(keyheight = 0.7, keywidth = 0.1, order = 1,
    guides(color = guide_legend(keyheight = 0.7, keywidth = 0.1, order = 2, nrow = 2, override.
    annotate("text", x = -0.338, y = -0.39, label = deparse(s_temp_pval), parse = TRUE, size =
    annotate("text", x = -0.33, y = -0.345, label = deparse(s_pco2_pval), parse = TRUE, size =
    xlim(-0.4, 0.4) +
    theme(legend.background = element_rect(fill = "transparent", color = NA), legend.position =

guide_pco2_color <- get_legend(s_pca_plot + guides(linetype = "none", shape = "none")) # extrac

s_pca_plot <- s_pca_plot + # add colour guide to new location and save
  guides(color = "none") +
  annotation_custom(guide_pco2_color, xmax = 0.78, ymax = -0.28)


# reef
s_reef_pca <- autoplot(s_pca, data = s_df,
         colour = "reef",
         fill = "reef",
         frame = TRUE,
         frame.type = "t", # displaying ellipses with multivariate t-distributions for small n
         frame.level = 0.95, # using 95% CI for all ellipses
         frame.alpha = 0.01,
         loadings = TRUE,
         loadings.colour = "grey29",
         loadings.label = TRUE,
         loadings.label.colour = "black",
         loadings.label.size = 4,
         loadings.label.hjust = 1.5,
         loadings.label.vjust = 0.5,
         loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b'
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  annotate("text", x = -0.3, y = -0.38, label = deparse(s_reef_pval), parse = TRUE, size = 3) +
  ggtitle(expression(paste(bold("A")   "), italic("S. siderea"))))

# pco2 * temp
s_df$pco2_f <- factor(s_df$pco2, labels = c("pre industrial; 28C", "pre industrial; 31C", "curr
s_pt_pca <- autoplot(s_pca, data = s_df,
         colour = "pco2_f",
         fill = "pco2_f",
         shape = "pco2_f",
         frame = TRUE,
         frame.type = "t", # displaying ellipses with multivariate t-distributions for small n
         frame.level = 0.95, # using 95% CI for all ellipses
         frame.alpha = 0.01,
         frame.size = 5,
         loadings = TRUE,
         loadings.colour = "grey29",
         loadings.label = TRUE,
         loadings.label.colour = "black",
         loadings.label.size = 4,
         loadings.label.hjust = 1.5,
         loadings.label.vjust = 0.5,
         loadings.label.repel = TRUE) +
  scale_fill_manual("", values = c("#d1e5f0", "#fddbc7", "#f4a582", "#92c5de", "#4393c3", "#d6e
  scale_colour_manual("", values = c("#d1e5f0", "#fddbc7", "#f4a582", "#92c5de", "#4393c3", "#c
  scale_shape_manual("", values = c(19, 19, 17, 17, 15, 15, 8, 8)) +
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 4, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",


### Trying a single PCA with colour for pCO2 and shape for temp:
# ellipse by temp, points by pCO2
# s_temp_pca2 <- autoplot(s_pca, data = s_df,
#     colour = "fpco2",
#     shape = "ftemp",
#     frame = FALSE,
#           frame = FALSE,
#           loadings = TRUE,
#           loadings.colour = "grey29",
#           loadings.label = TRUE,
#           loadings.label.colour = "black",
#           loadings.label.size = 4,
#           loadings.label.hjust = 1.5,
#           loadings.label.vjust = 0.5,
#           loadings.label.repel = TRUE) +
#      stat_ellipse(type = "t", aes(lty = ftemp, colour = ftemp)) +
#      scale_linetype_manual(values = c("solid", "dashed")) +
#      scale_shape_manual("temperature", labels = c("28 C", "31 C"), values = c(19, 1)) +
#      scale_color_manual("", labels = c("28 C", "31 C", "pre industrial", "current", "end-of-ce
#      guides(linetype = "none", shape = "none") +
#      guides(color = guide_legend(nrow = 2, override.aes = list(linetype = c(1, 2, 0, 0, 0, 0),
#      theme(legend.background = element_rect(fill = "transparent", color = NA), legend.positior
```

PCA for species Pseudodiploria strigosa

Taken from OA_OW_Physiology_manuscript.Rmd lines 436 - 459

```
# set up the dataframe
p_df <- unique(p_df) # remove duplicate rows
p_df_l <- gather(p_df, param, value, c(14:16,21:23))
p_df$fpco2 <- factor(p_df$fpco2, levels = c("300", "420", "680", "3290"))
dip_pca_df <- p_df[,c(14:17,21:23)]
dip_pca_df <- rename(dip_pca_df, colour = sum) # renaming the 'sum' column to 'colour'

# run the adonis
p_pca_mod_full <- adonis2(dip_pca_df ~ reef * ftemp * fpco2, data = p_df, method = 'eu', permut
p_pca_mod <- adonis2(dip_pca_df ~ reef + ftemp + fpco2, data = p_df, method = 'eu', permutatior
p_pca_mod # view PSTR adonis output

# pull AIC from the full and reduced PERMANOVA models
p_pca_aic_full <- round(AICc.PERMANOVA2(p_pca_mod_full)[[1]], 1)
p_pca_aic_final <- round(AICc.PERMANOVA2(p_pca_mod)[[1]], 1)



# extract pvalues
p_pval <- p_pca_mod["Pr(>F)"]

# perform principal component analysis (PCA)
p_pca <- prcomp(dip_pca_df, center = TRUE, scale= TRUE)
```

Error in rename function as before. Will specify dplyr package and that should solve the error.

```
# set up the dataframe
p_df <- unique(p_df) # remove duplicate rows
p_df_l <- gather(p_df, param, value, c(14:16,21:23))
p_df$fpco2 <- factor(p_df$fpco2, levels = c("300", "420", "680", "3290"))
```

```
p_df$fpco2 <- factor(p_df$fpco2, levels = c("300", "420", "680", "3290"))
dip_pca_df <- p_df[,c(14:17,21:23)]
dip_pca_df <- dplyr::rename(dip_pca_df, colour = sum) # renaming the 'sum' column to 'colour'

# run the adonis
p_pca_mod_full <- adonis2(dip_pca_df ~ reef * ftemp * fpco2, data = p_df, method = 'eu', permut
p_pca_mod <- adonis2(dip_pca_df ~ reef + ftemp + fpco2, data = p_df, method = 'eu', permutatior
p_pca_mod # view PSTR adonis output
```

```
Permutation test for adonis under reduced model
Marginal effects of terms
Permutation: free
Number of permutations: 2000

adonis2(formula = dip_pca_df ~ reef + ftemp + fpco2, data = p_df, permutations = bootnum,
method = "eu", by = "margin")
         Df SumOfSqs      R2       F    Pr(>F)    
reef      1   101796 0.09008 14.8653 0.0009995 ***
ftemp     1   519372 0.45958 75.8438 0.0004998 ***
fpco2     3    30444 0.02694  1.4819 0.2228886    
Residual 71   486202 0.43023                      
Total    76  1130099 1.00000                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# pull AIC from the full and reduced PERMANOVA models
p_pca_aic_full <- round(AICc.PERMANOVA2(p_pca_mod_full)[[1]], 1)
p_pca_aic_final <- round(AICc.PERMANOVA2(p_pca_mod)[[1]], 1)



# extract pvalues
p_pval <- p_pca_mod["Pr(>F)"]

# perform principal component analysis (PCA)
p_pca <- prcomp(dip_pca_df, center = TRUE, scale= TRUE)
```

PCA Plot

Taken from OA_OW_Physiology_manuscript.Rmd lines 461 - 571

```
# create labels for p values calculated above
p_reef_pval <-substitute(italic(P[reef])==p, list(p = format(p_pval[1,1], digits = 1)))
p_temp_pval <-substitute(italic(P[temp])==p, list(p = format(p_pval[2,1], digits = 1)))
p_pco2_pval <-substitute(italic(P[pCO[2]])==p, list(p = format(p_pval[3,1], digits = 2)))



# temperature = colour; pco2 = shape
p_pca_plot <- autoplot(p_pca, data = p_df,
         shape = "fpco2",
         colour = "ftemp",
         colour = "ftemp",
         frame = FALSE,
         loadings = TRUE,
         loadings.colour = "grey29",
         loadings.label = TRUE,
         loadings.label.colour = "black",
         loadings.label.size = 4,
         loadings.label.hjust = 1.5,
         loadings.label.vjust = 0.5,
         loadings.label.repel = TRUE) +
  stat_ellipse(type = "t", aes(colour = ftemp)) +
  scale_shape_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  scale_color_manual("", labels = c("28 C", "31 C"), values = c("#4393c3", "#b2182b")) +
  guides(linetype = "none", shape = guide_legend(keyheight = 0.7, keywidth = 0.1, order = 2, nr
  guides(color = guide_legend(keyheight = 0.7, keywidth = 0.1, order = 1, nrow = 2, override.ae
  annotate("text", x = -0.150, y = -0.39, label = deparse(p_temp_pval), parse = TRUE, size = 3)
  annotate("text", x = -0.155, y = -0.33, label = deparse(p_pco2_pval), parse = TRUE, size = 3)
  theme(legend.background = element_rect(fill = "transparent", color = NA), legend.position = c

guide_temp_color <- get_legend(p_pca_plot + guides(color = "none")) # extract the temp colour l

p_pca_plot <- p_pca_plot + # add colour guide to new location and save
  guides(linetype = "none", shape = "none") +
  annotation_custom(guide_temp_color, xmax = 0.7, ymax = -0.24)



# reef
p_reef_pca <- autoplot(p_pca, data = p_df,
                       colour = "reef",
                       fill = "reef",
                       frame = TRUE,
                       frame.type = "t", # displaying ellipses with multivariate t-distributior
                       frame.level = 0.95, # using 95% CI for all ellipses
                       frame.alpha = 0.01,
                       loadings = TRUE,
                       loadings.colour = "grey29",
                       loadings.label = TRUE,
                       loadings.label.colour = "black",
                       loadings.label.size = 4,
                       loadings.label.hjust = 1.5,
                       loadings.label.vjust = 0.5,
                       loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b"
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  annotate("text", x = -0.3, y = -0.38, label = deparse(p_reef_pval), parse = TRUE, size = 3) +
  ggtitle(expression(paste(bold("A)    "), italic("S. siderea"))))

# pco2 * temp
p_df$pco2_f <- factor(p_df$pco2, labels = c("pre industrial: 28C", "pre industrial: 31C", "curr
p_df$pco2_f <- factor(p_df$pco2, labels = c("pre industrial; 28C", "pre industrial; 31C", "curr
p_pt_pca <- autoplot(p_pca, data = p_df,
                     colour = "pco2_f",
                     fill = "pco2_f",
                     shape = "pco2_f",
                     frame = TRUE,
                     frame.type = "t", # displaying ellipses with multivariate t-distributions
                     frame.level = 0.95, # using 95% CI for all ellipses
                     frame.alpha = 0.01,
                     frame.size = 5,
                     loadings = TRUE,
                     loadings.colour = "grey29",
                     loadings.label = TRUE,
                     loadings.label.colour = "black",
                     loadings.label.size = 4,
                     loadings.label.hjust = 1.5,
                     loadings.label.vjust = 0.5,
                     loadings.label.repel = TRUE) +
  scale_fill_manual("", values = c("#d1e5f0", "#fddbc7", "#f4a582", "#92c5de", "#4393c3", "#d6
  scale_colour_manual("", values = c("#d1e5f0", "#fddbc7", "#f4a582", "#92c5de", "#4393c3", "#c
  scale_shape_manual("", values = c(19, 19, 17, 17, 15, 15, 8, 8)) +
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 4, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",



### Trying a single PCA with colour for pCO2 and shape for temp:
# ellipse by temp, points by pCO2
# p_temp_pca2 <- autoplot(p_pca, data = p_df,
#          colour = "fpco2",
#          shape = "ftemp",
#          frame = FALSE,
#          loadings = TRUE,
#          loadings.colour = "grey29",
#          loadings.label = TRUE,
#          loadings.label.colour = "black",
#          loadings.label.size = 4,
#          loadings.label.hjust = 1.5,
#          loadings.label.vjust = 0.5,
#          loadings.label.repel = TRUE) +
#     stat_ellipse(type = "t", aes(lty = ftemp, colour = ftemp)) +
#     scale_linetype_manual(values = c("solid", "dashed")) +
#     scale_shape_manual("temperature", labels = c("28 C", "31 C"), values = c(19, 1)) +
#     scale_color_manual("", labels = c("28 C", "31 C", "pre industrial", "current", "end-of-ce
#     guides(linetype = "none", shape = "none") +
#     guides(color = guide_legend(nrow = 2, override.aes = list(linetype = c(1, 2, 0, 0, 0, 0),
#     theme(legend.background = element_rect(fill = "transparent", color = NA), legend.positior
```

PCA for species Porites astreoides

PCA for species Porites astreoides

Taken from OA_OW_Physiology_manuscript.Rmd lines 579 - 605

```
# set up the dataframe
a_df <- unique(a_df)
a_df <- a_df[-17,] # we have two from the same colony in 400_28 that are performing similarly s
a_df_l <- gather(a_df, param, value, c(14:16,21:23))
a_df$fpco2 <- factor(a_df$fpco2, levels = c("300", "420", "680", "3290"))
por_pca_df <- a_df[,c(14:16,18,21:23)]
por_pca_df <- rename(por_pca_df, colour = red) # renaming the 'sum' column to 'colour'

# run the adonis
a_pca_mod_full <- adonis2(por_pca_df ~ fpco2 * ftemp * reef, data = a_df, method = 'eu', permut
a_pca_mod <- adonis2(por_pca_df ~ reef + ftemp + fpco2, data = a_df, method = 'eu', permutation
a_pca_mod # view PAST adonis output

# pull AIC from the full and reduced PERMANOVA models
a_pca_aic_full <- round(AICc.PERMANOVA2(a_pca_mod_full)[[1]], 1)
a_pca_aic_final <- round(AICc.PERMANOVA2(a_pca_mod)[[1]], 1)



# extract pvalues
a_pval <- a_pca_mod["Pr(>F)"]

# perform principal component analysis (PCA)
a_pca <- prcomp(por_pca_df, center = TRUE, scale= TRUE)
```

Error will be solved with dplyr specification

```
# set up the dataframe
a_df <- unique(a_df)
a_df <- a_df[-17,] # we have two from the same colony in 400_28 that are performing similarly s
a_df_l <- gather(a_df, param, value, c(14:16,21:23))
a_df$fpco2 <- factor(a_df$fpco2, levels = c("300", "420", "680", "3290"))
por_pca_df <- a_df[,c(14:16,18,21:23)]
por_pca_df <- dplyr::rename(por_pca_df, colour = red) # renaming the 'sum' column to 'colour'

# run the adonis
a_pca_mod_full <- adonis2(por_pca_df ~ fpco2 * ftemp * reef, data = a_df, method = 'eu', permut
a_pca_mod <- adonis2(por_pca_df ~ reef + ftemp + fpco2, data = a_df, method = 'eu', permutation
a_pca_mod # view PAST adonis output
```

```
Permutation test for adonis under reduced model
Marginal effects of terms
Permutation: free
Number of permutations: 2000

adonis2(formula = por_pca_df ~ reef + ftemp + fpco2, data = a_df, permutations = bootnum,
method = "eu", by = "margin")
```

```
         Df SumOfSqs      R2       F    Pr(>F)    
reef      1      724 0.00512  0.5264 0.4717641    
ftemp     1    27051 0.19129 19.6601 0.0004998 ***
fpco2     3    30537 0.21594  7.3978 0.0004998 ***
Residual 62    85309 0.60325                      
Total    67   141417 1.00000                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# pull AIC from the full and reduced PERMANOVA models
a_pca_aic_full <- round(AICc.PERMANOVA2(a_pca_mod_full)[[1]], 1)
a_pca_aic_final <- round(AICc.PERMANOVA2(a_pca_mod)[[1]], 1)



# extract pvalues
a_pval <- a_pca_mod["Pr(>F)"]

# perform principal component analysis (PCA)
a_pca <- prcomp(por_pca_df, center = TRUE, scale= TRUE)
```

PCA Plot

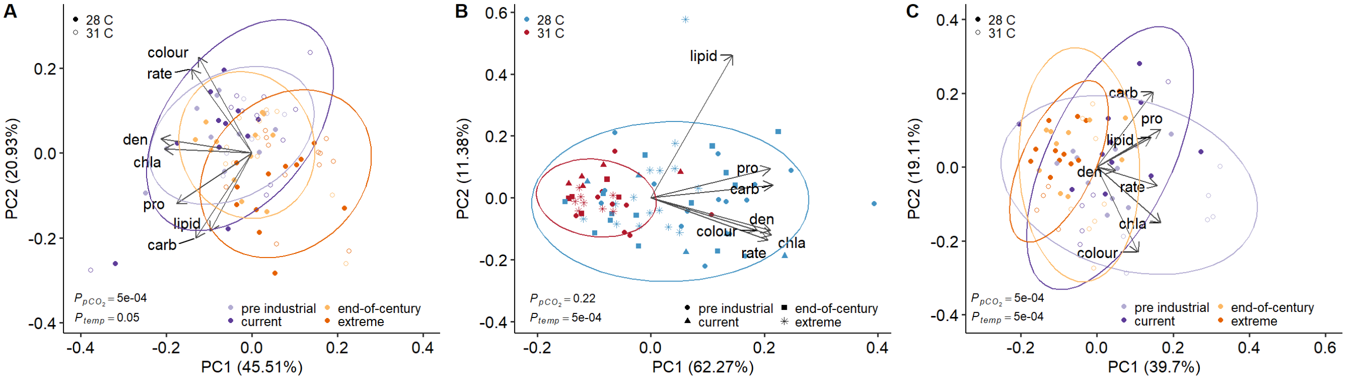
Taken from OA_OW_Physiology_manuscript.Rmd lines 607 - 714

```
# create labels for p values calculated above
a_reef_pval <-substitute(italic(P[reef])==p, list(p = format(a_pval[1,1], digits = 2)))
a_temp_pval <-substitute(italic(P[temp])==p, list(p = format(a_pval[2,1], digits = 1)))
a_pco2_pval <-substitute(italic(P[pCO[2]])==p, list(p = format(a_pval[3,1], digits = 1)))



# temperature = shape; pco2 = colours
a_pca_plot <- autoplot(a_pca, data = a_df,
                       colour = "fpco2",
                       shape = "ftemp",
                       frame = FALSE,
                       loadings = TRUE,
                       loadings.colour = "grey29",
                       loadings.label = TRUE,
                       loadings.label.colour = "black",
                       loadings.label.size = 4,
                       loadings.label.hjust = 1.5,
                       loadings.label.vjust = 0.5,
                       loadings.label.repel = TRUE) +
  stat_ellipse(type = "t", aes(colour = fpco2)) +
  scale_shape_manual("", labels = c("28 C", "31 C"), values = c(19, 1)) +
  scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  lims(y = c(-0.41, 0.4), x = c(-0.3, 0.6)) +
  guides(linetype = "none", shape = guide_legend(keyheight = 0.7, keywidth = 0.1, order = 1, nr
  guides(color = guide_legend(keyheight = 0.7, keywidth = 0.1, order = 2, nrow = 2, override.ae
  annotate("text", x = -0.235, y = -0.408, label = deparse(a_temp_pval), parse = TRUE, size = 3
  annotate("text", x = -0.235, y = -0.358, label = deparse(a_pco2_pval), parse = TRUE, size = 3
  theme(legend.background = element_rect(fill = "transparent", color = NA), legend.position = c

a_pca_plot <- a_pca_plot + # add colour guide to new location and save
  guides(color = "none") +
  annotation_custom(guide_pco2_color, xmax = 0.98, ymax = -0.28)


# reef
a_reef_pca <- autoplot(a_pca, data = a_df,
                       colour = "reef",
                       fill = "reef",
                       frame = TRUE,
                       frame.type = "t", # displaying ellipses with multivariate t-distribution
                       frame.level = 0.95, # using 95% CI for all ellipses
                       frame.alpha = 0.01,
                       loadings = TRUE,
                       loadings.colour = "grey29",
                       loadings.label = TRUE,
                       loadings.label.colour = "black",
                       loadings.label.size = 4,
                       loadings.label.hjust = 1.5,
                       loadings.label.vjust = 0.5,
                       loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b'
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  annotate("text", x = -0.3, y = -0.38, label = deparse(a_reef_pval), parse = TRUE, size = 3) +
  ggtitle(expression(paste(bold("A)   "), italic("S. siderea"))))

# pco2 * temp
a_df$pco2_f <- factor(a_df$pco2, labels = c("pre industrial; 28C", "pre industrial; 31C", "curr
a_pt_pca <- autoplot(a_pca, data = a_df,
                     colour = "pco2_f",
                     fill = "pco2_f",
                     shape = "pco2_f",
                     frame = TRUE,
                     frame.type = "t", # displaying ellipses with multivariate t-distributions
                     frame.level = 0.95, # using 95% CI for all ellipses
                     frame.alpha = 0.01,
                     frame.size = 5,
                     loadings = TRUE,
                     loadings.colour = "grey29",
                     loadings.label = TRUE,
                     loadings.label.colour = "black",
                     loadings.label.size = 4,
                     loadings.label.hjust = 1.5,
                     loadings.label.vjust = 0.5,
                     loadings.label.repel = TRUE) +
  scale_fill_manual("", values = c("#d1e5f0", "#fddbc7", "#f4a582", "#92c5de", "#4393c3", "#d6(
  scale_colour_manual("", values = c("#d1e5f0", "#fddbc7", "#f4a582", "#92c5de", "#4393c3", "#(
  scale_shape_manual("", values = c(19, 19, 17, 17, 15, 15, 8, 8)) +
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 4, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",


### Trying a single PCA with colour for pCO2 and shape for temp:
# ellipse by temp, points by pCO2
# a_temp_pca2 <- autoplot(a_pca, data = a_df,
#          colour = "fpco2",
#          shape = "ftemp",
#          frame = FALSE,
#          loadings = TRUE,
#          loadings.colour = "grey29",
#          loadings.label = TRUE,
#          loadings.label.colour = "black",
#          loadings.label.size = 4,
#          loadings.label.hjust = 1.5,
#          loadings.label.vjust = 0.5,
#          loadings.label.repel = TRUE) +
#     stat_ellipse(type = "t", aes(lty = ftemp, colour = ftemp)) +
#     scale_linetype_manual(values = c("solid", "dashed")) +
#     scale_shape_manual("temperature", labels = c("28 C", "31 C"), values = c(19, 1)) +
#     scale_color_manual("", labels = c("28 C", "31 C", "pre industrial", "current", "end-of-ce
#     guides(linetype = "none", shape = "none") +
#     guides(color = guide_legend(nrow = 2, override.aes = list(linetype = c(1, 2, 0, 0, 0, 0),
#     theme(legend.background = element_rect(fill = "transparent", color = NA), legend.positior
```

Figure 1

Taken from OA_OW_Physiology_manuscript.Rmd lines 722 - 733

```
ggarrange(s_pca_plot, p_pca_plot, a_pca_plot, ncol = 3, labels = c("A", "B", "C")) +
  theme(plot.background = element_rect(fill = "white", colour = NA))
```



```
ggsave("Figures/Final_Figures/Figure1_PhysPCA.pdf", width = 14, height = 4, useDingbats=FALSE
ggsave("Figures/Final_Figures/Figure1_PhysPCA.png", width = 14, height = 4, dpi = 650)
ggsave("Figures/Final_Figures/Figure1_PhysPCA.tiff", width = 14, height = 4, dpi = 650)
```

```
Warning in grDevices::dev.off(): agg could not write to the given file
```

These figures are identical to those in the paper.

Total PCA

Taken from OA_OW_Physiology_manuscript.Rmd lines 741 - 760

```
# set up the dataframe
all_df <- unique(df_90) # remove any duplicate rows
all_df_l <- gather(all_df, param, value, c(14:17,21:23))
all_df$fpco2 <- factor(all_df$fpco2, levels = c("300", "420", "680", "3290"))
all_pca_df <- all_df[,c(14:17,21:23)]

# run the adonis
#all_pca_mod <- adonis2(all_pca_df ~ reef * ftemp * fpco2 * species, data = all_df, method = 'e
all_pca_mod <- adonis2(all_pca_df ~ fpco2 + ftemp + reef + species + ftemp:species + fpco2:spec
all_pca_mod # view all adonis output
```

