Plot Cleaning Experiment

Ian Combs – icombs@mote.org

Contents

version: December 05, 2022

GitHub repository

This is the analysis pipeline for data generated from Virtual Point Intercept annotations of our EDR plot cleaning experiment outplant sites as part of the Mission: Iconic Reefs funded project.

All analyses performed with R verion 4.2.1

Basic setup of R environment

Loading required packages

For the following analyses we will require the use of a number of different R packages. Most of which can be sourced from CRAN, but some must be downloaded from GitHub. We can use the following code to load in the packages and install any packages not previously installed in the R console.

```
if (!require("pacman")) install.packages("pacman")
pacman::p_load("ggplot2","officer","ggpubr", "rcompanion", "RColorBrewer", "patchwork", "magrittr","resh
pacman::p_load_gh("pmartinezarbizu/pairwiseAdonis/pairwiseAdonis")
```

Loading Data

Data is exported from Viscore's VPI scriplet as a .csv file and named according to the file. Because we have a large number of csv files (from each individual model/project) we need a way to batch upload them and separate them. Here we created a function `read_plus` to pull the file names of each file, and download each .csv and include their file name as a column into a single dataframe. Since the file names have relevant information in them (like date and site), we are splitting the column of file name using the function `separate` into relevant factors like "date" and "site" and removing the irrelevant information. This is so we don't have to go into the original .csv files and manually split and add new columns. Since Viscore puts the "percent" column as a percentage sign (%) R cannot read it, we are using `dplyr`'s `rename` function to change it to something R can read and that is easier to call and read when we are handling the data.

```
read_plus <- function(flnm) {
    read_csv(flnm) %>%
        mutate(filename = flnm)
}

vpiData1 <- list.files(path = "../data/vpiData", pattern = "*.csv", full.names = T) %>%
    map_df(~read_plus(.))


# Sort out data to have proper columns for all the info i want, this might need
# to be changed to figure out the Marks vs Plot thing

vpiData <- vpiData1 %>%
    tidyr::separate(filename, into = c("Date", "Site"), extra = "merge", sep = "(?=T)") %>%
    tidyr::separate(Date, into = c("remove", "Date"), sep = "../data/vpiData/") %>%
    tidyr::separate(Site, into = c("Site", "delete"), sep = ".pqs") %>%
    dplyr::select(-c(remove, delete)) %>%
    mutate_if(is.character, str_replace_all, pattern = "_", replacement = "-") %>%
    tidyr::separate(Date, into = c("Date", "remove"), sep = 10) %>%
    dplyr::select(-remove) %>%
    tidyr::separate(Site, into = c("Site", "Plot"), sep = "(?=plot)", extra = "merge") %>%
    tidyr::separate(Site, into = c("Site", "Marker"), sep = "(?=mark)", extra = "merge") %>%
    tidyr::separate(Plot, into = c("Plot", "Marker"), sep = "(?=mark)", extra = "merge") %>%
    dplyr::rename(abundance = `%`) %>%
    tidyr::drop_na(abundance) %>%
    arrange(desc(abundance))

# Changing the Date to factors
```

```
vpiData$Date <- as.factor(vpiData$Date)
vpiData$Date <- strptime(vpiData$Date, format = "%Y-%m-%d")  #defining what is the original format of y
vpiData$Date <- as.Date.factor(vpiData$Date, format = "%Y-%m-%d")

# Making some corrections in the naming
vpiData$class <- gsub("Crustose coralline algae natans", "Crustose coralline algae",
    vpiData$class)
vpiData$class <- gsub("Rubble", "Bare Substrate", vpiData$class)
vpiData$class <- as.factor(vpiData$class)
vpiData$Site <- as.factor(vpiData$Site)
```