```
Permutation test for adonis under reduced model
Permutation: free
Number of permutations: 2000

adonis2(formula = all_pca_df ~ fpco2 + ftemp + reef + species + ftemp:species + fpco2:species
+ reef:species, data = all_df, permutations = bootnum, method = "eu")
          Df SumOfSqs      R2    F    Pr(>F)    
Model     17  2588852 0.66688 25.2 0.0004998 ***
Residual 214  1293204 0.33312                   
Total    231  3882055 1.00000                   
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# extract pvalues
all_pval <- all_pca_mod["Pr(>F)"]

# perform principal component analysis (PCA)
all_pca <- prcomp(all_pca_df, center = TRUE, scale= TRUE)
```

Total PCA Plot

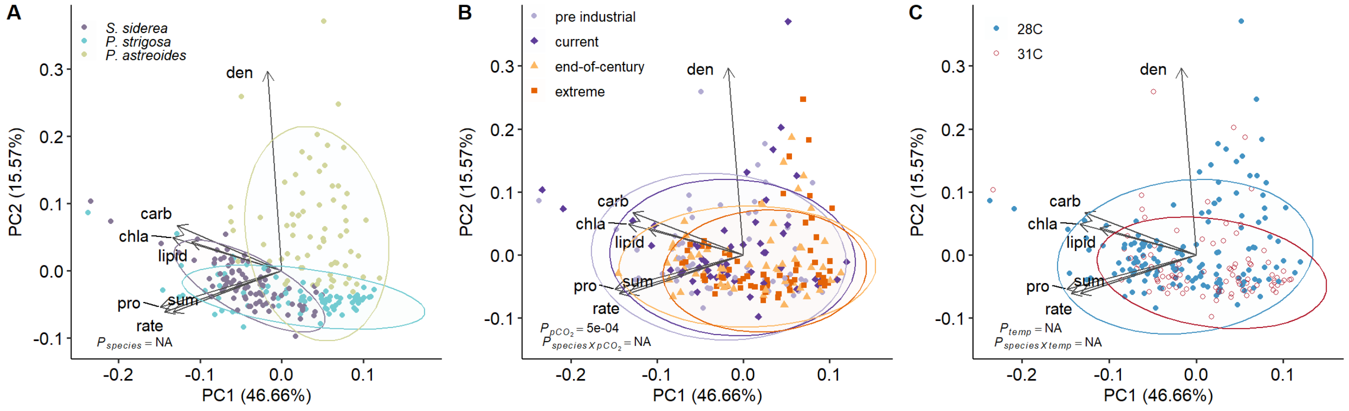
Taken from OA_OW_Physiology_manuscript.Rmd lines 762 - 849

```
# create labels for p values calculated above
all_pco2_pval <-substitute(italic(P[pCO[2]])==p, list(p = format(all_pval[1,1], digits = 1)))
all_temp_pval <-substitute(italic(P[temp])==p, list(p = format(all_pval[2,1], digits = 2)))
all_temp_pval <-substitute(italic(P[temp])==p, list(p = format(all_pval[2,1], digits = 1)))
all_reef_pval <-substitute(italic(P[reef])==p, list(p = format(all_pval[3,1], digits = 1)))
all_species_pval <-substitute(italic(P[species])==p, list(p = format(all_pval[4,1], digits = 1)
all_s_pco2_pval <-substitute(italic(P[species~X~pCO[2]])==p, list(p = format(all_pval[5,1], dig
all_s_reef_pval <-substitute(italic(P[species~X~reef])==p, list(p = format(all_pval[6,1], digit
all_s_temp_pval <-substitute(italic(P[species~X~temp])==p, list(p = format(all_pval[7,1], digit


# species
species_pca_plot <- autoplot(all_pca, data = all_df,
                       colour = "species",
                       #shape = "species",
                       #shape = "fpco2",
                       #fill = "species",
                       frame = TRUE,
                       frame.type = "t", # displaying ellipses with multivariate t-distributior
                       frame.level = 0.95, # using 95% CI for all ellipses
                       frame.alpha = 0.01,
                       loadings = TRUE,
                       loadings.colour = "grey29",
                       loadings.label = TRUE,
                       loadings.label.colour = "black",
                       loadings.label.size = 4,
                       loadings.label.hjust = 1.5,
                       loadings.label.vjust = 0.5,
                       loadings.label.repel = TRUE) +
  scale_color_manual("", labels = c("S. siderea", "P. strigosa", "P. astreoides"), values = c('
  guides(shape = FALSE, fill = FALSE, color = guide_legend(keyheight = 0.1, nrow = 3, byrow = T
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  annotate("text", x = -0.18, y = -0.108, label = deparse(all_species_pval), parse = TRUE, size

# temperature
all_temp_pca <- autoplot(all_pca, data = all_df,
                       colour = "ftemp",
                       #shape = "fpco2",
                       shape = "ftemp",
                       fill = "ftemp",
                       frame = TRUE,
                       frame.type = "t", # displaying ellipses with multivariate t-distributior
                       frame.level = 0.95, # using 95% CI for all ellipses
                       frame.alpha = 0.01,
                       loadings = TRUE,
                       loadings.colour = "grey29",
                       loadings.label = TRUE,
                       loadings.label.colour = "black",
                       loadings.label.size = 4,
                       loadings.label.hjust = 1.5,
                       loadings.label.vjust = 0.5,
                       loadings.label.repel = TRUE) +
  scale_shape_manual("temperature", labels = c("28C", "31C"), values = c(21, 21)) +
  scale_color_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "#b2182b"))
  scale_color_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "#b2182b"),
  scale_fill_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "transparent
  guides(color = guide_legend(keyheight = 0.3, nrow = 4, byrow = TRUE, override.aes = list(line
  theme(legend.position = c(0.11, 0.9), legend.title = element_blank(), legend.background = ele
  annotate("text", x = -0.19, y = -0.118, label = deparse(all_temp_pval), parse = TRUE, size =
  annotate("text", x = -0.17, y = -0.14, label = deparse(all_s_temp_pval), parse = TRUE, size =

# pco2
all_pco2_pca <- autoplot(all_pca, data = all_df,
                         colour = "fpco2",
                         #shape = "ftemp",
                         fill = "fpco2",
                         shape = "fpco2",
                         frame = TRUE,
                         frame.type = "t", # displaying ellipses with multivariate t-distributior
                         frame.level = 0.95, # using 95% CI for all ellipses
                         frame.alpha = 0.01,
                         loadings = TRUE,
                         loadings.colour = "grey29",
                         loadings.label = TRUE,
                         loadings.label.colour = "black",
                         loadings.label.size = 4,
                         loadings.label.hjust = 1.5,
                         loadings.label.vjust = 0.5,
                         loadings.label.repel = TRUE) +
  scale_shape_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  scale_fill_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"), v
  scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  guides(color = guide_legend(nrow = 4, override.aes = list(linetype = c(0, 0, 0, 0)))) +
  annotate("text", x = -0.19, y = -0.118, label = deparse(all_pco2_pval), parse = TRUE, size =
  annotate("text", x = -0.17, y = -0.14, label = deparse(all_s_pco2_pval), parse = TRUE, size =
  theme(legend.position = c(0.18, 0.9), legend.background = element_rect(fill = "transparent",
```

Figure 4

Taken from OA_OW_Physiology_manuscript.Rmd lines 851 - 858

```
ggarrange(species_pca_plot, all_pco2_pca, all_temp_pca, nrow =1, labels = "AUTO")
```



```
  ggsave("Figures/Final_Figures/Figure4_SpeciesPCA.pdf", width = 13, height = 4, useDingbats=FA
  ggsave("Figures/Final_Figures/Figure4_SpeciesPCA.png", width = 13, height = 4, dpi = 650)
  ggsave("Figures/Final_Figures/Figure4_SpeciesPCA.tiff", width = 13, height = 4, dpi = 650)
```

Correlation Matrices and Scatter Plots

Correlation Matrix SSID

Taken from OA_OW_Physiology_manuscript.Rmd lines 893 - 914

```
## Create new correlation dataframe for SSID
sid_corr_df <- s_df[,c(10:11,14:17,21:23)] %>%
  mutate(shape = case_when(ftemp == "28" ~ "19",
                           ftemp == "31" ~ "21"),
         color = case_when(fpco2 == "300" ~ "#b2abd2",
                            fpco2 == "420" ~ "#5e3c99",
                            fpco2 == "680" ~ "#fdb863",
                            fpco2 == "3290" ~ "#e66101"))

## Plot correlation matrix and scatter plot of all SSID physiology
png(file = "Figures/Supplemental_Figures/SSID_PhysCorrelations.png", width = 240.57, height = 1

corrgram(sid_corr_df, order = FALSE, lower.panel = panel.fill.R2, upper.panel = panel.pts.col,
         col.regions = colorRampPalette(c("#ffffcc", "#c7e9b4", "#225ea8")),
         color = sid_corr_df$color, pch = as.numeric(as.character(sid_corr_df$shape)),
         labels = c("symbiont \ndensity", "host \nprotein", "chlorophyll a", "color \nintensity

dev.off()
```

Passing parameter %>% brings up error. Switching to pipe operator |> bypasses the error. Mutate function not found, fixed by specifying dplyr package. The same procedure was executed to fix the error on the case_when function. PNG unable to open, so I created a folder as described in line 1104 to allow png to resume. Corrgram function not found, so I specified that it was inside the corrgram package and the chunk fully ran.

```
## Create new correlation dataframe for SSID
sid_corr_df <- s_df[,c(10:11,14:17,21:23)] |>
  dplyr::mutate(shape = dplyr::case_when(ftemp == "28" ~ "19",
                           ftemp == "31" ~ "21"),
         color = dplyr::case_when(fpco2 == "300" ~ "#b2abd2",
                            fpco2 == "420" ~ "#5e3c99",
                            fpco2 == "680" ~ "#fdb863",
                            fpco2 == "3290" ~ "#e66101"))

## Plot correlation matrix and scatter plot of all SSID physiology
## Plot correlation matrix and scatter plot of all SSID physiology
png(file = "Figures/Supplemental_Figures/SSID_PhysCorrelations.png", width = 240.57, height = 1

corrgram::corrgram(sid_corr_df, order = FALSE, lower.panel = panel.fill.R2, upper.panel = panel
        col.regions = colorRampPalette(c("#ffffcc", "#c7e9b4", "#225ea8")),
        color = sid_corr_df$color, pch = as.numeric(as.character(sid_corr_df$shape)),
        labels = c("symbiont \ndensity", "host \nprotein", "chlorophyll a", "color \nintensity

dev.off()
```

Correlation Matrix PSTR

Taken from OA_OW_Physiology_manuscript.Rmd lines 916 - 938

```
## Create new correlation dataframe for SSID
dip_corr_df <- p_df[,c(10:11,14:17,21:23)] %>%
  mutate(shape = case_when(ftemp == "28" ~ "19",
                           ftemp == "31" ~ "21"),
        color = case_when(fpco2 == "300" ~ "#b2abd2",
                            fpco2 == "420" ~ "#5e3c99",
                            fpco2 == "680" ~ "#fdb863",
                            fpco2 == "3290" ~ "#e66101"))



## Plot correlation matrix and scatter plot of all PSTR physiology
png(file = "Figures/Supplemental_Figures/PSTR_PhysCorrelations.png", width = 240.57, height = 1

corrgram(dip_corr_df, order = FALSE, lower.panel=panel.fill.R2, upper.panel=panel.pts.col, text
        col.regions=colorRampPalette(c("#ffffcc", "#c7e9b4", "#225ea8")),
        color = dip_corr_df$color, pch = as.numeric(as.character(dip_corr_df$shape)),
        labels = c("symbiont \ndensity", "host \nprotein", "chlorophyll a", "color \nintensity

dev.off()
```

Same adjustments were made as in the previous example.

```
## Create new correlation dataframe for SSID
dip_corr_df <- p_df[,c(10:11,14:17,21:23)] |>
  dplyr::mutate(shape = dplyr::case_when(ftemp == "28" ~ "19",
                           ftemp == "31" ~ "21"),
        color = dplyr::case_when(fpco2 == "300" ~ "#b2abd2",
                            fpco2 == "420" ~ "#5e3c99",
                            fpco2 == "680" ~ "#fdb863",
                            fpco2 == "3290" ~ "#e66101"))



## Plot correlation matrix and scatter plot of all PSTR physiology
png(file = "Figures/Supplemental_Figures/PSTR_PhysCorrelations.png", width = 240.57, height = 1

corrgram::corrgram(dip_corr_df, order = FALSE, lower.panel=panel.fill.R2, upper.panel=panel.pts
corrgram::corrgram(dip_corr_df, order = FALSE, lower.panel=panel.fill.R2, upper.panel=panel.pts
         col.regions=colorRampPalette(c("#ffffcc", "#c7e9b4", "#225ea8")),
         color = dip_corr_df$color, pch = as.numeric(as.character(dip_corr_df$shape)),
         labels = c("symbiont \ndensity", "host \nprotein", "chlorophyll a", "color \nintensity

dev.off()
```

Correlation matrix PAST

Taken from OA_OW_Physiology_manuscript.Rmd lines 940 - 962

```
## Create new correlation dataframe for SSID
por_corr_df <- a_df[,c(10:11,14:16,18,21:23)] %>%
  mutate(shape = case_when(ftemp == "28" ~ "19",
                           ftemp == "31" ~ "21"),
         color = case_when(fpco2 == "300" ~ "#b2abd2",
                            fpco2 == "420" ~ "#5e3c99",
                            fpco2 == "680" ~ "#fdb863",
                            fpco2 == "3290" ~ "#e66101"))



## Plot correlation matrix and scatter plot of all PAST physiology
png(file = "Figures/Supplemental_Figures/PAST_PhysCorrelations.png", width = 240.57, height = 1

corrgram(por_corr_df, order = FALSE, lower.panel=panel.fill.R2, upper.panel=panel.pts.col, text
         col.regions=colorRampPalette(c("#ffffcc", "#c7e9b4", "#225ea8")),
         color = por_corr_df$color, pch = as.numeric(as.character(por_corr_df$shape)),
         labels = c("symbiont \ndensity", "host \nprotein", "chlorophyll a", "color \nintensity

dev.off()
```

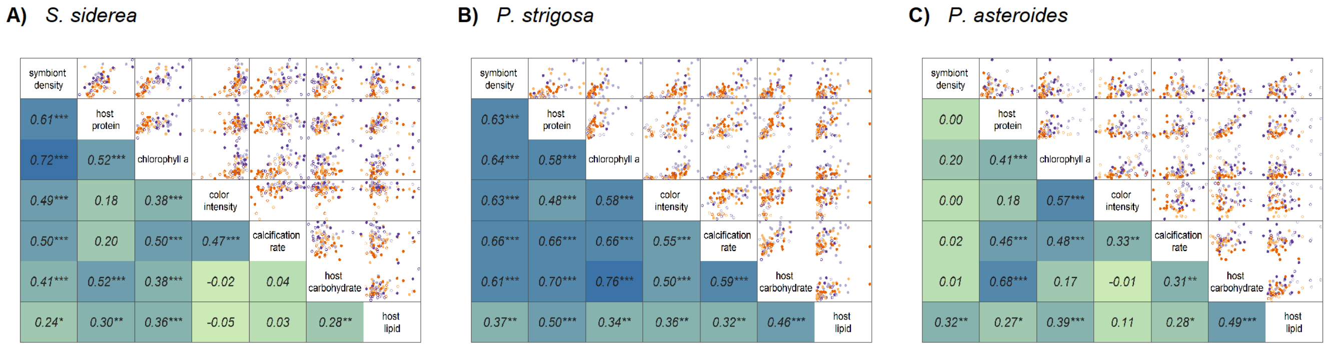
Same adjustments were made as in previous.

```
## Create new correlation dataframe for SSID
por_corr_df <- a_df[,c(10:11,14:16,18,21:23)] |>
  dplyr::mutate(shape = dplyr::case_when(ftemp == "28" ~ "19",
                           ftemp == "31" ~ "21"),
         color = dplyr::case_when(fpco2 == "300" ~ "#b2abd2",
                            fpco2 == "420" ~ "#5e3c99",
                            fpco2 == "680" ~ "#fdb863",
                            fpco2 == "3290" ~ "#e66101"))



## Plot correlation matrix and scatter plot of all PAST physiology
png(file = "Figures/Supplemental_Figures/PAST_PhysCorrelations.png", width = 240.57, height = 1

corrgram::corrgram(por_corr_df, order = FALSE, lower.panel=panel.fill.R2, upper.panel=panel.pts
         col.regions=colorRampPalette(c("#ffffcc", "#c7e9b4", "#225ea8")),
         color = por_corr_df$color, pch = as.numeric(as.character(por_corr_df$shape)),
         labels = c("symbiont \ndensity", "host \nprotein", "chlorophyll a", "color \nintensity
         labels = c("symbiont \ndensity", "host \nprotein", "chlorophyll a", "color \nintensity

dev.off()
```

Figure 2

Taken from OA_OW_Physiology_manuscript.Rmd lines 967 - 993

```
ssid_corr_plot <- readPNG("Figures/Supplemental_Figures/SSID_PhysCorrelations.png")
pstr_corr_plot <- readPNG("Figures/Supplemental_Figures/PSTR_PhysCorrelations.png")
past_corr_plot <- readPNG("Figures/Supplemental_Figures/PAST_PhysCorrelations.png")

ssid_corr_plot <- ggplot() +
  background_image(ssid_corr_plot) +
  theme_void() +
  ggtitle(expression(paste(bold("  A)   "), italic("S. siderea"))))

pstr_corr_plot <- ggplot() +
  background_image(pstr_corr_plot) +
  theme_void() +
  ggtitle(expression(paste(bold("  B)   "), italic("P. strigosa"))))

past_corr_plot <- ggplot() +
  background_image(past_corr_plot) +
  theme_void() +
  ggtitle(expression(paste(bold("  C)   "), italic("P. asteroides"))))

ggarrange(ssid_corr_plot, pstr_corr_plot, past_corr_plot, nrow = 1)
```



```
  ggsave("Figures/Final_Figures/Figure2_PhysCorrelations.pdf", width = 12, height = 3.1, useDir
  ggsave("Figures/Final_Figures/Figure2_PhysCorrelations.png", width = 12, height = 3.1, dpi =
  ggsave("Figures/Final_Figures/Figure2_PhysCorrelations.tiff", width = 12, height = 3.1, dpi =
```

Plasticity

Siderastrea

Siderastrea

SSID Distance analysis

Taken from OA_OW_Physiology_manuscript.Rmd lines 1025 - 1073

```
## Calculate the PCA distance with custom function
sid_dist <- PCAplast(pca = s_pca, # the PCA dataframe containing the PCA eigenvalues
          data = s_df[,c(1,7,8, 10,11,12,27,2)], # the condition/treatment data corresponding t
          sample_ID = "coral", # the name of column that provide unique ID per sample (if blank
          num_pca =  "all", # the number of PCAs to include in analysis (default is 'all', but
          control_col = "treat2", # what the 'treatment' column is called
          control_lvl = "420_28", # control level of the treatment
          group = "colony") # the grouping column (i.e., colony)



# with strong random effects of colony??
# marginal R2 --> fixed v random effect variance: likely more variance at colony level over tre

ssid_dist_mod <- glmer(dist ~ reef * fpco2 * ftemp + (1 | colony), family = Gamma(link = "log")
```

```
fixed-effect model matrix is rank deficient so dropping 2 columns / coefficients
```

```
ssid_dist_mod2 <- glmer(dist ~ reef * fpco2 + ftemp + (1 | colony), family = Gamma(link = "log'
ssid_dist_mod2b <- glmer(dist ~ reef * fpco2 + ftemp + (1 | colony) + (1 | tank), family = Gamm
ssid_dist_mod3 <- glmer(dist ~ reef + fpco2 * ftemp + (1 | colony), family = Gamma(link = "log'
```

```
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
```

```
ssid_dist_mod4 <- glmer(dist ~ reef + fpco2 + ftemp + (1 | colony), family = Gamma(link = "log'
ssid_dist_mod5 <- glmer(dist ~ reef * (fpco2 + ftemp) + (1 | colony), family = Gamma(link = "lo
ssid_dist_mod6 <- glmer(dist ~ fpco2 + ftemp + (1 | colony), family = Gamma(link = "log"), data

# check for best-fit model
ssid_plast_aic <- compare_performance(ssid_dist_mod, ssid_dist_mod2, ssid_dist_mod2b, ssid_dist
#plot(ssid_plast_aic)



## Best-fit GLMM with Gamma log link
ssid_dist_glm <- glmer(dist ~ reef * fpco2 + ftemp + (1 | colony) + (1 | tank), family = Gamma(
ssid_glm_out <- summary(ssid_dist_glm) # summary output of the GLM



## conditional and marginal R2
r2_nakagawa(ssid_dist_glm)
```

```
# R2 for Mixed Models

  Conditional R2: 0.553
```

```
     Marginal R2: 0.313
```

```
#  Conditional R2: 0.542 -- fixed and random effects
#     Marginal R2: 0.307 -- fixed effects only

## Below info taken from appendix 3 of Gok A, Ngendahimana DK, Fagerholm CL, French RH, Sun J,

# "Marginal and conditional are R2 values for generalized mixed-effects models
# calculated using the r.squaredGLMM function of the MuMIn [1] package that
# implements a method developed by Nakagawa and Schielzeth [2]. Marginal R2
# provides the variance explained only by fixed effects and conditional R2
# provides the variance explained by the entire model, i.e., both fixed effects
# and random effects. Fitted R2 is analogous to adjusted R2 generalized for
# measuring explained variation in linear mixed-effects models [3]."
```

Plenty of warnings with dropped columns. If this interferes with later steps, then may cause issues. For now, this does not need to be alterred.

Bootstrapping

Taken from OA_OW_Physiology_manuscript.Rmd lines 1075 - 1090

```
## Pull the treatment and distance data from the model
newdata_ssid <- data.frame(ssid_dist_glm@frame[-1])

## Bootstrap distances using custom bootstrap function
ssid_boot_out <- replicate(bootnum, bootFUN(model = ssid_dist_glm, newdata = newdata_ssid))

## Calculate the mean, 95% lowerCI, and 95% upperCI from the boot matrix and add it to datafram
ssid_boot <- cbind(sid_dist, as.data.frame(t(apply(ssid_boot_out, 1, function(x) c(mean(x), qua
colnames(ssid_boot)[10:12] <- c("estimate", "lowerci", "upperci") # rename mean/CI columns

## save as rda object
save(ssid_boot, file = "Data/Bootstrap/SSID_PlastBoot.rda")
```

Running this code chunk takes an incredibly long amount of time. EVAL should be set to false, but to ensure the process is correct, all should be run once to ensure it matches what is known.

In order to allow data to save, created folders within my directory to allow the file path to execute

Distance Plot

Taken from OA_OW_Physiology_manuscript.Rmd lines 1092 - 1123

```
## Load boot data
load(file = "Data/Bootstrap/SSID_PlastBoot.rda")
ssid_boot$treat2 <- paste(ssid_boot$fpco2, ssid_boot$ftemp, sep = "_") # need to rename treatme
ssid_boot$treat2 <- factor(ssid_boot$treat2,
                           levels = c("300_28", "300_31", "420_31", "680_28", "680_31", "3290_2
                           labels = c("pre industrial; 28C", "pre industrial; 31C", "current; 3


## Plasticity plot
ssid_plast_plot <- ggplot(ssid_boot, aes(x = reef, y = estimate, color = treat2, fill = treat2,
  theme_pubr() +
  theme(legend.title = element_blank(), axis.ticks.x = element_blank()) +
  guides(shape = guide_legend(ncol = 1), fill = guide_legend(override.aes = list(linetype = 0))
  geom_point(aes(x = reef, y = dist), size = 2, alpha = 0.4, position = position_jitterdodge(ji
  geom_linerange(aes(ymin = lowerci, ymax =upperci), size = 1, position = position_dodge(width
  geom_point(size = 3, alpha = 0.8, stroke = 1, position = position_dodge(width = 0.55)) +
  scale_shape_manual(values = c(21, 21, 23, 24, 24, 22, 22)) +
  scale_color_manual(values = c("#b2abd2", "#b2abd2", "#5e3c99", "#fdb863", "#fdb863", "#e66101
  scale_fill_manual(values = c("#b2abd2", "white", "white", "#fdb863", "white", "#e66101", "whi
  labs(y = "PC distance from control", x = "") +
  scale_y_continuous(expand = c(0, 0), limits = c(0, 8)) +
  scale_x_discrete(labels = c("F" = "Offshore", "N" = "Inshore")) +
  ggtitle(expression(paste(bold("A)   "), italic("S. siderea"))))


## Model output and R2
ssid_glm_out # summary output of the GLMER
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: Gamma  ( log )
Formula: dist ~ reef * fpco2 + ftemp + (1 | colony) + (1 | tank)
   Data: sid_dist

     AIC      BIC   logLik deviance df.resid
   218.4    245.4    -97.2    194.4       58

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.92121 -0.66739 -0.00235  0.42107  2.51040

Random effects:
 Groups   Name        Variance Std.Dev.
 tank     (Intercept) 0.01358  0.1165
 colony   (Intercept) 0.03209  0.1791
 Residual             0.08867  0.2978
Number of obs: 70, groups:  tank, 21; colony, 11

Fixed effects:
                 Estimate Std. Error  t value Pr(>|z|)
(Intercept)      1.050285   0.005612  187.162   <2e-16 ***
reefN            0.009748   0.005645    1.727   0.0842 .
fpco2420         0.378248   0.005645   67.008   <2e-16 ***
fpco2680         0.222661   0.005658   39.355   <2e-16 ***
fpco23290        0.443635   0.005627   78.839   <2e-16 ***
```

```
.
ftemp31          0.001807   0.005620    0.321   0.7479
reefN:fpco2420  -0.770755   0.005628 -136.941   <2e-16 ***
reefN:fpco2680  -0.445511   0.005611  -79.399   <2e-16 ***
reefN:fpco23290 -0.329679   0.005632  -58.535   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) reefN  fp2420 fp2680 f23290 ftmp31 rN:242 rN:268
reefN       -0.003
fpco2420     0.001 -0.002
fpco2680    -0.005  0.001 -0.001
fpco23290    0.003  0.001 -0.001  0.005
ftemp31     -0.005 -0.002  0.000 -0.005  0.001
rfN:fpc2420  0.001  0.001 -0.002  0.005 -0.001  0.001
rfN:fpc2680 -0.005 -0.002  0.002  0.003  0.005 -0.004  0.005
rfN:fp23290 -0.001 -0.002  0.002  0.001 -0.001  0.000  0.001  0.000
optimizer (Nelder_Mead) convergence code: 0 (OK)
Model failed to converge with max|grad| = 0.0126655 (tol = 0.002, component 1)
```

```
r2_nakagawa(ssid_dist_glm) # conditional and marginal R2
```

```
# R2 for Mixed Models

  Conditional R2: 0.553
     Marginal R2: 0.313
```

Pseudodiploria

PSTR distance analysis

Taken from OA_OW_Physiology_manuscript.Rmd lines 1131 - 1171

```
## Calculate the PCA distance with custom function
dip_dist <- PCAplast(pca = p_pca, # the PCA dataframe containing the PCA eigenvalues
          data = p_df[,c(1,7,8, 10,11,12,27,2)], # the condition/treatment data corresponding t
          sample_ID = "coral", # the name of column that provide unique ID per sample (if blank
          num_pca =  "all", # the number of PCAs to include in analysis (default is 'all', but
          control_col = "treat2", # what the 'treatment' column is called
          control_lvl = "420_28", # control level of the treatment
          group = "colony") # the grouping column (i.e., colony)


## Model selection (via AIC)
pstr_dist_mod <- glmer(dist ~ reef * fpco2 * ftemp + (1 | colony), family = Gamma(link = "log")
```

```
fixed-effect model matrix is rank deficient so dropping 4 columns / coefficients
```

```
pstr_dist_mod2 <- glmer(dist ~ reef * fpco2 + ftemp + (1 | colony), family = Gamma(link = "log'
```

```
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
```

```
pstr_dist_mod3 <- glmer(dist ~ reef + fpco2 * ftemp + (1 | colony), family = Gamma(link = "log'
```

```
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
```

```
#pstr_dist_mod4 <- glmer(dist ~ reef + fpco2 + ftemp + (1 | colony), family = Gamma(link = "log
#pstr_dist_mod5 <- glmer(dist ~ reef * (fpco2 + ftemp) + (1 | colony), family = Gamma(link = "l
pstr_dist_mod6 <- glmer(dist ~ fpco2 + ftemp + (1 | colony), family = Gamma(link = "log"), data

# check for best-fit model
pstr_plast_aic <- compare_performance(pstr_dist_mod, pstr_dist_mod2, pstr_dist_mod3, pstr_dist_
#plot(pstr_plast_aic)



##### NOTE: nly one sample remains in current day at 31C so need to drop this treatment (clarif
### Also, because N=2 inshore and N=3 offshore for PSTR, going to pool these by RZ for plastici
dip_dist <- filter(dip_dist, fpco2 != "420") %>% droplevels()



## Best-fit GLMM with Gamma log link
pstr_dist_glm <- glmer(dist ~ fpco2 + ftemp + (1 | colony), family = Gamma(link = "log"), data
pstr_glm_out <- summary(pstr_dist_glm) # summary output of the GLM

## conditional and marginal R2
r2_nakagawa(pstr_dist_glm)
```

```
# R2 for Mixed Models

  Conditional R2: 0.244
     Marginal R2: 0.198
```

```
#   Conditional R2: 0.232 -- fixed and random effects
#      Marginal R2: 0.188 -- fixed effects only
```

Bootstrapping

Taken from OA_OW_Physiology_manuscript.Rmd lines 1173 - 1188

```
## Pull the treatment and distance data from the model
newdata_pstr <- data.frame(pstr_dist_glm@frame[-1])

## Bootstrap distances using custom bootstrap function
pstr_boot_out <- replicate(bootnum, bootFUN(model = pstr_dist_glm, newdata = newdata_pstr))
```

```
## Calculate the mean, 95% lowerCI, and 95% upperCI from the boot matrix and add it to datafram
pstr_boot <- cbind(dip_dist, as.data.frame(t(apply(pstr_boot_out, 1, function(x) c(mean(x), qua
colnames(pstr_boot)[10:12] <- c("estimate", "lowerci", "upperci") # rename mean/CI columns

## save as rda object
save(pstr_boot, file = "Data/Bootstrap/PSTR_PlastBoot.rda")
```

Running this code chunk takes an incredibly long amount of time. EVAL should be set to false, but to ensure the process is correct, all should be run once to ensure it matches what is known.

Distance plot

Taken from OA_OW_Physiology_manuscript.Rmd lines 1190 - 1221

```
## Load boot data
load(file = "Data/Bootstrap/PSTR_PlastBoot.rda")
pstr_boot$treat2 <- paste(pstr_boot$fpco2, pstr_boot$ftemp, sep = "_") # need to rename treatme
pstr_boot$treat2 <- factor(pstr_boot$treat2,
                           levels = c("300_28", "300_31", "680_28", "680_31", "3290_28", "3290_
                           labels = c("pre industrial; 28C", "pre industrial; 31C", "end-of-cen



## Plasticity plot
pstr_plast_plot <- ggplot(pstr_boot, aes(x = species, y = estimate, color = treat2, fill = trea
  theme_pubr() +
  theme(legend.title = element_blank(), axis.ticks.x = element_blank()) +
  guides(shape = guide_legend(ncol = 1), fill = guide_legend(override.aes = list(linetype = 0))
  geom_point(aes(x = species, y = dist), size = 2, alpha = 0.4, position = position_jitterdodge
  geom_linerange(aes(ymin = lowerci, ymax =upperci), size = 1, position = position_dodge(width
  geom_point(size = 3, alpha = 0.8, stroke = 1, position = position_dodge(width = 0.55)) +
  scale_shape_manual(values = c(21, 21, 23, 24, 24, 22, 22)) +
  scale_color_manual(values = c("#b2abd2", "#b2abd2", "#5e3c99", "#fdb863", "#fdb863", "#e66101
  scale_fill_manual(values = c("#b2abd2", "white", "white", "#fdb863", "white", "#e66101", "whi
  labs(y = "", x = "") +
  scale_y_continuous(expand = c(0, 0), limits = c(0, 8)) +
  scale_x_discrete(labels = c("P" = " Reefs pooled")) +
  ggtitle(expression(paste(bold("B)   "), italic("P. strigosa"))))


## Model output and R2
pstr_glm_out # summary output of the GLM
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: Gamma  ( log )
Formula: dist ~ fpco2 + ftemp + (1 | colony)
   Data: dip_dist

     AIC      BIC   logLik deviance df.resid
```

```
    97.5    104.8    -42.7     85.5       19

Scaled residuals: 
     Min       1Q   Median       3Q      Max 
-1.70079 -0.72697  0.05779  0.77668  2.01138 

Random effects:
 Groups   Name        Variance Std.Dev.
 colony   (Intercept) 0.007898 0.08887 
 Residual             0.137879 0.37132 
Number of obs: 25, groups:  colony, 5

Fixed effects:
            Estimate Std. Error t value Pr(>|z|)    
(Intercept)  1.27912    0.14767   8.662   <2e-16 ***
fpco2680    -0.33844    0.19338  -1.750   0.0801 .  
fpco23290   -0.05892    0.18739  -0.314   0.7532    
ftemp31      0.22671    0.17289   1.311   0.1898    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
          (Intr) fp2680 f23290
fpco2680  -0.579              
fpco23290 -0.578  0.484       
ftemp31   -0.334  0.066  0.012
```

```
r2_nakagawa(pstr_dist_glm) # conditional and marginal R2
```

```
# R2 for Mixed Models

  Conditional R2: 0.244
     Marginal R2: 0.198
```

Porites

Past Distance Analysis

Taken from OA_OW_Physiology_manuscript.Rmd lines 1229 - 1264

```
## Calculate the PCA distance with custom function
por_dist <- PCAplast(pca = a_pca, # the PCA dataframe containing the PCA eigenvalues
          data = a_df[,c(1,7,8, 10,11,12,27, 2)], # the condition/treatment data corresponding
          sample_ID = "coral", # the name of column that provide unique ID per sample (if blank
          num_pca =  "all", # the number of PCAs to include in analysis (default is 'all', but
          control_col = "treat2", # what the 'treatment' column is called
          control_lvl = "420_28", # control level of the treatment
          group = "colony") # the grouping column (i.e., colony). If blank, will assume control
```

```
## Model selection (via AIC)
past_dist_mod <- glmer(dist ~ reef * fpco2 * ftemp + (1 | colony), family = Gamma(link = "log")
```

```
fixed-effect model matrix is rank deficient so dropping 3 columns / coefficients
```

```
past_dist_mod2 <- glmer(dist ~ reef * fpco2 + ftemp + (1 | colony), family = Gamma(link = "log'
past_dist_mod3 <- glmer(dist ~ reef + fpco2 * ftemp + (1 | colony), family = Gamma(link = "log'
```

```
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
```

```
past_dist_mod4 <- glmer(dist ~ reef + fpco2 + ftemp + (1 | colony), family = Gamma(link = "log'
past_dist_mod5 <- glmer(dist ~ reef * (fpco2 + ftemp) + (1 | colony), family = Gamma(link = "lo
past_dist_mod6 <- glmer(dist ~ fpco2 + ftemp + (1 | colony), family = Gamma(link = "log"), data
past_dist_mod6b <- glmer(dist ~ fpco2 + ftemp + (1 | colony) + (1 | tank), family = Gamma(link

# check for best-fit model
past_plast_aic <- compare_performance(past_dist_mod, past_dist_mod2, past_dist_mod3, past_dist_
#plot(past_plast_aic)



## Best-fit GLMM with Gamma log link
past_dist_glm <- glmer(dist ~ fpco2 + ftemp + (1 | colony) + (1 | tank), family = Gamma(link =
past_glm_out <- summary(past_dist_glm) # summary output of the GLM

## conditional and marginal R2
r2_nakagawa(past_dist_glm)
```

```
# R2 for Mixed Models

  Conditional R2: 0.500
     Marginal R2: 0.147
```

```
#  Conditional R2: 0.493 -- fixed and random effects
#     Marginal R2: 0.145 -- fixed effects only
```

Bootstrapping

Taken from OA_OW_Physiology_manuscript.Rmd lines 1266 - 1281

```
## Pull the treatment and distance data from the model
newdata_past <- data.frame(past_dist_glm@frame[-1])

## Bootstrap distances using custom bootstrap function
past_boot_out <- replicate(bootnum, bootFUN(model = past_dist_glm, newdata = newdata_past))

## Calculate the mean, 95% lowerCI, and 95% upperCI from the boot matrix and add it to datafram
past_boot <- cbind(por_dist, as.data.frame(t(apply(past_boot_out, 1, function(x) c(mean(x), qua
colnames(past_boot)[10:12] <- c("estimate", "lowerci", "upperci") # rename mean/CI columns
```

```
## save as rda object
save(past_boot, file = "Data/Bootstrap/PAST_PlastBoot.rda")
```

Received error of non-conformable arguments on line 1540. Going line by line through bootFUN in custom functions doc, nothing fails. Made an attempt to directly put the function into the code chunk. The code ran with no errors, but the data points were different than the original figure and the confidence intervals were lost for this species. The other species came out as expected.

This is a major issue, which must be resolved, but since we have the data file that the author used, we will proceed with the included file.

Distance Plot

Taken from OA_OW_Physiology_manuscript.Rmd lines 1283 - 1314

```
## Load boot data
load(file = "Data/Bootstrap/PAST_PlastBoot.rda")
past_boot$treat2 <- paste(past_boot$fpco2, past_boot$ftemp, sep = "_") # need to rename treatme
past_boot$treat2 <- factor(past_boot$treat2,
                           levels = c("300_28", "300_31", "420_31", "680_28", "680_31", "3290_
                           labels = c("pre industrial; 28C", "pre industrial; 31C", "current; 3


## Plasticity plot
past_plast_plot <- ggplot(past_boot, aes(x = species, y = estimate, color = treat2, fill = trea
  theme_pubr() +
  theme(legend.title = element_blank(), axis.ticks.x = element_blank()) +
  guides(shape = guide_legend(ncol = 1), fill = guide_legend(override.aes = list(linetype = 0))
  geom_point(aes(x = species, y = dist), size = 2, alpha = 0.4, position = position_jitterdodge
  geom_linerange(aes(ymin = lowerci, ymax =upperci), size = 1, position = position_dodge(width
  geom_point(size = 3, alpha = 0.8, stroke = 1, position = position_dodge(width = 0.55)) +
  scale_shape_manual(values = c(21, 21, 23, 24, 24, 22, 22)) +
  scale_color_manual(values = c("#b2abd2", "#b2abd2", "#5e3c99", "#fdb863", "#fdb863", "#e66101
  scale_fill_manual(values = c("#b2abd2", "white", "white", "#fdb863", "white", "#e66101", "whi
  labs(y = "", x = "") +
  scale_y_continuous(expand = c(0, 0), limits = c(0, 8)) +
  scale_x_discrete(labels = c("A" = " Reefs pooled")) +
  ggtitle(expression(paste(bold("C)   "), italic("P. astreoides"))))


## Model output and R2
past_glm_out # summary output of the GLM
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: Gamma  ( log )
Formula: dist ~ fpco2 + ftemp + (1 | colony) + (1 | tank)
   Data: por_dist
```

```
     AIC      BIC   logLik deviance df.resid 
   142.4    158.3    -63.2    126.4       46 

Scaled residuals: 
    Min      1Q  Median      3Q     Max 
-1.6543 -0.5670 -0.1035  0.5470  2.2029 

Random effects:
 Groups   Name        Variance Std.Dev.
 tank     (Intercept) 0.001233 0.03512 
 colony   (Intercept) 0.037406 0.19341 
 Residual             0.056156 0.23697 
Number of obs: 54, groups:  tank, 21; colony, 11

Fixed effects:
            Estimate Std. Error t value Pr(>|z|)    
(Intercept)  1.03344    0.12525   8.251  < 2e-16 ***
fpco2420    -0.04516    0.11478  -0.393 0.693972    
fpco2680     0.03277    0.07852   0.417 0.676442    
fpco23290    0.12410    0.08225   1.509 0.131338    
ftemp31      0.26204    0.06856   3.822 0.000132 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
          (Intr) fp2420 fp2680 f23290
fpco2420  -0.153                     
fpco2680  -0.318  0.380              
fpco23290 -0.353  0.306  0.526       
ftemp31   -0.166 -0.379 -0.073  0.068
```

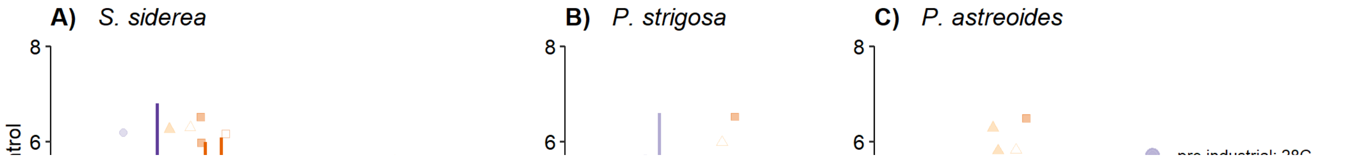
```
r2_nakagawa(past_dist_glm) # conditional and marginal R2
```

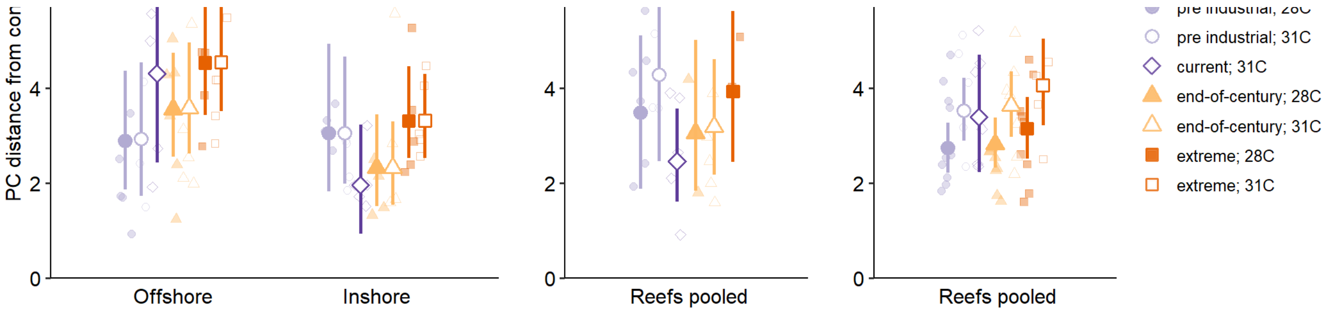
```
# R2 for Mixed Models

  Conditional R2: 0.500
     Marginal R2: 0.147
```

Figure 3

Taken from OA_OW_Physiology_manuscript.Rmd lines 1322 - 1329

```
ggarrange(ssid_plast_plot, pstr_plast_plot, past_plast_plot, common.legend = TRUE, legend = "ri
```





```
ggsave("Figures/Final_Figures/Figure3_PhysPlasticity.pdf", width = 14, height = 4, useDingbat
ggsave("Figures/Final_Figures/Figure3_PhysPlasticity.png", width = 14, height = 4, dpi = 650)
ggsave("Figures/Final_Figures/Figure3_PhysPlasticity.tiff", width = 14, height = 4, dpi = 65(
```

Figure S1

Taken from OA_OW_Physiology_manuscript.Rmd lines 1356 - 1401

Link is forbidden and I am unable to download. Since the raw data is included in the repository, I will proceed with their data.

S1 Actual Plot

Taken from OA_OW_Physiology_manuscript.Rmd lines 1403 - 1488

```
### Load the site lat/lon locations from .csv
sites <- data.frame(Site = c("PHMR", "SCMR"),
                    lon = c(-88.572760, -88.261440),
                    lat = c(16.189870, 16.116670))



## convert sampling points to spatial points and applying projection to points
LongLat <- cbind(sites$lon, sites$lat) # make lat/lon data frame
sitePTS <- SpatialPoints(LongLat) # convert data frame to spatialpoints object
proj <- "+proj=longlat +ellps=WGS84 +datum=WGS84" # this is the projection that will be applied
projection(sitePTS) <- proj # apply specified projection



### Open sst_BZ_cropped.nc file
raster_path <- "Data/Raw_data/sst_BZ_cropped.nc" # specify the path to the Day SST netCDF (no p
sst_raster <- brick(raster_path, varname = "sst") # create raster brick of daily SST data
# plot(sst_raster) # sometimes I plot to quickly have R plot this to make sure I am getting mul

netCDF <- nc_open(raster_path) # next, 'open' the netCDF to pull the time element that is extra
history <- ncatt_get(netCDF,0,"history")[[2]] # pull the 'history' parameter (this has all the

# create a dataframe of the pulled file name and then removed text strings per file (now each I
dates <- data.frame(time = strsplit(history, " ")[[1]][c(-1:-20, -249:-250)]) %>% # Range for r
  mutate(time = gsub("AQUA_MODIS.", "", time), # removes 'AQUA_MODIS.' from the file names
         time = gsub(".L3m.MO.SST.sst.4km.nc", "", time)) # removes '.L3m.DAY.SST4.sst4.4km_sub

# make new column of above data frame that converts the numbers to a date string
dates$date_time <- as.Date(paste(substr(dates$time, 1, 4), substr(dates$time, 5, 6), substr(dat


### Extract SST from raster by site (plus the 5 km buffer) and convert to usable data frame
site_SST <- raster::extract(sst_raster, sitePTS, fun = mean) # this will take the mean SST with


# convert the extracted matrix to final data frame
site_SST_xts <- xts(t(site_SST), dates$date_time) # creates extensible time-series object, t is
site_SST_df <- data.frame(date=index(t(site_SST_xts)), coredata(t(site_SST_xts))) # convert xts
site_SST_df$site <- factor(seq(1, length(site_SST_df$date), 1)) # add ID to identify unique sam
site_SST_df$date <- NULL # remove this column since it is repetitive
site_SST_long <- gather(site_SST_df, date, sst, X2002.12.01:X2021.11.01) # convert from wide to

# update the data frame for for better dates and site IDs
sst_gps <- site_SST_long %>% separate(date, c("year", "month", "day")) # create year, month, an
sst_gps$date <- paste(sst_gps$month, sst_gps$day, sst_gps$year, sep = "/") # creates a combined
sst_gps <- spread(sst_gps, site, sst) # converts from long to wide format

sst_gps <- sst_gps %>%
  rename(
    PHMR = `1`,
    SCMR = `2`,
  )


## convert data from wide to long
sst_long <- gather(sst_gps, site, sst, PHMR:SCMR) %>%
  mutate(date = as.Date(gsub("X2", "2", date), format = "%m/%d/%Y"))

p_mean <- sst_long %>% filter(site == "PHMR") %>%
  summarise(mean = mean(sst, na.rm = TRUE))

s_mean <- sst_long %>% filter(site == "SCMR") %>%
  summarise(mean = mean(sst, na.rm = TRUE))


ggplot(data = sst_long, aes(x = date, y = sst, colour = site)) +
  geom_hline(aes(yintercept = 28), colour = "#4393c3", size = 1, alpha = 0.2, linetype = "dashe
  annotate("text", x = max(sst_long$date), y = 31.2, label = "31C treat", colour = "#b2182b") +
  geom_hline(aes(yintercept = 31), colour = "#b2182b", size = 1, alpha = 0.2, linetype = "dashe
  annotate("text", x = max(sst_long$date), y = 27.8, label = "28C treat", colour = "#4393c3") +
  geom_hline(data = sst_long %>% filter(site == "PHMR"), aes(yintercept = mean(sst, na.rm = TRU
  #annotate("text", x = min(sst_long$date), y = p_mean[[1]] + 0.2, label = paste0("mean: ", rou
  geom_hline(data = sst_long %>% filter(site == "SCMR"), aes(yintercept = mean(sst, na.rm = TRU
  #annotate("text", x = min(sst_long$date), y = s_mean[[1]] - 0.2, label = paste0("mean: ", rou
  geom_line(aes(group = site)) +
  scale_color_manual("", labels = c("PHMR (inshore)", "SCMR (offshore)"), values = c("#e9aa2b",
  labs(x = "", y = "SST (C)") +
  theme_classic() +
  theme(legend.position = c(0.9, 0.13))

ggsave("Figures/Supplemental_Figures/S1_Fig.pdf", width = 10, height = 4, useDingbats=FALSE)
ggsave("Figures/Supplemental_Figures/S1_Fig.png", width = 10, height = 4, dpi = 650)
ggsave("Figures/Supplemental_Figures/S1_Fig.tiff", width = 10, height = 4, dpi = 650)
```

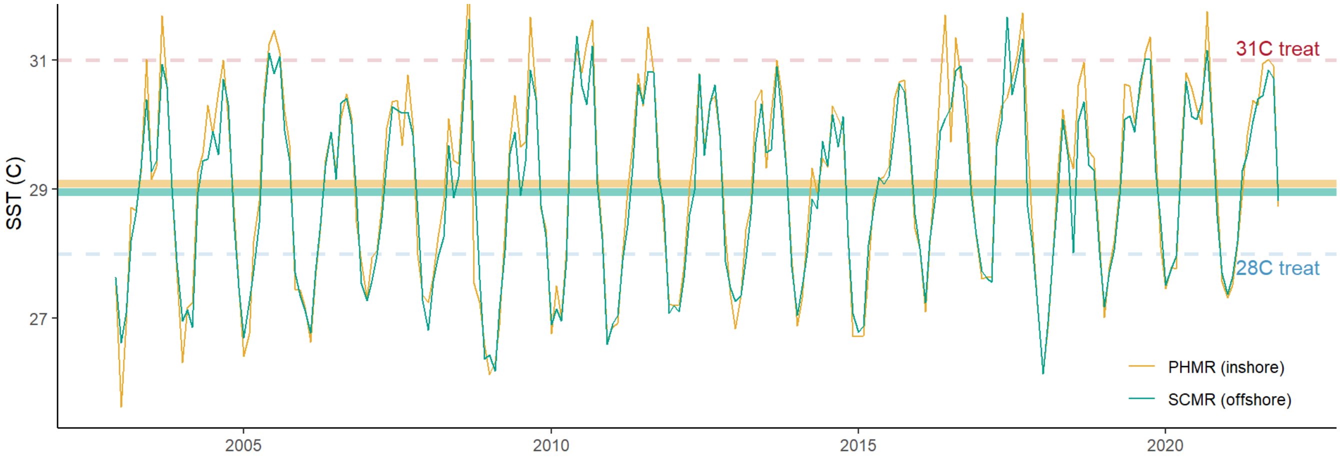
With the data file taken from the repository, the analysis went further, but because of an error in the rename function, the code needed the specification of the dplyr package.