Filtering Data

Only subsetting the data for the plot cleaning experiment. This will consist of sites **T-12**, **T-13**, and **T-AP-3**.

```
edr <- vpiData %>%
    dplyr::select(Date, Site, Plot, Marker, class, count) %>%
    dplyr::filter(Site %in% c("T-13a", "T-12a", "T-AP-3a", "T-13b", "T-12b", "T-AP-3b"))

head(edr)
```

```
## # A tibble: 6 x 6
##   Date       Site    Plot      Marker    class          count
##   <date>     <fct>   <chr>     <chr>     <fct>          <dbl>
## 1 2021-04-22 T-13a   plot36-40 mark10-8  Bare Substrate   514
## 2 2021-04-22 T-13a   plot36-40 mark20-16 Bare Substrate   671
## 3 2021-04-22 T-13a   plot36-40 mark11-10 Bare Substrate   616
## 4 2021-04-22 T-13a   plot31-35 mark8-10  Bare Substrate   528
## 5 2022-04-28 T-AP-3a plot5-8   mark13-37 Bare Substrate   984
## 6 2021-04-22 T-13a   plot41-45 mark12-14 Bare Substrate   252
```

More Filtering

We are filtering the data once more to exclude anything with an abundance lower than 2%. This makes our graphs much cleaner.

```
edr_2p1 <- edr %>%
    select(Date, Site, Plot, Marker, class, count) %>%
    group_by(Date, Site, class) %>%
    filter(class != "Review") %>%
    summarise(count = sum(count)) %>%
    group_by(Date, Site) %>%
    mutate(totalObservations = sum(count, na.rm = TRUE)) %>%
    group_by(Date, Site, class) %>%
    summarise(abundance = ((count/totalObservations) * 100))
```

```
## `summarise()` has grouped output by 'Date', 'Site'. You can override using the
## `.groups` argument.
## `summarise()` has grouped output by 'Date', 'Site'. You can override using the
## `.groups` argument.
```

```
edr_2p <- edr_2p1 %>%
    filter(abundance > 0.5) %>%
    droplevels()

# reordering factor levels so they're ranked by abundance levels using the
# package 'data table'
edr_2p = data.table(edr_2p)
edr_2p[, `:=`(class, reorder(class, abundance))]

edr_2p$Date <- as.factor(edr_2p$Date)
edr_2p$Date <- droplevels(edr_2p$Date)
edr_2p$Date <- as.Date.factor(edr_2p$Date)

myColors <- gg_color_hue(length(edr_2p$class), c = 50)
```