```
### Load the site lat/lon locations from .csv
sites <- data.frame(Site = c("PHMR", "SCMR"),
                    lon = c(-88.572760, -88.261440),
                    lat = c(16.189870, 16.116670))


## convert sampling points to spatial points and applying projection to points
LongLat <- cbind(sites$lon, sites$lat) # make lat/lon data frame
sitePTS <- SpatialPoints(LongLat) # convert data frame to spatialpoints object
proj <- "+proj=longlat +ellps=WGS84 +datum=WGS84" # this is the projection that will be appliec
projection(sitePTS) <- proj # apply specified projection


### Open sst_BZ_cropped.nc file
raster_path <- "Data/Raw_data/sst_BZ_cropped.nc" # specify the path to the Day SST netCDF (no p
sst_raster <- brick(raster_path, varname = "sst") # create raster brick of daily SST data
```

```
[1] "vobjtovarid4: **** WARNING **** I was asked to get a varid for dimension named record BUT
this dimension HAS NO DIMVAR! Code will probably fail at this point"
```

```
# plot(sst_raster) # sometimes I plot to quickly have R plot this to make sure I am getting mul

netCDF <- nc_open(raster_path) # next, 'open' the netCDF to pull the time element that is extra
history <- ncatt_get(netCDF,0,"history")[[2]] # pull the 'history' parameter (this has all the

# create a dataframe of the pulled file name and then removed text strings per file (now each I
dates <- data.frame(time = strsplit(history, " ")[[1]][c(-1:-20, -249:-250)]) %>% # Range for r
  mutate(time = gsub("AQUA_MODIS.", "", time), # removes 'AQUA_MODIS.' from the file names
         time = gsub(".L3m.MO.SST.sst.4km.nc", "", time)) # removes '.L3m.DAY.SST4.sst4.4km_sub

# make new column of above data frame that converts the numbers to a date string
dates$date_time <- as.Date(paste(substr(dates$time, 1, 4), substr(dates$time, 5, 6), substr(dat


### Extract SST from raster by site (plus the 5 km buffer) and convert to usable data frame
```

```
site_SST <- raster::extract(sst_raster, sitePTS, fun = mean) # this will take the mean SST with


# convert the extracted matrix to final data frame
site_SST_xts <- xts(t(site_SST), dates$date_time) # creates extensible time-series object, t is
site_SST_df <- data.frame(date=index(t(site_SST_xts)), coredata(t(site_SST_xts))) # convert xts
site_SST_df$site <- factor(seq(1, length(site_SST_df$date), 1)) # add ID to identify unique sam
site_SST_df$date <- NULL # remove this column since it is repetitive
site_SST_long <- gather(site_SST_df, date, sst, X2002.12.01:X2021.11.01) # convert from wide to

# update the data frame for for better dates and site IDs
sst_gps <- site_SST_long %>% separate(date, c("year", "month", "day")) # create year, month, an
sst_gps$date <- paste(sst_gps$month, sst_gps$day, sst_gps$year, sep = "/") # creates a combined
sst_gps <- spread(sst_gps, site, sst) # converts from long to wide format

sst_gps <- sst_gps %>%
  dplyr::rename(
    PHMR = `1`,
    SCMR = `2`,
  )


## convert data from wide to long
sst_long <- gather(sst_gps, site, sst, PHMR:SCMR) %>%
  mutate(date = as.Date(gsub("X2", "2", date), format = "%m/%d/%Y"))

p_mean <- sst_long %>% filter(site == "PHMR") %>%
  summarise(mean = mean(sst, na.rm = TRUE))

s_mean <- sst_long %>% filter(site == "SCMR") %>%
  summarise(mean = mean(sst, na.rm = TRUE))


ggplot(data = sst_long, aes(x = date, y = sst, colour = site)) +
  geom_hline(aes(yintercept = 28), colour = "#4393c3", size = 1, alpha = 0.2, linetype = "dashe
  annotate("text", x = max(sst_long$date), y = 31.2, label = "31C treat", colour = "#b2182b") +
  geom_hline(aes(yintercept = 31), colour = "#b2182b", size = 1, alpha = 0.2, linetype = "dashe
  annotate("text", x = max(sst_long$date), y = 27.8, label = "28C treat", colour = "#4393c3") +
  geom_hline(data = sst_long %>% filter(site == "PHMR"), aes(yintercept = mean(sst, na.rm = TRU
  #annotate("text", x = min(sst_long$date), y = p_mean[[1]] + 0.2, label = paste0("mean: ", rou
  geom_hline(data = sst_long %>% filter(site == "SCMR"), aes(yintercept = mean(sst, na.rm = TRU
  #annotate("text", x = min(sst_long$date), y = s_mean[[1]] - 0.2, label = paste0("mean: ", rou
  geom_line(aes(group = site)) +
  scale_color_manual("", labels = c("PHMR (inshore)", "SCMR (offshore)"), values = c("#e9aa2b",
  labs(x = "", y = "SST (C)") +
  theme_classic() +
  theme(legend.position = c(0.9, 0.13))
```



```
ggsave("Figures/Supplemental_Figures/S1_Fig.pdf", width = 10, height = 4, useDingbats=FALSE)
ggsave("Figures/Supplemental_Figures/S1_Fig.png", width = 10, height = 4, dpi = 650)
ggsave("Figures/Supplemental_Figures/S1_Fig.tiff", width = 10, height = 4, dpi = 650)
```

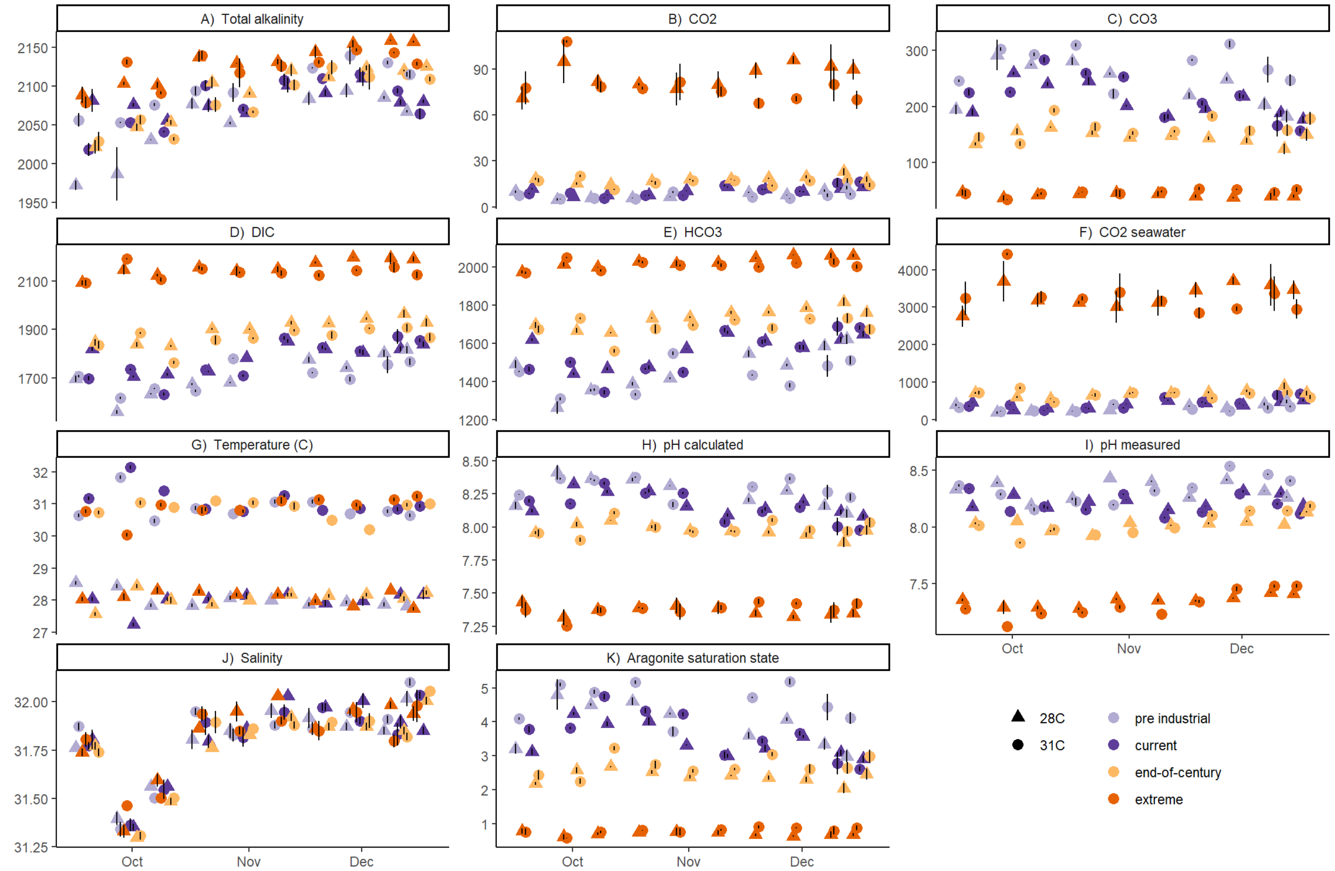
Figure S3

Taken from OA_OW_Physiology_manuscript.Rmd lines 1509 - 1541

```
## Read in the water chem values from the experiment
chem <- read.csv("Data/Raw_data/Exp_waterchem.csv") %>%
  mutate(date = as.Date(date, format = "%m/%d/%Y"), # convert date to date format)
         pco2 = factor(pco2, labels = c("pre industrial", "pre industrial", "current", "current
         temp = factor(temp, labels = c("28C", "31C"))) %>%
  dplyr::group_by(date, pco2, temp, param, treat) %>%
  dplyr::summarise(mean = mean(value),
            sd = sd(value),
            n = n())
```

```
`summarise()` has grouped output by 'date', 'pco2', 'temp', 'param'. You can
override using the `.groups` argument.
```

```
dodge <- position_dodge(width = 7)
labels <- c(alk = "A)  Total alkalinity", co2 = "B)  CO2", co3 = "C)  CO3", dic = "D)  DIC", hc



ggplot(data = chem, aes(x = date, y = mean, colour = pco2, shape = temp, group = treat)) +
  geom_point(size = 3, position = dodge) +
  geom_errorbar(aes(ymin = mean - sd, ymax = mean + sd, group = treat), width = 0, size = 0.5,
  #scale_y_continuous(trans = 'log2') +
  scale_shape_manual(values = c(17, 16)) +
  scale_color_manual(values = c("#b2abd2", "#5e3c99", "#fdb863", "#e66101")) +
  facet_wrap(~ param, ncol = 3, scales = "free_y", labeller = labeller(param = labels)) +
  theme_classic() +
  theme_classic() +
  labs(x = "", y = "") +
  theme(legend.position = c(0.83, 0.11), legend.box = "horizontal", legend.title = element_blan
```



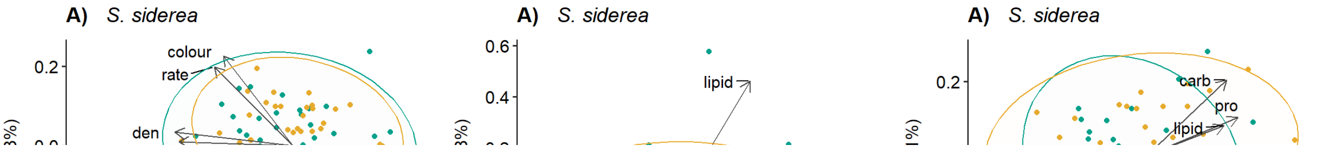
```
ggsave("Figures/Supplemental_Figures/S3_Fig.pdf", width = 12, height = 8, useDingbats=FALSE)
ggsave("Figures/Supplemental_Figures/S3_Fig.png", width = 12, height = 8, dpi = 650)
ggsave("Figures/Supplemental_Figures/S3_Fig.tiff", width = 12, height = 8, dpi = 500)
```

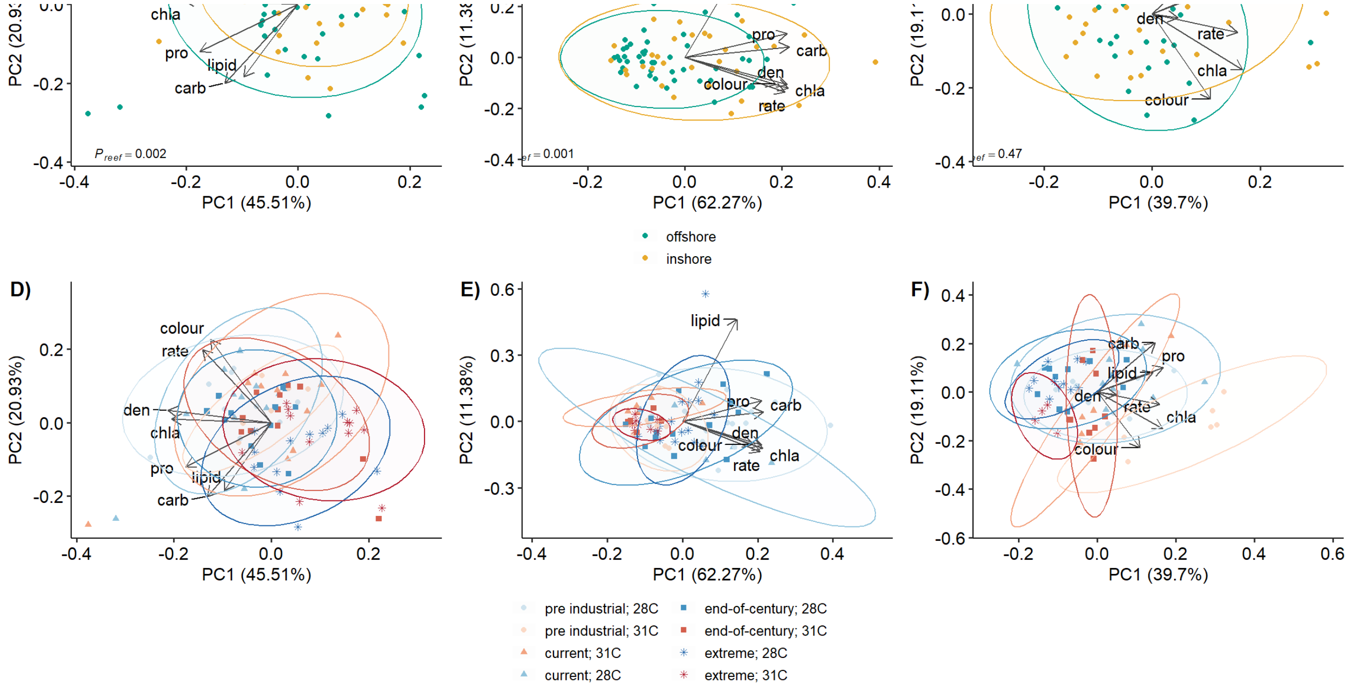
After copying in the data from the experiment, everything ran correctly.

Figure S4

Taken from OA_OW_Physiology_manuscript.Rmd lines 1551 - 1561

```
reef_pca_all <- ggarrange(s_reef_pca, p_reef_pca, a_reef_pca, ncol = 3, common.legend = TRUE, l
trt_pca_all <- ggarrange(s_pt_pca, p_pt_pca, a_pt_pca, ncol = 3, common.legend = TRUE, legend =

ggarrange(reef_pca_all, trt_pca_all, ncol = 1)
```





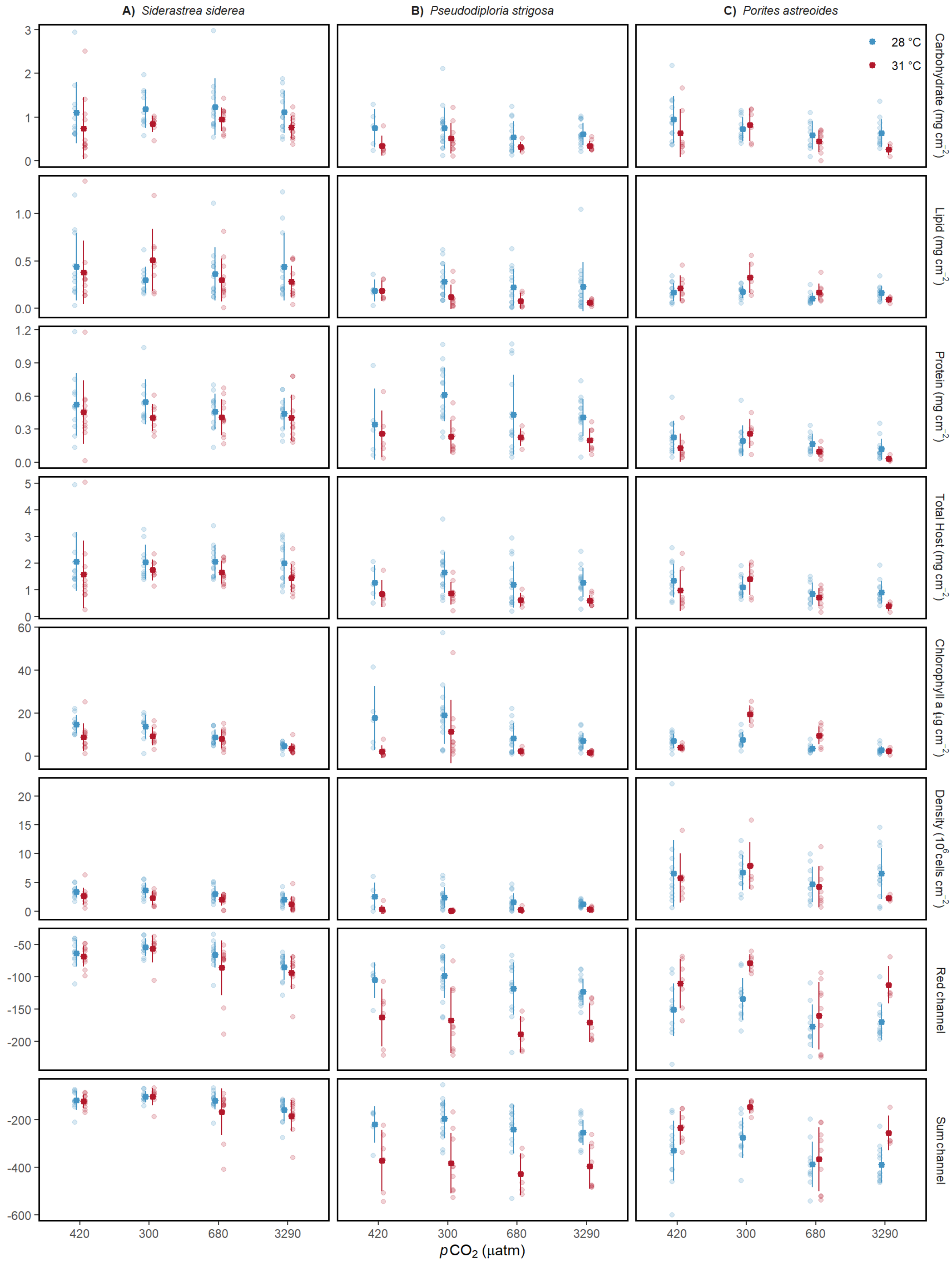
```
ggsave("Figures/Supplemental_Figures/S4_Fig.pdf", width = 13, height = 8, useDingbats=FALSE)
ggsave("Figures/Supplemental_Figures/S4_Fig.png", width = 13, height = 8, dpi = 650)
ggsave("Figures/Supplemental_Figures/S4_Fig.tiff", width = 13, height = 8, dpi = 450)
```

Everything ran as expected.

Figure S5

Taken from OA_OW_Physiology_manuscript.Rmd lines 1571 - 1599

```
## subset for the physiology parameters per species and convert from wide to long format
df_withT_phys <- gather(df_withT, param, value, c(14:17, 34, 38:39, 42))[,-13:-33]
df_noT_phys <- filter(df_withT_phys, species != "T") %>% droplevels() # dropping U. tenuifolia

# reorder the parameters and assign labels for plotting
df_noT_phys <- df_noT_phys %>%
  mutate(param = factor(param, levels = c("carb", "lipid", "pro", "host", "chla", "den", "red",
         species = factor(species, levels = c("S", "P", "A"), labels = c(expression(paste(bold(


## Plot all phys for all species (mean, SE, and raw observations)
ggplot(df_noT_phys, aes(x = fpco2, y = value, color = ftemp)) +
  theme_bw() +
  theme(panel.grid.major=element_blank(), panel.grid.minor=element_blank(), panel.background=el
  geom_point(alpha = 0.2, position = position_dodge(width = 0.3)) +
  stat_summary(fun.data = mean_sdl, fun.args = list(mult = 1), geom = "errorbar", width = 0, po
  stat_summary(fun = "mean", size = 0.3, position = position_dodge(width = 0.2)) +
  scale_color_manual("", labels = c("28 °C", "31 °C"), values = c("#4393c3", "#b2182b"))+
  facet_grid(param ~ species, scales = "free_y", labeller = label_parsed) +
  guides(color = guide_legend(override.aes = list(linetype = c(0, 0)))) +
  labs(y = "", x = alab)
```



```
ggsave("Figures/Supplemental_Figures/S5_Fig.pdf", width = 10, height = 13, useDingbats=FALSE, 
ggsave("Figures/Supplemental_Figures/S5_Fig.png", width = 10, height = 13, dpi = 650)
ggsave("Figures/Supplemental_Figures/S5_Fig.tiff", width = 10, height = 13, dpi = 400)
```

S6

Taken from OA_OW_Physiology_manuscript.Rmd lines 1609 - 1659

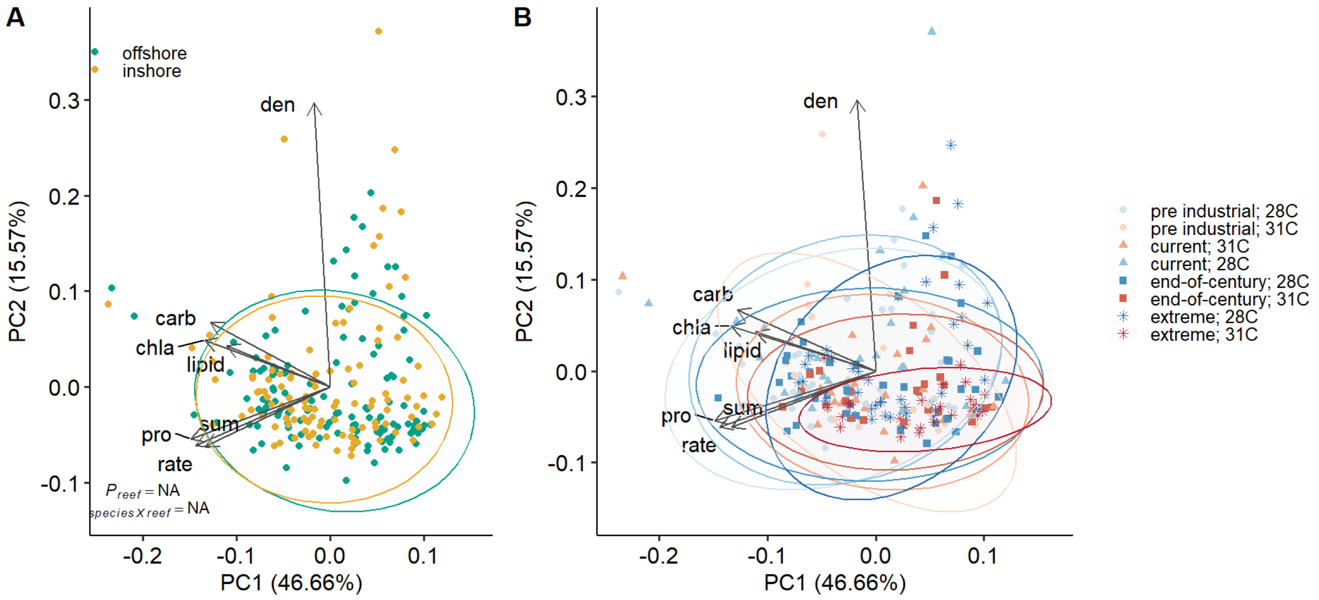
```
all_reef_pca <- autoplot(all_pca, data = all_df,
                       colour = "reef",
                       fill = "reef",
                       frame = TRUE,
                       frame.type = "t", # displaying ellipses with multivariate t-distributio
                       frame.level = 0.95, # using 95% CI for all ellipses
                       frame.alpha = 0.01,
                       loadings = TRUE,
                       loadings.colour = "grey29",
                       loadings.label = TRUE,
                       loadings.label.colour = "black",
                       loadings.label.size = 4,
                       loadings.label.hjust = 1.5,
                       loadings.label.vjust = 0.5,
                       loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b'
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(lin
  theme(legend.position = c(0.11, 0.9), legend.title = element_blank(), legend.background = ele
  annotate("text", x = -0.2, y = -0.108, label = deparse(all_reef_pval), parse = TRUE, size = 
  annotate("text", x = -0.2, y = -0.128, label = deparse(all_s_reef_pval), parse = TRUE, size 


# pco2 * temp
all_df$pco2_f <- factor(all_df$pco2, labels = c("pre industrial; 28C", "pre industrial; 31C", 
all_pt_pca <- autoplot(all_pca, data = all_df,
         colour = "pco2_f",
         fill = "pco2_f",
         shape = "pco2_f",
         frame = TRUE,
         frame.type = "t", # displaying ellipses with multivariate t-distributions for small n
         frame.level = 0.95, # using 95% CI for all ellipses
         frame.alpha = 0.01,
         frame.size = 5,
         loadings = TRUE,
         loadings.colour = "grey29",
         loadings.label = TRUE,
         loadings.label.colour = "black",
         loadings.label.size = 4,
         loadings.label.hjust = 1.5,
         loadings.label.vjust = 0.5,
         loadings.label.repel = TRUE) +
  scale_fill_manual("", values = c("#d1e5f0", "#fddbc7", "#f4a582", "#92c5de", "#4393c3", "#d6
  scale_colour_manual("", values = c("#d1e5f0", "#fddbc7", "#f4a582", "#92c5de", "#4393c3", "#
  scale_shape_manual("", values = c(19, 19, 17, 17, 15, 15, 8, 8)) +
  guides(fill = FALSE, color = guide_legend(keyheight = 0.5, ncol = 1, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
```

Everything runs as expected.

Save figure

Taken from OA_OW_Physiology_manuscript.Rmd lines 1661 - 1668

```
ggarrange(all_reef_pca, all_pt_pca, ncol = 2, labels = "AUTO", widths = c(0.6, 1))
```



```
  ggsave("Figures/Supplemental_Figures/S6_Fig.pdf", width = 10, height = 4.5, useDingbats=FALSE
  ggsave("Figures/Supplemental_Figures/S6_Fig.tiff", width = 10, height = 4.5, dpi = 650)
  ggsave("Figures/Supplemental_Figures/S6_Fig.png", width = 10, height = 4.5, dpi = 650)
```

Everything runs as expected.

Figure S7

Taken from OA_OW_Physiology_manuscript.Rmd lines 1678 - 1700

```
## subset for host or symbiont physiology
```

```
sid_pca_df_host <- s_df[,c(15,22:23)] # host only phys (lipid, protein, carb)
sid_pca_df_symb <- s_df[,c(14,16,17)] # symbiont only phys (chla, density, color intensity)


## run the adonis for HOST phys
#s_pca_host_mod <- adonis2(sid_pca_df_host ~ reef * ftemp * fpco2, data = s_df, method = 'eu',
s_pca_host_mod <- adonis2(sid_pca_df_host ~ fpco2 + ftemp + reef, data = s_df, method = 'eu', p
#s_pca_host_mod # view SSID adonis output

## run the adonis for SYMB phys
#s_pca_symb_mod <- adonis2(sid_pca_df_symb ~ reef * ftemp * fpco2, data = s_df, method = 'eu',
s_pca_symb_mod <- adonis2(sid_pca_df_symb ~ fpco2 + ftemp + reef, data = s_df, method = 'eu', p
#s_pca_symb_mod # view SSID adonis output


## perform principal component analysis (PCA)
s_pca_host <- prcomp(sid_pca_df_host, center = TRUE, scale= TRUE)
s_pca_symb <- prcomp(sid_pca_df_symb, center = TRUE, scale= TRUE)
```

Everything runs as expected.However, in the final plot, the values for the S. sidera symbionts appear to be reflected across the x-axis. So, the x scores for sid_pca_df_symb will be negated.

```
## subset for host or symbiont physiology
sid_pca_df_host <- s_df[,c(15,22:23)] # host only phys (lipid, protein, carb)
sid_pca_df_symb <- s_df[,c(14,16,17)] # symbiont only phys (chla, density, color intensity)

## run the adonis for HOST phys
#s_pca_host_mod <- adonis2(sid_pca_df_host ~ reef * ftemp * fpco2, data = s_df, method = 'eu',
s_pca_host_mod <- adonis2(sid_pca_df_host ~ fpco2 + ftemp + reef, data = s_df, method = 'eu', p
#s_pca_host_mod # view SSID adonis output

## run the adonis for SYMB phys
#s_pca_symb_mod <- adonis2(sid_pca_df_symb ~ reef * ftemp * fpco2, data = s_df, method = 'eu',
s_pca_symb_mod <- adonis2(sid_pca_df_symb ~ fpco2 + ftemp + reef, data = s_df, method = 'eu', p
#s_pca_symb_mod # view SSID adonis output


## perform principal component analysis (PCA)
s_pca_host <- prcomp(sid_pca_df_host, center = TRUE, scale= TRUE)
s_pca_symb <- prcomp(sid_pca_df_symb, center = TRUE, scale= TRUE)

# Reflect over the x axis
s_pca_symb$x[, 2] <- -s_pca_symb$x[, 2]
s_pca_symb$rotation[, 2] <- -s_pca_symb$rotation[, 2]
```

S7 Plot

Taken from OA_OW_Physiology_manuscript.Rmd lines 1702 - 1854

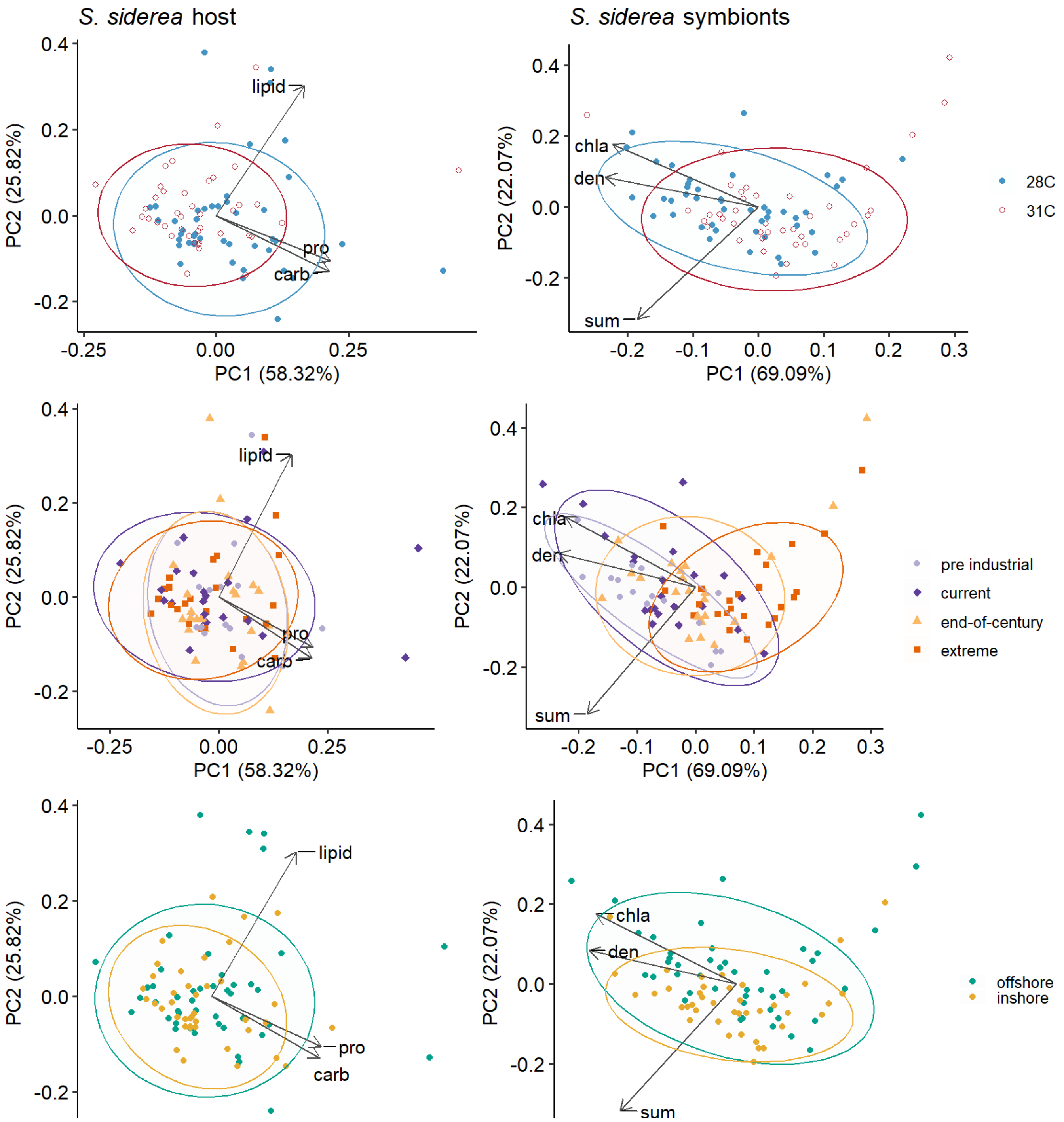
```
## Temperature PCAs
# HOST temperature
sid_host_temp_pca <- autoplot(s_pca_host, data = s_df,
         colour = "ftemp",
         #shape = "fpco2",
         shape = "ftemp",
         fill = "ftemp",
         frame = TRUE,
         frame.type = "t", # displaying ellipses with multivariate t-distributions for small n
         frame.level = 0.95, # using 95% CI for all ellipses
         frame.alpha = 0.01,
         loadings = TRUE,
         loadings.colour = "grey29",
         loadings.label = TRUE,
         loadings.label.colour = "black",
         loadings.label.size = 4,
         loadings.label.hjust = 1.5,
         loadings.label.vjust = 0.5,
         loadings.label.repel = TRUE) +
  scale_shape_manual("temperature", labels = c("28C", "31C"), values = c(21, 21)) +
  scale_color_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "#b2182b")
  scale_fill_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "transparent
  guides(color = guide_legend(keyheight = 0.3, nrow = 4, byrow = TRUE, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  ggtitle(expression(paste(italic("S. siderea")," host")))


# SYMB temperature
sid_symb_temp_pca <- autoplot(s_pca_symb, data = s_df,
         colour = "ftemp",
         #shape = "fpco2",
         shape = "ftemp",
         fill = "ftemp",
         frame = TRUE,
         frame.type = "t", # displaying ellipses with multivariate t-distributions for small n
         frame.level = 0.95, # using 95% CI for all ellipses
         frame.alpha = 0.01,
         loadings = TRUE,
         loadings.colour = "grey29",
         loadings.label = TRUE,
         loadings.label.colour = "black",
         loadings.label.size = 4,
         loadings.label.hjust = 1.5,
         loadings.label.vjust = 0.5,
         loadings.label.repel = TRUE) +
  scale_shape_manual("temperature", labels = c("28C", "31C"), values = c(21, 21)) +
  scale_color_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "#b2182b")
  scale_fill_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "transparent
  guides(color = guide_legend(keyheight = 0.3, nrow = 4, byrow = TRUE, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  ggtitle(expression(paste(italic("S. siderea")," symbionts")))

## pCO2 PCAs
# HOST pco2
sid_host_pco2_pca <- autoplot(s_pca_host, data = s_df,
                          colour = "fpco2",
                          #shape = "ftemp",
                          fill = "fpco2",
                          shape = "fpco2",
                          frame = TRUE,
                          frame.type = "t", # displaying ellipses with multivariate t-distribut
                          frame.level = 0.95, # using 95% CI for all ellipses
                          frame.alpha = 0.01,
                          loadings = TRUE,
                          loadings.colour = "grey29",
                          loadings.label = TRUE,
                          loadings.label.colour = "black",
                          loadings.label.size = 4,
                          loadings.label.hjust = 1.5,
                          loadings.label.vjust = 0.5,
                          loadings.label.repel = TRUE) +
  scale_shape_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  scale_fill_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"), v
  scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  guides(color = guide_legend(nrow = 4, override.aes = list(linetype = c(0, 0, 0, 0)))) +
  theme(legend.background = element_rect(fill = "transparent", color = NA))


# SYMB pco2
sid_symb_pco2_pca <- autoplot(s_pca_symb, data = s_df,
                          colour = "fpco2",
                          #shape = "ftemp",
                          fill = "fpco2",
                          shape = "fpco2",
                          frame = TRUE,
                          frame.type = "t", # displaying ellipses with multivariate t-distribut
                          frame.level = 0.95, # using 95% CI for all ellipses
                          frame.alpha = 0.01,
                          loadings = TRUE,
                          loadings.colour = "grey29",
                          loadings.label = TRUE,
                          loadings.label.colour = "black",
                          loadings.label.size = 4,
                          loadings.label.hjust = 1.5,
                          loadings.label.vjust = 0.5,
                          loadings.label.repel = TRUE) +
  scale_shape_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  scale_fill_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"), v
  scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  guides(color = guide_legend(nrow = 4, override.aes = list(linetype = c(0, 0, 0, 0)))) +
  guides(color = guide_legend(nrow = 4, override.aes = list(linetype = c(0, 0, 0, 0)))) +
  theme(legend.background = element_rect(fill = "transparent", color = NA))



## Reef PCAs
# HOST reef
sid_host_reef_pca <- autoplot(s_pca_host, data = s_df,
                          colour = "reef",
                          fill = "reef",
                          frame = TRUE,
                          frame.type = "t", # displaying ellipses with multivariate t-distribut
                          frame.level = 0.95, # using 95% CI for all ellipses
                          frame.alpha = 0.01,
                          loadings = TRUE,
                          loadings.colour = "grey29",
                          loadings.label = TRUE,
                          loadings.label.colour = "black",
                          loadings.label.size = 4,
                          loadings.label.hjust = -0.6,
                          loadings.label.vjust = 0.5,
                          loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b'
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",


# SYMB reef
sid_symb_reef_pca <- autoplot(s_pca_symb, data = s_df,
                          colour = "reef",
                          fill = "reef",
                          frame = TRUE,
                          frame.type = "t", # displaying ellipses with multivariate t-distribut
                          frame.level = 0.95, # using 95% CI for all ellipses
                          frame.alpha = 0.01,
                          loadings = TRUE,
                          loadings.colour = "grey29",
                          loadings.label = TRUE,
                          loadings.label.colour = "black",
                          loadings.label.size = 4,
                          loadings.label.hjust = -0.6,
                          loadings.label.vjust = 0.5,
                          loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b'
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
```

Everything runs as expected.

Arranging and plotting

Taken from OA_OW_Physiology_manuscript.Rmd lines 1856 - 1870

```
## Combine PCAs by treatment for common legend:
ssid_host_symb_temp <- ggarrange(sid_host_temp_pca, sid_symb_temp_pca,ncol = 2, common.legend =
ssid_host_symb_pco2 <- ggarrange(sid_host_pco2_pca, sid_symb_pco2_pca,ncol = 2, common.legend =
ssid_host_symb_reef <- ggarrange(sid_host_reef_pca, sid_symb_reef_pca,ncol = 2, common.legend =

## Add all together for final figure:
ggarrange(ssid_host_symb_temp, ssid_host_symb_pco2, ssid_host_symb_reef, nrow = 3)
```



```
ggsave("Figures/Supplemental_Figures/S7_Fig.pdf", width = 9, height = 10, useDingbats=FALSE)
ggsave("Figures/Supplemental_Figures/S7_Fig.png", width = 9, height = 10, dpi = 650)
ggsave("Figures/Supplemental_Figures/S7_Fig.tiff", width = 9, height = 10, dpi = 450)
```

Everything runs as expected

Figure S8

Taken from OA_OW_Physiology_manuscript.Rmd lines 1880 - 1902

```
## subset for host or symbiont physiology
dip_pca_df_host <- p_df[,c(15,22:23)] # host only phys (lipid, protein, carb)
dip_pca_df_symb <- p_df[,c(14,16,17)] # symbiont only phys (chla, density, color intensity)



## run the adonis for HOST phys
#p_pca_host_mod <- adonis2(dip_pca_df_host ~ reef * ftemp * fpco2, data = p_df, method = 'eu',
p_pca_host_mod <- adonis2(dip_pca_df_host ~ fpco2 + ftemp + reef, data = p_df, method = 'eu',
#p_pca_host_mod # view PSTR adonis output

## run the adonis for SYMB phys
#p_pca_symb_mod <- adonis2(dip_pca_df_symb ~ reef * ftemp * fpco2, data = p_df, method = 'eu',
p_pca_symb_mod <- adonis2(dip_pca_df_symb ~ fpco2 + ftemp + reef, data = p_df, method = 'eu',
#p_pca_symb_mod # view PSTR adonis output



## perform principal component analysis (PCA)
p_pca_host <- prcomp(dip_pca_df_host, center = TRUE, scale= TRUE)
p_pca_symb <- prcomp(dip_pca_df_symb, center = TRUE, scale= TRUE)
```

Everything runs as expected. The final figure, however, has reversed the proper y axis for symbionts. I will correct it as I did in S7

```
## subset for host or symbiont physiology
dip_pca_df_host <- p_df[,c(15,22:23)] # host only phys (lipid, protein, carb)
dip_pca_df_symb <- p_df[,c(14,16,17)] # symbiont only phys (chla, density, color intensity)



## run the adonis for HOST phys
#p_pca_host_mod <- adonis2(dip_pca_df_host ~ reef * ftemp * fpco2, data = p_df, method = 'eu',
p_pca_host_mod <- adonis2(dip_pca_df_host ~ fpco2 + ftemp + reef, data = p_df, method = 'eu',
#p_pca_host_mod # view PSTR adonis output
```

```
## run the adonis for SYMB phys
#p_pca_symb_mod <- adonis2(dip_pca_df_symb ~ reef * ftemp * fpco2, data = p_df, method = 'eu',
p_pca_symb_mod <- adonis2(dip_pca_df_symb ~ fpco2 + ftemp + reef, data = p_df, method = 'eu', p
#p_pca_symb_mod # view PSTR adonis output



## perform principal component analysis (PCA)
p_pca_host <- prcomp(dip_pca_df_host, center = TRUE, scale= TRUE)
p_pca_symb <- prcomp(dip_pca_df_symb, center = TRUE, scale= TRUE)

# Reflect over the x axis
p_pca_symb$x[, 2] <- -p_pca_symb$x[, 2]
p_pca_symb$rotation[, 2] <- -p_pca_symb$rotation[, 2]
```

Plot PSTR

Taken from OA_OW_Physiology_manuscript.Rmd lines 1904 - 2056

```
## Temperature PCAs
# HOST temperature
dip_host_temp_pca <- autoplot(p_pca_host, data = p_df,
                              colour = "ftemp",
                              #shape = "fpco2",
                              shape = "ftemp",
                              fill = "ftemp",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = 1.5,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_shape_manual("temperature", labels = c("28C", "31C"), values = c(21, 21)) +
  scale_color_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "#b2182b")
  scale_fill_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "transparent
  guides(color = guide_legend(keyheight = 0.3, nrow = 4, byrow = TRUE, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  ggtitle(expression(paste(italic("P. strigosa")," host")))



# SYMB temperature
dip_symb_temp_pca <- autoplot(p_pca_symb, data = p_df,
                              colour = "ftemp",
                              #shape = "fpco2",
                              shape = "ftemp",
                              shape = "ftemp",
                              fill = "ftemp",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = 1.5,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_shape_manual("temperature", labels = c("28C", "31C"), values = c(21, 21)) +
  scale_color_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "#b2182b")
  scale_fill_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "transparent
  guides(color = guide_legend(keyheight = 0.3, nrow = 4, byrow = TRUE, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  ggtitle(expression(paste(italic("P. strigosa")," symbionts")))



## pCO2 PCAs
# HOST pco2
dip_host_pco2_pca <- autoplot(p_pca_host, data = p_df,
                              colour = "fpco2",
                              #shape = "ftemp",
                              fill = "fpco2",
                              shape = "fpco2",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = 1.5,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_shape_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  scale_fill_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"), v
  scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  guides(color = guide_legend(nrow = 4, override.aes = list(linetype = c(0, 0, 0, 0)))) +
  theme(legend.background = element_rect(fill = "transparent", color = NA))


# SYMB pco2
```

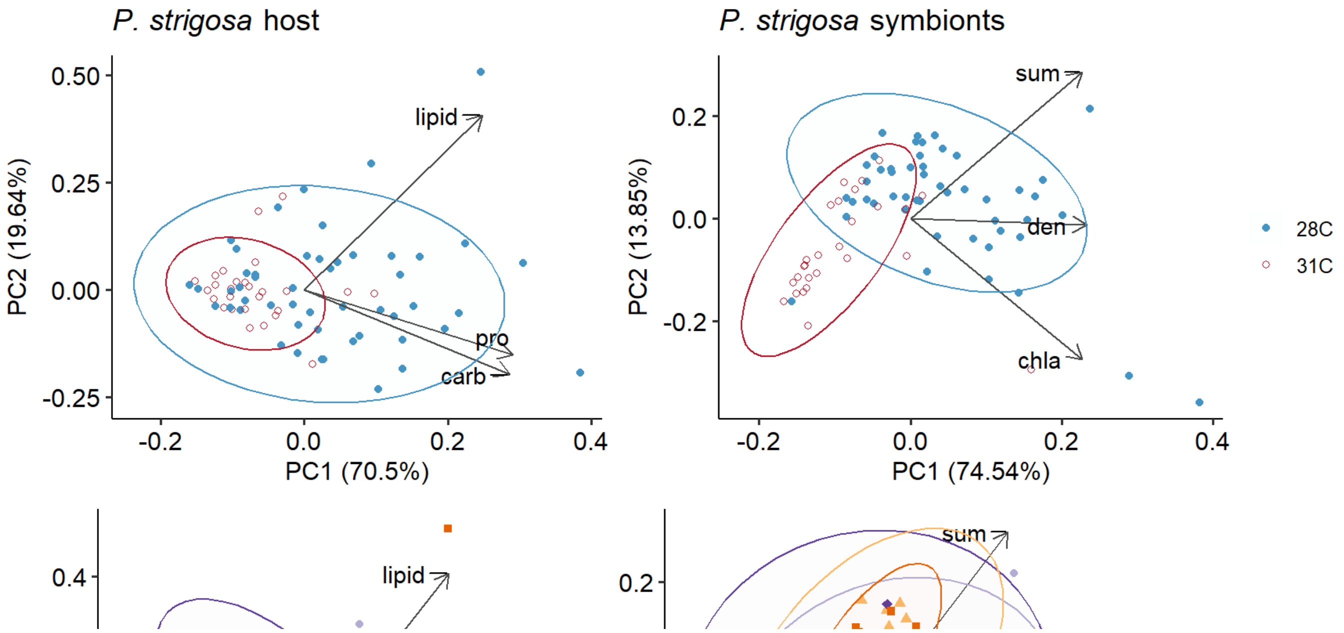
```
dip_symb_pco2_pca <- autoplot(p_pca_symb, data = p_df,
                              colour = "fpco2",
                              #shape = "ftemp",
                              fill = "fpco2",
                              shape = "fpco2",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = 1.5,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_shape_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  scale_fill_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"), v
  scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  guides(color = guide_legend(nrow = 4, override.aes = list(linetype = c(0, 0, 0, 0)))) +
  theme(legend.background = element_rect(fill = "transparent", color = NA))