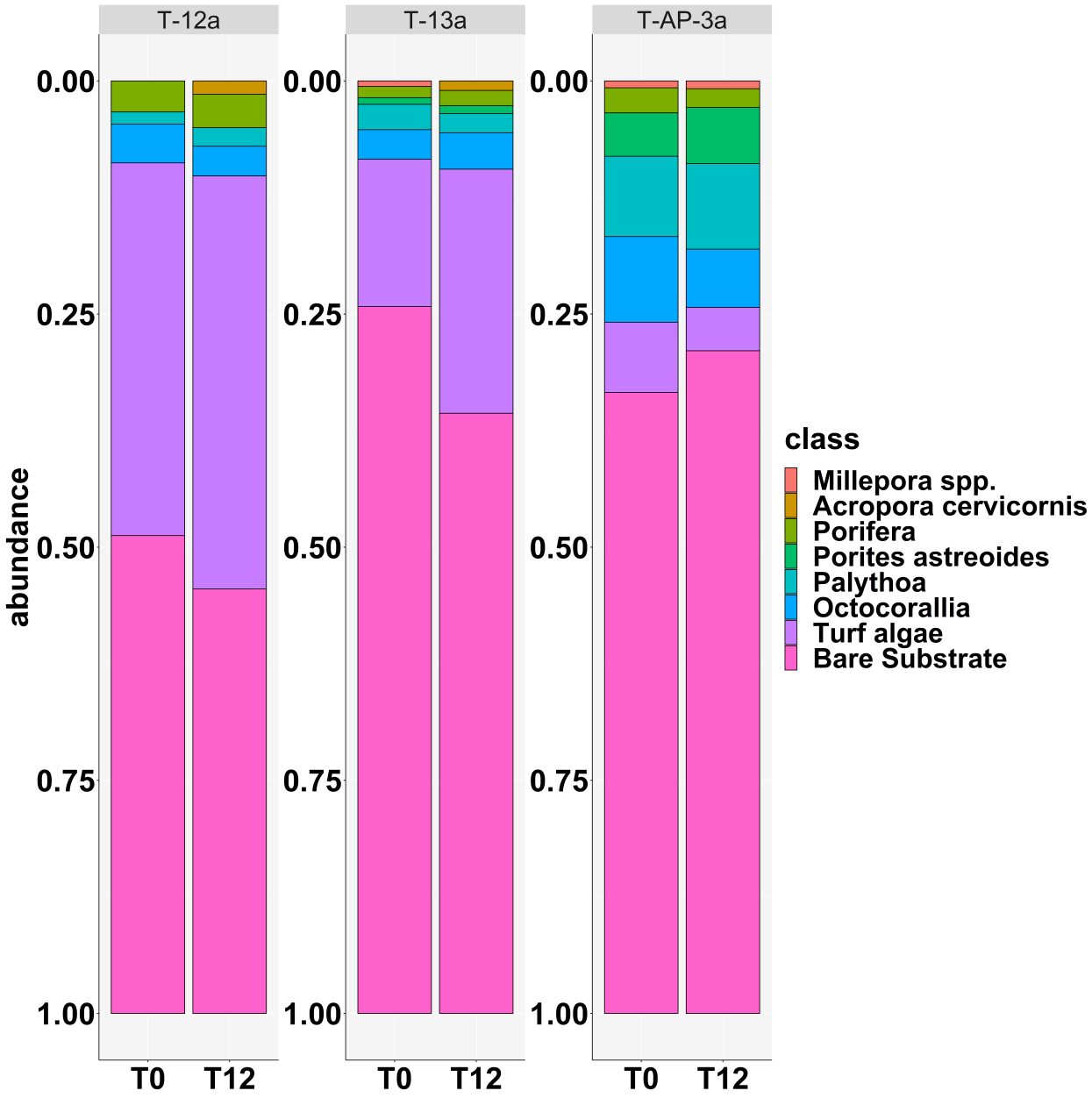
Plotting the Data

Plotting the data as a stacked bargraph.

```
edrStack1 <- ggplot(edr_2p, aes(x = as.factor(Date), y = abundance)) + geom_bar(aes(fill = class),
    position = position_fill(reverse = TRUE), stat = "identity", color = "black") +
    # scale_fill_manual(values = myColors)+
scale_x_discrete(labels = c("T0", "T12")) + scale_y_reverse() + facet_wrap(~Site,
    scales = "free")


edrStack <- edrStack1 + theme(axis.text.x = element_text(size = 40, colour = "black",
    vjust = 0.5, hjust = 0.5, face = "bold"), axis.title.x = element_blank(), axis.title.y = element_te
    face = "bold"), axis.text.y = element_text(colour = "black", size = 40, face = "bold"),
    legend.title = element_text(size = 40, face = "bold"), legend.text = element_text(size = 36,
        face = "bold", colour = "black"), panel.grid.major = element_line(size = 0.5,
        linetype = "solid", colour = "white"), panel.background = element_rect(fill = "#F5F5F5"),
    plot.title = element_text(size = 40, face = "bold"), axis.line = element_line(colour = "black"),
    axis.ticks = element_line(color = "black"), text = element_text(size = 40, color = "black"),
    legend.position = "right")
edrStack

ggsave("../figures/plotCleaningStudyGraph.png", plot = edrStack, width = 20, height = 20,
    units = "in", dpi = 600)
```


T-12a
T-13a
T-AP-3a
abundance
0.00
0.25
0.50
0.75
1.00
T0
T12
class
Millepora spp.
Acropora cervicornis
Porifera
Porites astreoides
Palythoa
Octocorallia
Turf algae
Bare Substrate