## Reef PCAs
# HOST reef
dip_host_reef_pca <- autoplot(p_pca_host, data = p_df,
                              colour = "reef",
                              fill = "reef",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = -0.6,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b"
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",


# SYMB reef
dip_symb_reef_pca <- autoplot(p_pca_symb, data = p_df,
                              colour = "reef",
                              fill = "reef",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = -0.6,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b"
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
```

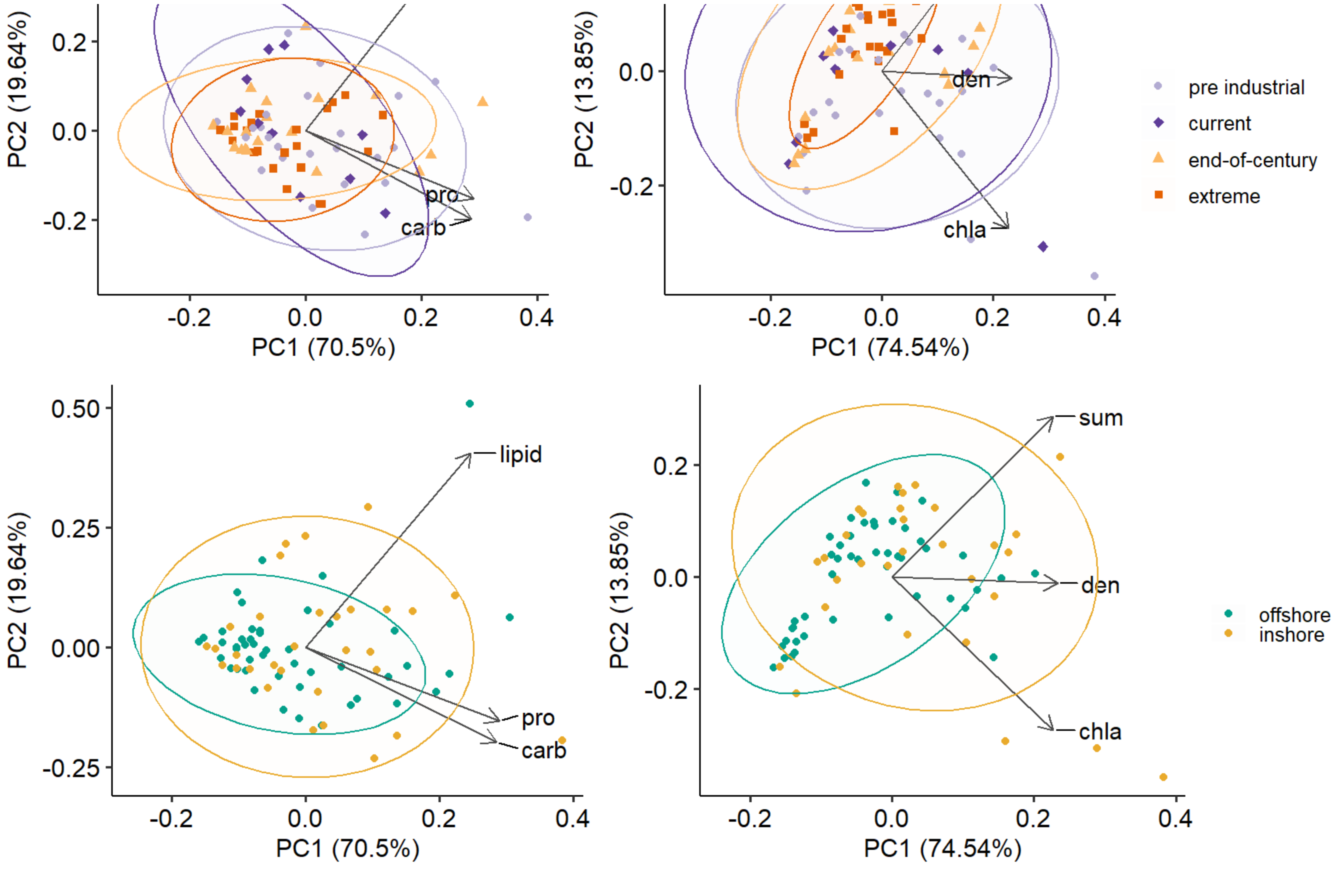
Everything runs as expected

Arranging and Plotting

Taken from OA_OW_Physiology_manuscript.Rmd lines 2058 - 2072

```
## Combine PCAs by treatment for common legend:
PSTR_host_symb_temp <- ggarrange(dip_host_temp_pca, dip_symb_temp_pca,ncol = 2, common.legend =
PSTR_host_symb_pco2 <- ggarrange(dip_host_pco2_pca, dip_symb_pco2_pca,ncol = 2, common.legend =
PSTR_host_symb_reef <- ggarrange(dip_host_reef_pca, dip_symb_reef_pca,ncol = 2, common.legend =

## Add all together for final figure:
ggarrange(PSTR_host_symb_temp, PSTR_host_symb_pco2, PSTR_host_symb_reef, nrow = 3)
```





```
ggsave("Figures/Supplemental_Figures/S8_Fig.pdf", width = 9, height = 10, useDingbats=FALSE)
ggsave("Figures/Supplemental_Figures/S8_Fig.png", width = 9, height = 10, dpi = 650)
ggsave("Figures/Supplemental_Figures/S8_Fig.tiff", width = 9, height = 10, dpi = 450)
```

Everything runs as expected.

Figure S9

Taken from OA_OW_Physiology_manuscript.Rmd lines 2082 - 2104

```
## subset for host or symbiont physiology
por_pca_df_host <- a_df[,c(15,22:23)] # host only phys (lipid, protein, carb)
por_pca_df_symb <- a_df[,c(14,16,18)] # symbiont only phys (chla, density, color intensity)


## run the adonis for HOST phys
#a_pca_host_mod <- adonis2(por_pca_df_host ~ reef * ftemp * fpco2, data = a_df, method = 'eu',
a_pca_host_mod <- adonis2(por_pca_df_host ~ fpco2 + ftemp + reef, data = a_df, method = 'eu',
#a_pca_host_mod # view PAST adonis output

## run the adonis for SYMB phys
#a_pca_symb_mod <- adonis2(por_pca_df_symb ~ reef * ftemp * fpco2, data = a_df, method = 'eu',
a_pca_symb_mod <- adonis2(por_pca_df_symb ~ fpco2 + ftemp + reef, data = a_df, method = 'eu',
#a_pca_symb_mod # view PAST adonis output


## perform principal component analysis (PCA)
a_pca_host <- prcomp(por_pca_df_host, center = TRUE, scale= TRUE)
a_pca_symb <- prcomp(por_pca_df_symb, center = TRUE, scale= TRUE)
```

Everything runs as expected

Plot PAST

Taken from OA_OW_Physiology_manuscript.Rmd lines 2106 - 2258

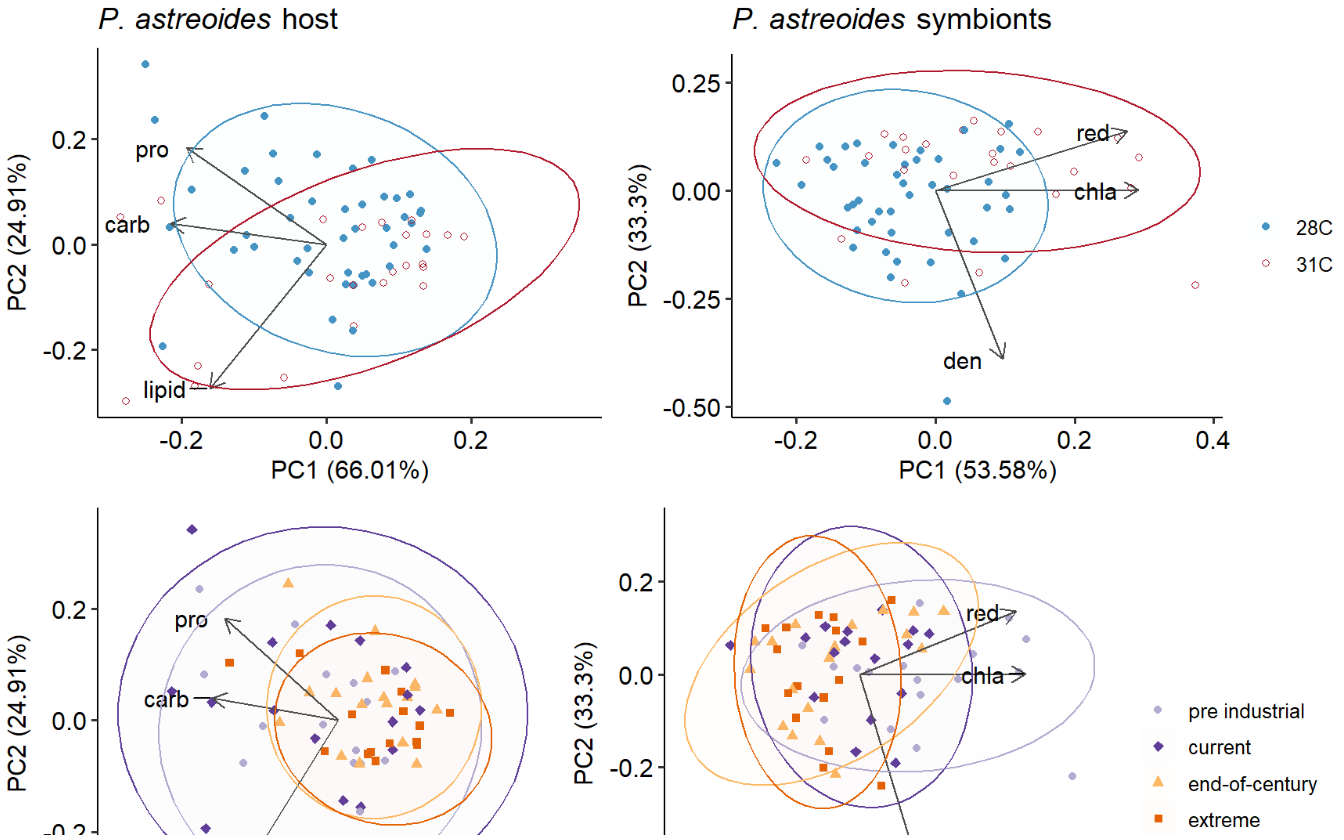
```
## Temperature PCAs
# HOST temperature
por_host_temp_pca <- autoplot(a_pca_host, data = a_df,
                              colour = "ftemp",
                              #shape = "fpco2",
                              shape = "ftemp",
                              fill = "ftemp",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = 1.5,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_shape_manual("temperature", labels = c("28C", "31C"), values = c(21, 21)) +
  scale_color_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "#b2182b"))
  scale_fill_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "transparent
  guides(color = guide_legend(keyheight = 0.3, nrow = 4, byrow = TRUE, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  ggtitle(expression(paste(italic("P. astreoides")," host")))


# SYMB temperature
por_symb_temp_pca <- autoplot(a_pca_symb, data = a_df,
                              colour = "ftemp",
                              #shape = "fpco2",
                              shape = "ftemp",
                              fill = "ftemp",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = 1.5,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_shape_manual("temperature", labels = c("28C", "31C"), values = c(21, 21)) +
  scale_color_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "#b2182b")
  scale_fill_manual("temperature", labels = c("28C", "31C"), values = c("#4393c3", "transparent
  guides(color = guide_legend(keyheight = 0.3, nrow = 4, byrow = TRUE, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
  ggtitle(expression(paste(italic("P. astreoides")," symbionts")))



## pCO2 PCAs
# HOST pco2
por_host_pco2_pca <- autoplot(a_pca_host, data = a_df,
                              colour = "fpco2",
                              #shape = "ftemp",
                              fill = "fpco2",
                              shape = "fpco2",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = 1.5,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_shape_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  scale_fill_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"), v
  scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  guides(color = guide_legend(nrow = 4, override.aes = list(linetype = c(0, 0, 0, 0)))) +
  theme(legend.background = element_rect(fill = "transparent", color = NA))


# SYMB pco2
por_symb_pco2_pca <- autoplot(a_pca_symb, data = a_df,
                              colour = "fpco2",
                              #shape = "ftemp",
                              fill = "fpco2",
                              shape = "fpco2",
                              frame = TRUE,
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = 1.5,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_shape_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  scale_fill_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"), v
  scale_color_manual("", labels = c("pre industrial", "current", "end-of-century", "extreme"),
  guides(color = guide_legend(nrow = 4, override.aes = list(linetype = c(0, 0, 0, 0)))) +
  theme(legend.background = element_rect(fill = "transparent", color = NA))



## Reef PCAs
# HOST reef
por_host_reef_pca <- autoplot(a_pca_host, data = a_df,
                              colour = "reef",
                              fill = "reef",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = -0.6,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b'
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",


# SYMB reef
por_symb_reef_pca <- autoplot(a_pca_symb, data = a_df,
                              colour = "reef",
                              fill = "reef",
                              frame = TRUE,
                              frame.type = "t", # displaying ellipses with multivariate t-distr
                              frame.level = 0.95, # using 95% CI for all ellipses
                              frame.alpha = 0.01
                              frame.alpha = 0.01,
                              loadings = TRUE,
                              loadings.colour = "grey29",
                              loadings.label = TRUE,
                              loadings.label.colour = "black",
                              loadings.label.size = 4,
                              loadings.label.hjust = -0.6,
                              loadings.label.vjust = 0.5,
                              loadings.label.repel = TRUE) +
  scale_color_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b
  scale_fill_manual("reef environment", labels = c("offshore", "inshore"), values = c("#00a08b'
  guides(fill = FALSE, color = guide_legend(keyheight = 0.3, nrow = 2, override.aes = list(line
  theme(legend.title = element_blank(), legend.background = element_rect(fill = "transparent",
```

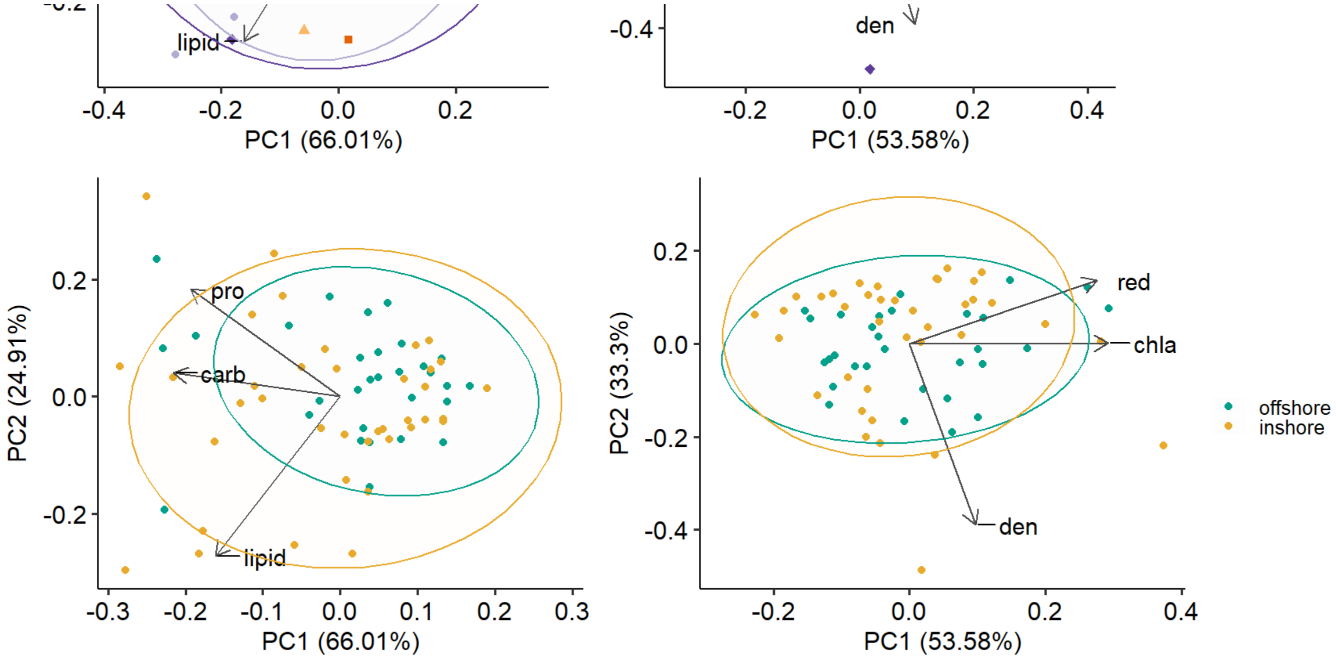
Everything runs as expected

Arranging and Plotting

Taken from OA_OW_Physiology_manuscript.Rmd lines 2260 - 2273

```
## Combine PCAs by treatment for common legend:
PAST_host_symb_temp <- ggarrange(por_host_temp_pca, por_symb_temp_pca,ncol = 2, common.legend =
PAST_host_symb_pco2 <- ggarrange(por_host_pco2_pca, por_symb_pco2_pca,ncol = 2, common.legend =
PAST_host_symb_reef <- ggarrange(por_host_reef_pca, por_symb_reef_pca,ncol = 2, common.legend =

## Add all together for final figure:
ggarrange(PAST_host_symb_temp, PAST_host_symb_pco2, PAST_host_symb_reef, nrow = 3)
```





```
ggsave("Figures/Supplemental_Figures/S9_Fig.pdf", width = 9, height = 10, useDingbats=FALSE)
ggsave("Figures/Supplemental_Figures/S9_Fig.png", width = 9, height = 10, dpi = 650)
ggsave("Figures/Supplemental_Figures/S9_Fig.tiff", width = 9, height = 10, dpi = 500)
```

Everything runs as expected.

Removed figure with host vs symb

Taken from OA_OW_Physiology_manuscript.Rmd lines 2295 - 2374

```
### Notes for how to use PCAplast function are in earlier uses as well as in the 'CustomFunctio

#### SSID
# host
ssid_host_dist <- PCAplast(pca = s_pca_host,
                           data = s_df[,c(1,7,8, 10,11,12,27)],
                           sample_ID = "coral",
                           num_pca =  "all",
                           control_col = "treat2",
                           control_lvl = "420_28",
                           group = "colony")

# symbiont
ssid_symb_dist <- PCAplast(pca = s_pca_symb,
                           data = s_df[,c(1,7,8, 10,11,12,27)],
                           sample_ID = "coral",
                           num_pca =  "all",
                           control_col = "treat2",
                           control_lvl = "420_28",
                           control_lvl = "420_28",
                           group = "colony")


#### PSTR
# host
pstr_host_dist <- PCAplast(pca = p_pca_host,
                           data = p_df[,c(1,7,8, 10,11,12,27)],
                           sample_ID = "coral",
                           num_pca =  "all",
                           control_col = "treat2",
                           control_lvl = "420_28",
                           group = "colony")

# symbiont
pstr_symb_dist <- PCAplast(pca = p_pca_symb,
                           data = p_df[,c(1,7,8, 10,11,12,27)],
                           sample_ID = "coral",
                           num_pca =  "all",
                           control_col = "treat2",
                           control_lvl = "420_28",
                           group = "colony")


#### PAST
# host
past_host_dist <- PCAplast(pca = a_pca_host,
                           data = a_df[,c(1,7,8, 10,11,12,27)],
                           sample_ID = "coral",
                           num_pca =  "all",
                           control_col = "treat2",
                           control_lvl = "420_28",
                           group = "colony")

# symbiont
past_symb_dist <- PCAplast(pca = a_pca_symb,
                           data = a_df[,c(1,7,8, 10,11,12,27)],
                           sample_ID = "coral",
                           num_pca =  "all",
                           control_col = "treat2",
                           control_lvl = "420_28",
                           group = "colony")

#### Combined data by host or symbiont
## Host:
host_dist <- rbind(ssid_host_dist, pstr_host_dist, past_host_dist)

## Symbionts:
symb_dist <- rbind(ssid_symb_dist, pstr_symb_dist, past_symb_dist)
```

```
# add a 'part' label for combining datasets
host_dist$part <- rep("host", nrow(host_dist))
symb_dist$part <- rep("symb", nrow(symb_dist))

# combine datasets
combined_dist <- rbind(host_dist, symb_dist)
```

Everything runs as expected

Plasticity analysis

Taken from OA_OW_Physiology_manuscript.Rmd lines 2376 - 2395

```
## Model selection (via AIC)
comb_dist_mod  <- glmer(dist ~ species * part * reef * fpco2 * ftemp + (1 | colony), family = G
```

```
fixed-effect model matrix is rank deficient so dropping 18 columns / coefficients
```

```
comb_dist_mod2 <- glmer(dist ~ species * part * reef * fpco2 + ftemp + (1 | colony), family = G
```

```
fixed-effect model matrix is rank deficient so dropping 2 columns / coefficients
```

```
comb_dist_mod3 <- glmer(dist ~ species * part * reef + fpco2 * ftemp + (1 | colony), family = G
```

```
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
```

```
comb_dist_mod4 <- glmer(dist ~ species * reef + part + fpco2 + ftemp + (1 | colony), family = G
comb_dist_mod5 <- glmer(dist ~ species * part * reef * (fpco2 + ftemp) + (1 | colony), family =
```

```
fixed-effect model matrix is rank deficient so dropping 2 columns / coefficients
```

```
comb_dist_mod6 <- glmer(dist ~ species * part * fpco2 + ftemp + (1 | colony), family = Gamma(li
comb_dist_mod7 <- glmer(dist ~ reef * species * part * (fpco2 + ftemp) + (1 | colony), family =
```

```
fixed-effect model matrix is rank deficient so dropping 2 columns / coefficients
```

```
comb_dist_mod8 <- glmer(dist ~ reef * species * part * (fpco2 + ftemp) + (1 | colony), family =
```

```
fixed-effect model matrix is rank deficient so dropping 2 columns / coefficients
```

```
comb_dist_mod9 <- glmer(dist ~ species * part * (fpco2 + ftemp) + (1 | colony), family = Gamma(
comb_dist_mod10 <- glmer(dist ~ species * part * (fpco2 + ftemp) + reef + (1 | colony), family
comb_dist_mod11 <- glmer(dist ~ species * part * (fpco2 + ftemp + reef) + (1 | colony), family

# check for best-fit model
```

```
comb_plast_aic <- compare_performance(comb_dist_mod, comb_dist_mod2, comb_dist_mod3, comb_dist_
#plot(comb_plast_aic)
```

Final Model

Taken from OA_OW_Physiology_manuscript.Rmd lines 2397 - 2405

```
## Best-fit GLMM with Gamma log link
comb_dist_glm <- glmer(dist ~ species * part * (fpco2 + ftemp) + (1 | colony), family = Gamma(
comb_glm_out <- summary(comb_dist_glm) # summary output of the GLM
```

Model failed to converge, but no error.

Bootstrapping

```
## Pull the treatment and distance data from the model
newdata_comb <- data.frame(comb_dist_glm@frame[-1])

## Bootstrap distances using custom bootstrap function
comb_boot_out <- replicate(bootnum, bootFUN(model = comb_dist_glm, newdata = newdata_comb))

## Calculate the mean, 95% lowerCI, and 95% upperCI from the boot matrix and add it to datafram
comb_boot <- cbind(combined_dist, as.data.frame(t(apply(comb_boot_out, 1, function(x) c(mean(x)
colnames(comb_boot)[16:18] <- c("estimate", "lowerci", "upperci") # rename mean/CI columns

## save as rda object
save(comb_boot, file = "Data/Bootstrap/HostVSymb_PlastBoot.rda")
```

The first portion of the chunk ran smoothly, though it took several hours to run through. However, at the line where column names were assigned, there was an error that names [18] must be the same length as the vector [12]. The comb_boot frame has 12 columns, but the columns which had names being reassigned were 16-18 and didn’t exist. To solve this, I will run the code to reassign names [10:12].

```
## Pull the treatment and distance data from the model
newdata_comb <- data.frame(comb_dist_glm@frame[-1])

## Bootstrap distances using custom bootstrap function
comb_boot_out <- replicate(bootnum, bootFUN(model = comb_dist_glm, newdata = newdata_comb))

## Calculate the mean, 95% lowerCI, and 95% upperCI from the boot matrix and add it to datafram
comb_boot <- cbind(combined_dist, as.data.frame(t(apply(comb_boot_out, 1, function(x) c(mean(x)
colnames(comb_boot)[10:12] <- c("estimate", "lowerci", "upperci") # rename mean/CI columns

## save as rda object
save(comb_boot, file = "Data/Bootstrap/HostVSymb_PlastBoot.rda")
```

Species labels

Species labels

Taken from OA_OW_Physiology_manuscript.Rmd lines 2422 - 2439

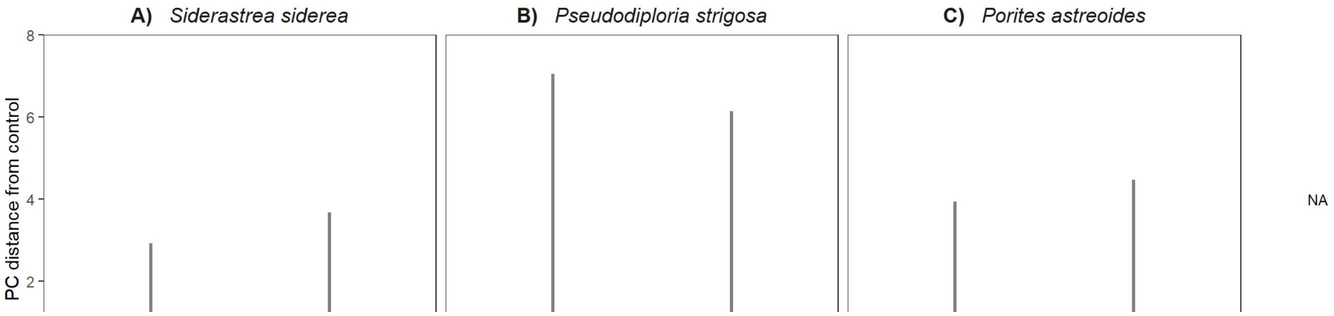
```
## Load bootstrap data from above
load(file = "Data/Bootstrap/HostVSymb_PlastBoot.rda")

## Adding pretty species labels for all dataframes
comb_boot <- comb_boot %>%
  mutate(species = factor(species, levels = c("S", "P", "A"),
                          labels = c(expression(bold("A) ")~ italic(" Siderastrea siderea")),
                                     expression(bold("B) ")~ italic(" Pseudodiploria strigosa")
                                     expression(bold("C) ")~ italic(" Porites astreoides")))),
         part = factor(part, levels = c("host", "symb"),
                       labels = c("Coral host",
                                  "Algal symbiont")),
         treat = factor(treat2, levels = c("288_28", "311_31", "405_31", "673_28", "701_31", "3
                        labels = c("pre industrial; 28C", "pre industrial; 31C", "current; 31C'
```

Runs smoothly.

Plotting

Taken from OA_OW_Physiology_manuscript.Rmd lines 2441 - 2464

```
## Plot host v symbiont plasticity per species
ggplot(comb_boot, aes(x = part, y = estimate, color = treat, fill = treat, shape = treat)) +
  theme_bw() +
  theme(legend.title = element_blank(), axis.ticks.x = element_blank(),  legend.position = "rig
  guides(shape = guide_legend(ncol = 1), fill = guide_legend(override.aes = list(linetype = 0))
  geom_point(aes(x = part, y = dist), size = 2, alpha = 0.4, position = position_jitterdodge(ji
  geom_linerange(aes(ymin = lowerci, ymax = upperci), size = 1, position = position_dodge(width
  geom_point(size = 3, stroke = 1, position = position_dodge(width = 0.7)) +
  scale_shape_manual(values = c(21, 21, 23, 24, 24, 22, 22)) +
  scale_color_manual(values = c("#b2abd2", "#b2abd2", "#5e3c99", "#fdb863", "#fdb863", "#e66101
  scale_fill_manual(values = c("#b2abd2", "white", "white", "#fdb863", "white", "#e66101", "whi
  labs(y = "PC distance from control", x = "") +
  scale_y_continuous(expand = c(0, 0), limits = c(0, 8)) +
  #scale_x_discrete(labels = c("F" = "Offshore", "N" = "Inshore")) +
  #facet_grid(species ~ part, labeller = label_parsed) +
  facet_grid( ~ species, labeller = label_parsed)
```



```
# ggsave("Figures/Supplemental_Figures/FigureS9_hostVsymb_plasticity.pdf", width = 11, height =
# ggsave("Figures/Supplemental_Figures/FigureS9_hostVsymb_plasticity.png", width = 11, height =
# ggsave("Figures/Supplemental_Figures/FigureS9_hostVsymb_plasticity.tiff", width = 11, height
```

Runs smoothly.

Table A

Taken from OA_OW_Physiology_manuscript.Rmd lines 2479 - 2528

```
## Create sample size table (phys)
N_phys_table <- as.data.frame(table(df_90$colony, paste(df_90$fpco2, df_90$ftemp, sep = "_")))
  separate(Var1, c("Species", "reef")) %>%
  separate(Var2, c("pCO2", "Treatment")) %>%
  mutate(colony = substr(Species, 2, 2),
         Species = substr(Species, 1, 1),
         pCO2 = factor(pCO2, levels = c("300", "420", "3290", "680"), labels = c("pre industria
         Treatment = factor(Treatment, labels = c("28C", "31C"))) %>%
  dplyr::group_by(Species, reef, pCO2, Treatment) %>%
  dplyr::summarise(n = sum(Freq)) %>%
  filter(Species != "T") %>%
  spread(reef, n) %>%
  mutate(group = paste(Species, pCO2, Treatment, sep = "_")) %>%
  dplyr::rename(`Offshore` = `F`, `Inshore` = `N`, ` ` = `pCO2`)
```

```
`summarise()` has grouped output by 'Species', 'reef', 'pCO2'. You can override
using the `.groups` argument.
```

```
## Create sample size table (plasticity)
N_plast_table <- rbind(ssid_boot, pstr_boot, past_boot)

N_plast_table <- as.data.frame(table(N_plast_table$colony, paste(N_plast_table$fpco2, N_plast_t
  separate(Var1, c("Species", "reef")) %>%
  separate(Var2, c("pCO2", "Treatment")) %>%
  mutate(colony = substr(Species, 2, 2),
         ` Species` = substr(Species, 1, 1),
         pCO2 = factor(pCO2, levels = c("300", "420", "3290", "680"), labels = c("pre industria
         ` Treatment` = factor(Treatment, labels = c("28C", "31C"))) %>%
  dplyr::group_by(` Species`, reef, pCO2, ` Treatment`) %>%
  dplyr::summarise(n = sum(Freq)) %>%
  filter(` Species` != "T") %>%
  spread(reef, n) %>%
  mutate(group = paste(` Species`, pCO2, ` Treatment`, sep = "_")) %>%
  mutate(group = paste(` Species`, pCO2, ` Treatment`, sep = "_")) %>%
  dplyr::rename(` Offshore` = "F", ` Inshore` = "N", `   ` = pCO2)
```

```
`summarise()` has grouped output by ' Species', 'reef', 'pCO2'. You can
override using the `.groups` argument.
```

```
## Combine both tables into single one
N_table <- merge(N_phys_table, N_plast_table, by = "group", all = TRUE)


## Update with nice formatting
Table_S1 <- kable(N_table[c(-1, -2, -7:-9)], booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  add_header_above(c(" " = 2, "Physiology N" = 2, "Plasticity N" = 2)) %>%
  pack_rows("Siderastrea Siderea", 17, 24, italic = TRUE) %>%
  pack_rows("Pseudodiploria strigosa", 9, 16, italic = TRUE) %>%
  pack_rows("Porites astreoides", 1, 8, italic = TRUE)
Table_S1
```

		Physiology N		Plasticity N	
	Treatment	Offshore	Inshore	Offshore	Inshore
Porites astreoides					
current day (uatm)	28C	6	6	NA	NA
current day (uatm)	31C	2	4	2	4
end-of-century (uatm)	28C	6	6	5	6
end-of-century (uatm)	31C	0	4	0	4
extreme (uatm)	28C	5	5	5	5
extreme (uatm)	31C	3	5	3	5
pre industrial (uatm)	28C	6	5	5	5
pre industrial (uatm)	31C	3	3	2	3
Pseudodiploria strigosa					
current day (uatm)	28C	3	2	NA	NA
current day (uatm)	31C	3	2	0	0

end-of-century (uatm)	28C	9	6	4	2
end-of-century (uatm)	31C	5	3	2	1
extreme (uatm)	28C	8	6	3	2
extreme (uatm)	31C	3	2	2	0
pre industrial (uatm)	28C	10	6	4	2
pre industrial (uatm)	31C	5	4	2	1
Siderastrea Siderea					
current day (uatm)	28C	6	5	NA	NA
current day (uatm)	31C	5	6	5	5
end-of-century (uatm)	28C	6	6	6	5
end-of-century (uatm)	31C	6	6	6	5
extreme (uatm)	28C	7	5	7	4
extreme (uatm)	31C	6	5	6	4
pre industrial (uatm)	28C	6	4	6	4
pre industrial (uatm)	31C	3	4	3	4

Runs smoothly.

Table B

Taken from OA_OW_Physiology_manuscript.Rmd lines 2538 - 2556

```
# combine AICs for different PERMANOVAs
pca_AICs <- data.frame("Species" = c("S. siderea", "P. strigosa", "P. astreoides"),
                       "full" = c(s_pca_aic_full, p_pca_aic_full, a_pca_aic_full),
                       "best" = c(s_pca_aic_final, p_pca_aic_final, a_pca_aic_final))

## update formatting of text
pca_AICs <- pca_AICs %>%
  dplyr::rename("Full interactive model AIC" = "full",
                "Best fit (additive) model AIC" = "best")

## combine them with format
Table_S2 <- kable(pca_AICs, booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  column_spec(1, italic = TRUE)
Table_S2
```

Species	Full interactive model AIC	Best fit (additive) model AIC
S. siderea	692.3	678.5
P. strigosa	696.9	685.8
P. astreoides	500.5	497.1

Runs smoothly.

Table C

Taken from OA_OW_Physiology_manuscript.Rmd lines 2566 - 2602

```
## Make Table S3 with correct labels
plast_aic <- ssid_plast_aic %>%
  bind_rows(pstr_plast_aic, past_plast_aic) %>% # combine all PERMANOVA model outputs
  separate(Name, c("Species", NA, "Formula")) %>% # create column for species ID and formula
  mutate(Formula = gsub("mod2b", Reduce(paste, deparse(ssid_dist_mod2b@call[["formula"]])), For
         Formula = gsub("mod2", Reduce(paste, deparse(ssid_dist_mod2@call[["formula"]])), Formu
         Formula = gsub("mod3", Reduce(paste, deparse(ssid_dist_mod3@call[["formula"]])), Formu
         Formula = gsub("mod4", Reduce(paste, deparse(ssid_dist_mod4@call[["formula"]])), Formu
         Formula = gsub("mod5", Reduce(paste, deparse(ssid_dist_mod5@call[["formula"]])), Formu
         Formula = gsub("mod6b", Reduce(paste, deparse(past_dist_mod6b@call[["formula"]])), For
         Formula = gsub("mod6", Reduce(paste, deparse(ssid_dist_mod6@call[["formula"]])), Formu
         Formula = gsub("mod", Reduce(paste, deparse(ssid_dist_mod@call[["formula"]])), Formula
  mutate(Formula = gsub("fpco2", "pCO2", Formula), # rename the formula components
         Formula = gsub("ftemp", "temperature", Formula),
         Formula = gsub("reef", "reef environment", Formula),
         Formula = gsub("dist ~ ", "", Formula)) %>%
  mutate(AIC = round(AIC, 1), # round the columns
         BIC = round(BIC, 1),
         R2_conditional = round(R2_conditional, 3),
         R2_marginal = round(R2_marginal, 3)) %>%
  dplyr::rename(`Model formula` = Formula,
         `Conditional R2` = R2_conditional,
         `Marginal R2` = R2_marginal)


## combine them with format
Table_S3 <- kable(plast_aic[c(2, 4, 6, 7)], booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  pack_rows("Siderastrea Siderea", 1, 7, italic = TRUE) %>%
  pack_rows("Pseudodiploria strigosa", 8, 11, italic = TRUE) %>%
  pack_rows("Porites astreoides", 12, 18, italic = TRUE) %>%
  row_spec(c(3,11,17), bold = TRUE, background = "lightgrey")
Table_S3
```

Runs smoothly, but columns should be conditional R^2 and marginal R^2. The indices being used correspond to the wrong columns, so they will be changed to 2, 4, 10, and 11. The values are slightly off, but there is no clear reason why.

```
## Make Table S3 with correct labels
plast_aic <- ssid_plast_aic %>%
  bind_rows(pstr_plast_aic, past_plast_aic) %>% # combine all PERMANOVA model outputs
  separate(Name, c("Species", NA, "Formula")) %>% # create column for species ID and formula
  mutate(Formula = gsub("mod2b", Reduce(paste, deparse(ssid_dist_mod2b@call[["formula"]])), For
         Formula = gsub("mod2", Reduce(paste, deparse(ssid_dist_mod2@call[["formula"]])), Formu
         Formula = gsub("mod3", Reduce(paste, deparse(ssid_dist_mod3@call[["formula"]])), Formu
         Formula = gsub("mod4", Reduce(paste, deparse(ssid_dist_mod4@call[["formula"]])), Formu
         Formula = gsub("mod5", Reduce(paste, deparse(ssid_dist_mod5@call[["formula"]])), Formu
         Formula = gsub("mod6b", Reduce(paste, deparse(past_dist_mod6b@call[["formula"]])), For
         Formula = gsub("mod6", Reduce(paste, deparse(ssid_dist_mod6@call[["formula"]])), Formu
         Formula = gsub("mod", Reduce(paste, deparse(ssid_dist_mod@call[["formula"]])), Formula
  mutate(Formula = gsub("fpco2", "pCO2", Formula), # rename the formula components
         Formula = gsub("ftemp", "temperature", Formula),
         Formula = gsub("reef", "reef environment", Formula),
         Formula = gsub("dist ~ ", "", Formula)) %>%
  mutate(AIC = round(AIC, 1), # round the columns
         BIC = round(BIC, 1),
         R2_conditional = round(R2_conditional, 3),
         R2_marginal = round(R2_marginal, 3)) %>%
  dplyr::rename(`Model formula` = Formula,
         `Conditional R2` = R2_conditional,
         `Marginal R2` = R2_marginal)


## combine them with format
Table_S3 <- kable(plast_aic[c(2, 4, 10, 11)], booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  pack_rows("Siderastrea Siderea", 1, 7, italic = TRUE) %>%
  pack_rows("Pseudodiploria strigosa", 8, 11, italic = TRUE) %>%
  pack_rows("Porites astreoides", 12, 18, italic = TRUE) %>%
  row_spec(c(3,11,17), bold = TRUE, background = "lightgrey")
Table_S3
```

Model formula	AIC	Conditional R2	Marginal R2
Siderastrea Siderea			

reef environment * pCO2 * temperature + (1 &#124; colony)	223.2	0.555	0.373
reef environment * pCO2 + temperature + (1 &#124; colony)	218.8	0.517	0.330
reef environment * pCO2 + temperature + (1 &#124; colony) + (1 &#124; tank)	**218.4**	**0.553**	**0.313**
reef environment + pCO2 * temperature + (1 &#124; colony)	225.6	0.454	0.260
reef environment + pCO2 + temperature + (1 &#124; colony)	221.6	0.454	0.261
reef environment * (pCO2 + temperature) + (1 &#124; colony)	220.1	0.523	0.336
pCO2 + temperature + (1 &#124; colony)	222.1	0.382	0.091
Pseudodiploria strigosa			
reef environment * pCO2 * temperature + (1 &#124; colony)	110.7	0.441	0.358
reef environment * pCO2 + temperature + (1 &#124; colony)	106.0	0.354	0.303
reef environment + pCO2 * temperature + (1 &#124; colony)	106.9	0.326	0.283
pCO2 + temperature + (1 &#124; colony)	**102.8**	**0.280**	**0.235**
Porites astreoides			
reef environment * pCO2 * temperature + (1 &#124; colony)	153.1	0.534	0.202
reef environment * pCO2 + temperature + (1 &#124; colony)	145.9	0.528	0.197
reef environment + pCO2 * temperature + (1 &#124; colony)	146.2	0.506	0.176
reef environment + pCO2 + temperature + (1 &#124; colony)	142.3	0.506	0.176
reef environment * (pCO2 + temperature) + (1 &#124; colony)	147.9	0.528	0.198
pCO2 + temperature + (1 &#124; colony)	**140.4**	**0.492**	**0.149**
pCO2 + temperature + (1 &#124; colony) + (1 &#124; tank)	142.4	0.500	0.147

Table D

Taken from OA_OW_Physiology_manuscript.Rmd lines 2614 - 2646

```
## combine model output
pca_mod_out <- rbind(s_pca_mod, p_pca_mod, a_pca_mod) # combine all PERMANOVA model outputs
pca_mod_out$treat <- rownames(pca_mod_out) # add column for rownames
pca_mod_out$species <- c(rep("SSID", 5), rep("PSTR", 5), rep("PAST", 5)) # add column for speci
pca_mod_out <- pca_mod_out[c(6, 1:5, 7)] # reorder columns so treatment is first


## update formatting of text
pca_mod_out <- pca_mod_out %>%
  mutate(SumOfSqs = round(as.numeric(SumOfSqs), 0), # round values
         R2 = round(as.numeric(R2), 3),
         `F` = round(as.numeric(`F`), 2),
         `Pr(>F)` = round(as.numeric(`Pr(>F)`), 5)) %>%
  mutate(treat = gsub("[0-9]", "", treat), # rename the treatment IDs
  mutate(treat = gsub("[0-9]", "", treat), # rename the treatment IDs
         treat = gsub("fpco", "pCO2", treat),
         treat = gsub("ftemp", "temperature", treat),
         treat = gsub("reef", "reef environment", treat)) %>%
  dplyr::rename("P-value" = `Pr(>F)`, # rename columns
         " " = treat,
         "Sum of Squares" = SumOfSqs)

## combine them with format
Table_S4 <- kable(pca_mod_out[-7], booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  pack_rows("Siderastrea Siderea", 1, 5, italic = TRUE) %>%
  pack_rows("Pseudodiploria strigosa", 6, 10, italic = TRUE) %>%
  pack_rows("Porites astreoides", 11, 15, italic = TRUE) %>%
  row_spec(c(4:5,9:10,14:15), italic = TRUE)
Table_S4
```

	Df	Sum of Squares	R2	F	P-value
Siderastrea Siderea					
pCO2	3	59423	0.203	7.93	0.00050
temperature	1	9320	0.032	3.73	0.04598
reef environment	1	24705	0.084	9.89	0.00200
Residual	*80*	*199740*	*0.682*	*NA*	*NA*
Total	*85*	*292988*	*1.000*	*NA*	*NA*
Pseudodiploria strigosa					
reef environment	1	101796	0.090	14.87	0.00100
temperature	1	519372	0.460	75.84	0.00050
pCO2	3	30444	0.027	1.48	0.22289
Residual	*71*	*486202*	*0.430*	*NA*	*NA*
Total	*76*	*1130099*	*1.000*	*NA*	*NA*
Porites astreoides					
reef environment	1	724	0.005	0.53	0.47176
temperature	1	27051	0.191	19.66	0.00050
pCO2	3	30537	0.216	7.40	0.00050
Residual	*62*	*85309*	*0.603*	*NA*	*NA*
Total	*67*	*141417*	*1.000*	*NA*	*NA*

Runs smoothly. Numbers are slightly off, but there is no clear reason why.

Table E

Taken from OA_OW_Physiology_manuscript.Rmd lines 2658 - 2702

```
## combine model output per species
dist_mod_out <- rbind.fill(data.frame(treat = rownames(ssid_glm_out[["coefficients"]]), ssid_gl
                           data.frame(treat = rownames(data.frame(unlist(r2_nakagawa(ssid_dist_
                           data.frame(treat = rownames(pstr_glm_out[["coefficients"]]), pstr_gl
                           data.frame(treat = rownames(data.frame(unlist(r2_nakagawa(pstr_dist_
                           data.frame(treat = rownames(past_glm_out[["coefficients"]]), past_gl
                           data.frame(treat = rownames(data.frame(unlist(r2_nakagawa(past_dist_
dist_mod_out$species <- c(rep("SSID", 11), rep("PSTR", 6), rep("PAST", 7)) # add column for spe



## update formatting of text
dist_mod_out <- dist_mod_out %>%
  mutate(Estimate = round(as.numeric(Estimate), 3), # round values
         Std..Error = round(as.numeric(Std..Error), 3),
         t.value = round(as.numeric(t.value), 2),
         Pr...z.. = round(as.numeric(Pr...z..), 3)) %>%
  mutate(treat = gsub("[.]", " ", treat), # rename the treatment IDs
         treat = gsub(" ", "", treat),
         treat = gsub("2420", "-current", treat),
         treat = gsub("2680", "-EOC", treat),
         treat = gsub("23290", "-extreme", treat),
         treat = gsub("[0-9]", "", treat),
         treat = gsub("fpco", "pCO2", treat),
         treat = gsub("ftemp", "temperature (31C)", treat),
         treat = gsub("reefN", "reef environment (offshore)", treat),
         treat = gsub("R_conditionalConditionalR", "Conditional R2", treat),
         treat = gsub("R_marginalMarginalR", "Marginal R2", treat)) %>%
  dplyr::rename(" " = treat, # rename columns
        "Standard error" = Std..Error,
        "Statistic" = t.value,
        "P-value" = Pr...z..)

## combine them with format
Table_S5 <- kable(dist_mod_out[,-6], booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  pack_rows("Siderastrea Siderea", 1, 11, italic = TRUE) %>%
  pack_rows("Pseudodiploria strigosa", 12, 17, italic = TRUE) %>%
  pack_rows("Porites astreoides", 18, 24, italic = TRUE) %>%
  row_spec(c(10:11,16:17,23:24), italic = TRUE)
Table_S5
```

	Estimate	Standard error	Statistic	P-value
Siderastrea Siderea				
(Intercept)	1.050	0.006	187.16	0.000
reef environment (offshore)	0.010	0.006	1.73	0.084

reef environment (offshore)	[illegible]	[illegible]	[illegible]	[illegible]
pCO2-current	0.378	0.006	67.01	0.000
pCO2-EOC	0.223	0.006	39.36	0.000
pCO2-extreme	0.444	0.006	78.84	0.000
temperature (31C)	0.002	0.006	0.32	0.748
reef environment (offshore):pCO2-current	-0.771	0.006	-136.94	0.000
reef environment (offshore):pCO2-EOC	-0.446	0.006	-79.40	0.000
reef environment (offshore):pCO2-extreme	-0.330	0.006	-58.53	0.000
Conditional R2	*0.553*	*NA*	*NA*	*NA*
Marginal R2	*0.313*	*NA*	*NA*	*NA*
Pseudodiploria strigosa				
(Intercept)	1.279	0.148	8.66	0.000
pCO2-EOC	-0.338	0.193	-1.75	0.080
pCO2-extreme	-0.059	0.187	-0.31	0.753
temperature (31C)	0.227	0.173	1.31	0.190
Conditional R2	*0.244*	*NA*	*NA*	*NA*
Marginal R2	*0.198*	*NA*	*NA*	*NA*
Porites astreoides				
(Intercept)	1.033	0.125	8.25	0.000
pCO2-current	-0.045	0.115	-0.39	0.694
pCO2-EOC	0.033	0.079	0.42	0.676
pCO2-extreme	0.124	0.082	1.51	0.131
temperature (31C)	0.262	0.069	3.82	0.000
Conditional R2	*0.500*	*NA*	*NA*	*NA*
Marginal R2	*0.147*	*NA*	*NA*	*NA*

Runs smoothly. Some numbers are slightly off for no clear reason.

Table F

Taken from OA_OW_Physiology_manuscript.Rmd lines 2714 - 2741

```
## combine model output
all_pca_mod$treat <- rownames(all_pca_mod) # add column for rownames
all_pca_mod <- all_pca_mod[c(6, 1:5)] # reorder columns so treatment is first



## update formatting of text
all_pca_mod <- all_pca_mod %>%
  mutate(SumOfSqs = round(as.numeric(SumOfSqs), 0), # round values
  mutate(SumOfSqs = round(as.numeric(SumOfSqs), 0), # round values
         R2 = round(as.numeric(R2), 3),
         `F` = round(as.numeric(`F`), 2),
         `Pr(>F)` = round(as.numeric(`Pr(>F)`), 5)) %>%
  mutate(treat = gsub("[0-9]", "", treat), # rename the treatment IDs
         treat = gsub("fpco", "pCO2", treat),
         treat = gsub("ftemp", "temperature", treat),
         treat = gsub("reef", "reef environment", treat)) %>%
  dplyr::rename("P-value" = `Pr(>F)`, # rename columns
         " " = treat,
         "Sum of Squares" = SumOfSqs)

## combine them with format
Table_S6 <- kable(all_pca_mod, booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  row_spec(c(8:9), italic = TRUE)
Table_S6
```

Received error of subscript out of bounds. There seems to be three rows, so the 2nd and 3rd rows will be put in italics instead to avoid subscript error. Table rendered is not the same as in paper, unclear as to why.

```
## combine model output
all_pca_mod$treat <- rownames(all_pca_mod) # add column for rownames
all_pca_mod <- all_pca_mod[c(6, 1:5)] # reorder columns so treatment is first


## update formatting of text
all_pca_mod <- all_pca_mod %>%
  mutate(SumOfSqs = round(as.numeric(SumOfSqs), 0), # round values
         R2 = round(as.numeric(R2), 3),
         `F` = round(as.numeric(`F`), 2),
         `Pr(>F)` = round(as.numeric(`Pr(>F)`), 5)) %>%
  mutate(treat = gsub("[0-9]", "", treat), # rename the treatment IDs
         treat = gsub("fpco", "pCO2", treat),
         treat = gsub("ftemp", "temperature", treat),
         treat = gsub("reef", "reef environment", treat)) %>%
  dplyr::rename("P-value" = `Pr(>F)`, # rename columns
         " " = treat,
         "Sum of Squares" = SumOfSqs)

## combine them with format
Table_S6 <- kable(all_pca_mod, booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  row_spec(c(2:3), italic = TRUE)
Table_S6
```

	Df	Sum of Squares	R2	F	P-value
Model	17	2588852	0.667	25.2	5e-04

Model	17	2588852	0.667	25.2	5e-04
Residual	*214*	*1293204*	*0.333*	*NA*	*NA*
Total	*231*	*3882055*	*1.000*	*NA*	*NA*

Table G

Taken from OA_OW_Physiology_manuscript.Rmd lines 2754 - 2800

```
## combine model output
pca_host_mod_out <- rbind(s_pca_host_mod, p_pca_host_mod, a_pca_host_mod) # combine all PERMANO
pca_symb_mod_out <- rbind(s_pca_symb_mod, p_pca_symb_mod, a_pca_symb_mod) # combine all PERMANO
pca_host_symb_mod_out <- cbind(pca_host_mod_out, pca_symb_mod_out)


pca_host_symb_mod_out$treat <- rownames(pca_host_symb_mod_out) # add column for rownames
pca_host_symb_mod_out$species <- c(rep("SSID", 5), rep("PSTR", 5), rep("PAST", 5)) # add column
pca_host_symb_mod_out <- pca_host_symb_mod_out[c(11, 1:10, 12)] # reorder columns so treatment


## update formatting of text
pca_host_symb_mod_out <- pca_host_symb_mod_out %>%
  mutate(SumOfSqs = round(as.numeric(SumOfSqs), 0), # round values from host
         R2 = round(as.numeric(R2), 3),
         `F` = round(as.numeric(`F`), 2),
         `Pr(>F)` = round(as.numeric(`Pr(>F)`), 5)) %>%
  mutate(SumOfSqs.1 = round(as.numeric(SumOfSqs.1), 0), # round values from symbiont
         R2.1 = round(as.numeric(R2.1), 3),
         `F.1` = round(as.numeric(`F.1`), 2),
         `Pr(>F).1` = round(as.numeric(`Pr(>F).1`), 5)) %>%
  mutate(treat = gsub("[0-9]", "", treat), # rename the treatment IDs
         treat = gsub("fpco", "pCO2", treat),
         treat = gsub("ftemp", "temperature", treat),
         treat = gsub("reef", "reef environment", treat)) %>%
  dplyr::rename("P-value" = `Pr(>F)`, # rename columns
         " " = treat,
         "Sum of Squares" = SumOfSqs,
         "Df " = Df.1,
         "Sum of Squares " = SumOfSqs.1,
         "R2 " = R2.1,
         "F " = F.1,
         "P-value " = `Pr(>F).1`)

## combine them with format
Table_S7 <- kable(pca_host_symb_mod_out[-12], booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  add_header_above(c(" " = 1, "Coral host" = 5, "Algal symbionts" = 5)) %>%
  pack_rows("Siderastrea Siderea", 1, 5, italic = TRUE) %>%
  pack_rows("Pseudodiploria strigosa", 6, 10, italic = TRUE) %>%
  pack_rows("Pseudodiploria strigosa", 6, 10, italic = TRUE) %>%
  pack_rows("Porites astreoides", 11, 15, italic = TRUE) %>%
  row_spec(c(4:5,9:10,14:15), italic = TRUE)
Table_S7
```

Received error that the replacement has 15 rows while the data has 9. The pca_host_symb_mod_out frame has 9 rows, and we are adding in 5 rows in three categories (15). So, I will change the number from 5 to 3 in each and proceed. There was also an error in the combine them with format section for an out of bounds subscript. I will change the subscripts each to be 2:3, 5:6, and 8:9 respectively so that it is within the 9 rows. Table rendered is not the same as in paper, unclear as to why.

```
## combine model output
pca_host_mod_out <- rbind(s_pca_host_mod, p_pca_host_mod, a_pca_host_mod) # combine all PERMANO
pca_symb_mod_out <- rbind(s_pca_symb_mod, p_pca_symb_mod, a_pca_symb_mod) # combine all PERMANO
pca_host_symb_mod_out <- cbind(pca_host_mod_out, pca_symb_mod_out)


pca_host_symb_mod_out$treat <- rownames(pca_host_symb_mod_out) # add column for rownames
pca_host_symb_mod_out$species <- c(rep("SSID", 3), rep("PSTR", 3), rep("PAST", 3)) # add column
pca_host_symb_mod_out <- pca_host_symb_mod_out[c(11, 1:10, 12)] # reorder columns so treatment


## update formatting of text
pca_host_symb_mod_out <- pca_host_symb_mod_out %>%
  mutate(SumOfSqs = round(as.numeric(SumOfSqs), 0), # round values from host
         R2 = round(as.numeric(R2), 3),
         `F` = round(as.numeric(`F`), 2),
         `Pr(>F)` = round(as.numeric(`Pr(>F)`), 5)) %>%
  mutate(SumOfSqs.1 = round(as.numeric(SumOfSqs.1), 0), # round values from symbiont
         R2.1 = round(as.numeric(R2.1), 3),
         `F.1` = round(as.numeric(`F.1`), 2),
         `Pr(>F).1` = round(as.numeric(`Pr(>F).1`), 5)) %>%
  mutate(treat = gsub("[0-9]", "", treat), # rename the treatment IDs
         treat = gsub("fpco", "pCO2", treat),
         treat = gsub("ftemp", "temperature", treat),
         treat = gsub("reef", "reef environment", treat)) %>%
  dplyr::rename("P-value" = `Pr(>F)`, # rename columns
         " " = treat,
         "Sum of Squares" = SumOfSqs,
         "Df " = Df.1,
         "Sum of Squares " = SumOfSqs.1,
         "R2 " = R2.1,
         "F " = F.1,
         "P-value " = `Pr(>F).1`)

## combine them with format
Table_S7 <- kable(pca_host_symb_mod_out[-12], booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  add_header_above(c(" " = 1, "Coral host" = 5, "Algal symbionts" = 5)) %>%
  pack_rows("Siderastrea Siderea", 1, 3, italic = TRUE) %>%
  pack_rows("Siderastrea Siderea", 1, 3, italic = TRUE) %>%
  pack_rows("Pseudodiploria strigosa", 4, 6, italic = TRUE) %>%
  pack_rows("Porites astreoides", 7, 9, italic = TRUE) %>%
  row_spec(c(2:3,5:6,8:9), italic = TRUE)
Table_S7
```

	Coral host					Algal symbionts				
	Df	**Sum of Squares**	**R2**	**F**	**P-value**	**Df**	**Sum of Squares**	**R2**	**F**	**P-value**
Siderastrea Siderea										
Model	5	3	0.099	1.76	0.06947	5	93229	0.318	7.47	5e-04
Residual	*80*	*30*	*0.901*	*NA*	*NA*	*80*	*199684*	*0.682*	*NA*	*NA*
Total	*85*	*34*	*1.000*	*NA*	*NA*	*85*	*292913*	*1.000*	*NA*	*NA*
Pseudodiploria strigosa										
Model	5	4	0.207	3.71	0.00100	5	643865	0.570	18.81	5e-04
Residual	*71*	*14*	*0.793*	*NA*	*NA*	*71*	*486140*	*0.430*	*NA*	*NA*
Total	*76*	*18*	*1.000*	*NA*	*NA*	*76*	*1130005*	*1.000*	*NA*	*NA*
Porites astreoides										
Model	5	2	0.193	2.97	0.00850	5	56098	0.397	8.16	5e-04
Residual	*62*	*10*	*0.807*	*NA*	*NA*	*62*	*85288*	*0.603*	*NA*	*NA*
Total	*67*	*13*	*1.000*	*NA*	*NA*	*67*	*141387*	*1.000*	*NA*	*NA*

Cut Table host/symb

Taken from OA_OW_Physiology_manuscript.Rmd lines 2813 - 2849

```
## Make Table S3 with correct labels
comb_plast_aic <- comb_plast_aic %>%
  #bind_rows(pstr_plast_aic, past_plast_aic) %>% # combine all PERMANOVA model outputs
  separate(Name, c(NA, NA, "Formula")) %>% # create column for species ID and formula
  mutate(Formula = gsub("mod2", Reduce(paste, deparse(comb_dist_mod2@call[["formula"]])), Formu
         Formula = gsub("mod3", Reduce(paste, deparse(comb_dist_mod3@call[["formula"]])), Formu
         Formula = gsub("mod4", Reduce(paste, deparse(comb_dist_mod4@call[["formula"]])), Formu
         Formula = gsub("mod5", Reduce(paste, deparse(comb_dist_mod5@call[["formula"]])), Formu
         Formula = gsub("mod6", Reduce(paste, deparse(comb_dist_mod6@call[["formula"]])), Formu
         Formula = gsub("mod7", Reduce(paste, deparse(comb_dist_mod7@call[["formula"]])), Formu
         Formula = gsub("mod8", Reduce(paste, deparse(comb_dist_mod8@call[["formula"]])), Formu
         Formula = gsub("mod9", Reduce(paste, deparse(comb_dist_mod9@call[["formula"]])), Formu
         Formula = gsub("mod10", Reduce(paste, deparse(comb_dist_mod10@call[["formula"]])), For
         Formula = gsub("mod11", Reduce(paste, deparse(comb_dist_mod11@call[["formula"]])), For
         Formula = gsub("mod", Reduce(paste, deparse(comb_dist_mod@call[["formula"]])), Formula
  mutate(Formula = gsub("fpco2", "pCO2", Formula), # rename the formula components
         Formula = gsub("ftemp", "temperature", Formula),
         Formula = gsub("reef", "reef environment", Formula),
         Formula = gsub("dist ~ ", "", Formula)) %>%
  mutate(AIC = round(AIC, 1), # round the columns
         BIC = round(BIC, 1),
         R2_conditional = round(R2_conditional, 3),
         R2_marginal = round(R2_marginal, 3)) %>%
  dplyr::rename(`Model formula` = Formula,
         `Conditional R2` = R2_conditional,
         `Marginal R2` = R2_marginal)


## combine them with format
Table_S8 <- kable(comb_plast_aic[c(1, 3, 5, 6)], booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  row_spec(6, bold = TRUE, background = "lightgrey")
Table_S8
```

Model formula	AIC	AICc	AICc_wt
species * part * reef environment * pCO2 * temperature + (1 &#124; colony)	923.1	982.2465	0.0000000
species * part * reef environment * pCO2 + temperature + (1 &#124; colony)	904.6	924.2461	0.0000000
species * part * reef environment + pCO2 * temperature + (1 &#124; colony)	890.2	893.1746	0.0045846
species * reef environment + part + pCO2 + temperature + (1 &#124; colony)	881.2	882.4859	0.9600943
species * part * reef environment * (pCO2 + temperature) + (1 &#124; colony)	902.1	932.7122	0.0000000
species * part * pCO2 + temperature + (1 &#124; colony)	**890.2**	**895.7945**	**0.0012371**
reef environment * species * part * (pCO2 + temperature) + (1 &#124; colony)	902.1	932.7122	0.0000000
reef environment * species * part * (pCO2 + temperature) + (1 &#124; colony)	902.1	932.7122	0.0000000
species * part * (pCO2 + temperature) + (1 &#124; colony)	882.0	889.8614	0.0240303
species * part * (pCO2 + temperature) + reef environment + (1 &#124; colony)	883.5	891.9518	0.0084495
species * part * (pCO2 + temperature + reef environment) + (1 &#124; colony)	883.9	895.2747	0.0016042

Runs smoothly.

GLMM Table Host/Symb

Taken from OA_OW_Physiology_manuscript.Rmd lines 2863 - 2902

```
## combine model output per species
comb_dist_mod_out <- rbind.fill(data.frame(treat = rownames(comb_glm_out[["coefficients"]]),
                                      comb_glm_out[["coefficients"]]),
                           data.frame(treat = rownames(data.frame(unlist(r2_nakagawa(comb_dist_
                                      Estimate = unlist(r2_nakagawa(comb_dist_glm))))


## update formatting of text
## update formatting of text
comb_dist_mod_out <- comb_dist_mod_out %>%
  mutate(Estimate = round(as.numeric(Estimate), 3), # round values
         Std..Error = round(as.numeric(Std..Error), 3),
         t.value = round(as.numeric(t.value), 2),
         Pr...z.. = round(as.numeric(Pr...z..), 3)) %>%
  mutate(treat = gsub("[.]", " ", treat), # rename the treatment IDs
         treat = gsub(" ", "", treat),
         treat = gsub("2420", "-current", treat),
         treat = gsub("2680", "-EOC", treat),
         treat = gsub("23290", "-extreme", treat),
         treat = gsub("[0-9]", "", treat),
         treat = gsub("fpco", "pCO2", treat),
         treat = gsub("ftemp", "temperature (31C)", treat),
         treat = gsub("speciesP", "PSTR", treat),
         treat = gsub("speciesA", "PAST", treat),
         treat = gsub("partsymb", "symbionts", treat),
         treat = gsub("R_conditionalConditionalR", "Conditional R2", treat),
         treat = gsub("R_marginalMarginalR", "Marginal R2", treat)) %>%
  dplyr::rename(" " = treat, # rename columns
        "Standard error" = Std..Error,
        "Statistic" = t.value,
        "P-value" = Pr...z..)

## combine them with format
Table_S9 <- kable(comb_dist_mod_out, booktabs = TRUE, row.names = FALSE) %>%
  kable_styling(font_size = 12, full_width = FALSE) %>%
  row_spec(31:32, italic = TRUE)
Table_S9
```

	Estimate	Standard error	Statistic	P-value
(Intercept)	0.800	0.172	4.66	0.000
PSTR	0.199	0.290	0.69	0.492
PAST	-0.357	0.244	-1.46	0.143
symbionts	-0.515	0.185	-2.78	0.005
pCO2-current	0.234	0.214	1.10	0.273
pCO2-EOC	0.005	0.164	0.03	0.974
pCO2-extreme	-0.021	0.161	-0.13	0.895
temperature (31C)	-0.252	0.128	-1.96	0.050
PSTR:symbionts	0.223	0.308	0.72	0.470
PAST:symbionts	0.685	0.270	2.53	0.011
PSTR:pCO2-current	-0.088	0.592	-0.15	0.882
PAST:pCO2-current	-0.568	0.339	-1.68	0.094
PSTR:pCO2-EOC	-0.450	0.298	-1.51	0.131

PAST:pCO2-EOC	0.051	0.240	0.21	0.833
PSTR:pCO2-extreme	-0.154	0.288	-0.53	0.593
PAST:pCO2-extreme	0.236	0.245	0.96	0.335
PSTR:temperature (31C)	0.072	0.250	0.29	0.774
PAST:temperature (31C)	0.608	0.200	3.04	0.002
symbionts:pCO2-current	-0.299	0.298	-1.00	0.316
symbionts:pCO2-EOC	0.134	0.227	0.59	0.555
symbionts:pCO2-extreme	0.564	0.226	2.50	0.013
symbionts:temperature (31C)	0.524	0.181	2.89	0.004
PSTR:symbionts:pCO2-current	0.362	0.818	0.44	0.659
PAST:symbionts:pCO2-current	0.673	0.473	1.42	0.154
PSTR:symbionts:pCO2-EOC	-0.179	0.418	-0.43	0.669
PAST:symbionts:pCO2-EOC	-0.180	0.335	-0.54	0.590
PSTR:symbionts:pCO2-extreme	-0.482	0.401	-1.20	0.229
PAST:symbionts:pCO2-extreme	-0.890	0.341	-2.61	0.009
PSTR:symbionts:temperature (31C)	0.189	0.350	0.54	0.589
PAST:symbionts:temperature (31C)	-0.607	0.281	-2.16	0.031
Conditional R2	*0.302*	*NA*	*NA*	*NA*
Marginal R2	*0.150*	*NA*	*NA*	*NA*

Runs smoothly.

Excel WB

Taken from OA_OW_Physiology_manuscript.Rmd lines 2912 - 2966

```
### Create workbook
wb <- createWorkbook()

# add 'Table S1' worksheet
addWorksheet(wb, "Table A - Sample Size")
writeData(wb, sheet = 1, x = N_table)
setColWidths(wb, sheet = 1, cols = 1:11, widths = "auto")

# add 'Table S2' worksheet
addWorksheet(wb, "Table B - PERMANOVA AIC")
writeData(wb, sheet = 2, x = pca_AICs)
setColWidths(wb, sheet = 2, cols = 1:3, widths = "auto")

# add 'Table S3' worksheet
addWorksheet(wb, "Table C - Plasticity AIC")
```

```
writeData(wb, sheet = 3, x = plast_aic)
setColWidths(wb, sheet = 3, cols = 1:7, widths = "auto")

# add 'Table S4' worksheet
addWorksheet(wb, "Table D - PERMANOVA")
writeData(wb, sheet = 4, x = pca_mod_out)
setColWidths(wb, sheet = 4, cols = 1:7, widths = "auto")

# add 'Table S5' worksheet
addWorksheet(wb, "Table E - Plasticity GLM")
writeData(wb, sheet = 5, x = dist_mod_out)
setColWidths(wb, sheet = 5, cols = 1:6, widths = "auto")

# add 'Table S6' worksheet
addWorksheet(wb, "Table F - Species PERMANOVA")
writeData(wb, sheet = 6, x = all_pca_mod)
setColWidths(wb, sheet = 6, cols = 1:6, widths = "auto")

# add 'Table S7' worksheet
addWorksheet(wb, "Table G - HostVsymb PERMANOVA")
writeData(wb, sheet = 7, x = pca_host_symb_mod_out)
setColWidths(wb, sheet = 7, cols = 1:12, widths = "auto")

# add 'HostVsymb Plast AIC' worksheet
addWorksheet(wb, "HostVsymb Plast AIC")
writeData(wb, sheet = 8, x = comb_plast_aic)
setColWidths(wb, sheet = 8, cols = 1:7, widths = "auto")

# add 'HostVsymb Plast GLM' worksheet
addWorksheet(wb, "HostVsymb Plast GLM")
writeData(wb, sheet = 9, x = comb_dist_mod_out)
setColWidths(wb, sheet = 9, cols = 1:6, widths = "auto")


# save workbook
saveWorkbook(wb, file="Data/Supplemental/Supplemental_Tables.xlsx", overwrite = TRUE)
```

Runs smoothly.

Session Info

Taken from OA_OW_Physiology_manuscript.Rmd lines 2972 - 2976

```
sessionInfo()
```

```
R version 4.4.1 (2024-06-14 ucrt)
Platform: x86_64-w64-mingw32/x64
Running under: Windows 11 x64 (build 26100)
```

```
Matrix products: default


locale:
[1] LC_COLLATE=English_United States.utf8
[2] LC_CTYPE=English_United States.utf8
[3] LC_MONETARY=English_United States.utf8
[4] LC_NUMERIC=C
[5] LC_TIME=English_United States.utf8

time zone: America/Chicago
tzcode source: internal

attached base packages:
[1] grid      stats     graphics  grDevices utils     datasets  methods
[8] base

other attached packages:
 [1] ggrepel_0.9.6      xts_0.14.1         zoo_1.8-13         raster_3.6-32
 [5] sp_2.2-0           ncdf4_1.24         janitor_2.2.1      rcompanion_2.5.0
 [9] car_3.1-3          carData_3.0-5      png_0.1-8          MASS_7.3-64
[13] performance_0.14.0 wesanderson_0.3.7  RColorBrewer_1.1-3 gridGraphics_0.5-1
[17] corrplot_0.95      Hmisc_5.2-3        magick_2.8.5       ggpubr_0.6.0
[21] vroom_1.6.5        lmerTest_3.1-3     lme4_1.1-36        Matrix_1.7-0
[25] kableExtra_1.4.0   finalfit_1.0.8     ggfortify_0.4.17   cowplot_1.1.3
[29] Rmisc_1.5.1        plyr_1.8.9         lattice_0.22-6     shiny_1.10.0
[33] vegan_2.7-1        permute_0.9-7      lubridate_1.9.4    forcats_1.0.0
[37] stringr_1.5.1      purrr_1.0.2        tibble_3.2.1       tidyverse_2.0.0
[41] plotly_4.10.4      openxlsx_4.2.8     corrgram_1.14      tidyr_1.3.1
[45] ggbiplot_0.6.2     dplyr_1.1.4        ggplot2_3.5.2      broom_1.0.7
[49] readr_2.1.5        knitr_1.49

loaded via a namespace (and not attached):
  [1] splines_4.4.1       later_1.4.1         cellranger_1.1.0
  [4] rpart_4.1.23        lifecycle_1.0.4     Rdpack_2.6.3
  [7] rstatix_0.7.2       insight_1.3.0       backports_1.5.0
 [10] magrittr_2.0.3      rmarkdown_2.29      yaml_2.3.10
 [13] httpuv_1.6.15       zip_2.3.1           gld_2.6.7
 [16] minqa_1.2.8         multcomp_1.4-28     abind_1.4-8
 [19] expm_1.0-0          nnet_7.3-19         TH.data_1.1-3
 [22] sandwich_3.1-1      terra_1.8-54        nortest_1.0-4
 [25] svglite_2.1.3       codetools_0.2-20    coin_1.4-3
 [28] xml2_1.3.6          tidyselect_1.2.1    shape_1.4.6.1
 [31] farver_2.1.2        matrixStats_1.5.0   stats4_4.4.1
 [34] base64enc_0.1-3     jsonlite_1.8.9      e1071_1.7-16
 [37] mitml_0.4-5         Formula_1.2-5       survival_3.6-4
 [40] iterators_1.0.14    systemfonts_1.1.0   foreach_1.5.2
 [43] tools_4.4.1         ragg_1.3.3          DescTools_0.99.60
 [46] Rcpp_1.0.13-1       glue_1.8.0          gridExtra_2.3
 [49] pan_1.9            xfun_0.50          mgcv_1.9-1
 [52] withr_3.0.2        numDeriv_2016.8-1.1 fastmap_1.2.0
 [55] boot_1.3-30        digest_0.6.37      timechange_0.3.0
 [58] R6_2.5.1           mime_0.12          textshaping_0.4.1
 [61] mice_3.18.0        colorspace_2.1-1   generics_0.1.3
 [64] data.table_1.16.4  class_7.3-22       httr_1.4.7
 [67] htmlwidgets_1.6.4  pkgconfig_2.0.3    gtable_0.3.6
 [70] Exact_3.3          modeltools_0.2-24  lmtest_0.9-40
 [73] htmltools_0.5.8.1  multcompView_0.1-10 scales_1.3.0
 [76] lmom_3.2           snakecase_0.11.1   reformulas_0.4.0
 [79] rstudioapi_0.17.1  tzdb_0.4.0         checkmate_2.3.2
 [82] nlme_3.1-164       nloptr_2.2.1       proxy_0.4-27
 [85] rootSolve_1.8.2.4  parallel_4.4.1     libcoin_1.0-10
 [88] foreign_0.8-86     pillar_1.10.1      vctrs_0.6.5
 [91] promises_1.3.2     jomo_2.7-6         xtable_1.8-4
 [94] cluster_2.1.6      htmlTable_2.4.3    evaluate_1.0.1
 [97] mvtnorm_1.3-3      cli_3.6.3          compiler_4.4.1
[100] rlang_1.1.4        crayon_1.5.3       ggsignif_0.6.4
[103] labeling_0.4.3     fs_1.6.5           stringi_1.8.4
[106] viridisLite_0.4.2  munsell_0.5.1      lazyeval_0.2.2
[109] glmnet_4.1-9       hms_1.1.3          bit64_4.5.2
[112] haven_2.5.4        rbibutils_2.3      bit_4.5.0.1
[115] readxl_1.4.3
```

Runs smoothly.