Figure3A

Emiliano Pereira

2025-08-06

Intro

Compares the functional losses when transitioning from Summer-Autumn to Winter-Spring and vice versa to reproduce the analyses and the Figure 3D from the original publication Seasonal dynamics of the coastal microbiome and its association with environmental factors.

1. Set the environment

```
library(tidyverse)
library(vegan)
source("../scripts/resources/custom_bray_curtis.R")
```

2. Load data

`opus_workable.tsv` contains the rarefied OPU abundance profiles, with samples as rows and OPUs as columns.
`date2season2community.tsv` is table mapping the date, season, and community columns.

```
ABUND <- read_tsv("../data/opus_workable.tsv.gz", show_col_types = FALSE)

DATE2SEASON2COMMUITY <- read_tsv("../data/date2season2community.tsv", show_col_types = FALSE)
```

3. Format ABUND as wide

```
ABUND <- ABUND %>%
         pivot_wider(values_fill = 0,
                     values_from = abundance,
                     names_from = opu_id) %>%
         column_to_rownames("Date")
```

4. Compute D diff, B and C components

Computes the D, B, and C components as defined in (Legendre et al. 2018)[https://onlinelibrary.wiley.com/doi/full/10.1002/ece3.4984]

```
# ABUND <- (ABUND > 0) + 0

# ABUND_dist_list <- custom_bray_curtis(mtx = ABUND,
#                                       comp_log = T,
#                                       ncores = 40)

# We are going to load a precomputed dissimilarities, becaouse this takes too long.
ABUND_dist_list <- readRDS(file = "../data/ABUND_dist22_shared_list.rds")
```

```
ABUND_D_diff_mtx <- ABUND_dist_list[["mtx"]][["D_diff"]]
ABUND_B_loss_mtx <- ABUND_dist_list[["mtx"]][["B_loss"]]
ABUND_C_gain_mtx <- ABUND_dist_list[["mtx"]][["C_gain"]]

# populate lower matrix
ABUND_D_diff_mtx[is.na(ABUND_D_diff_mtx)] <- 0
ABUND_D_diff_mtx <- ABUND_D_diff_mtx + t(ABUND_D_diff_mtx)

ABUND_B_loss_mtx[is.na(ABUND_B_loss_mtx)] <- 0
ABUND_B_loss_mtx <- ABUND_B_loss_mtx + t(ABUND_B_loss_mtx)

ABUND_C_gain_mtx[is.na(ABUND_C_gain_mtx)] <- 0
ABUND_C_gain_mtx <- ABUND_C_gain_mtx + t(ABUND_C_gain_mtx)
```

5. Format all dist tables to long

```
ABUND_D_diff_long <- ABUND_D_diff_mtx %>%
                     as.data.frame() %>%
                     rownames_to_column("Date") %>%
                     pivot_longer(names_to = "Date_vs",
                                  values_to = "D_diff",
                                  cols = 2:(dim(ABUND_D_diff_mtx)[2] +1)) %>%
                     filter(is.na(D_diff) != T) %>%
                     mutate(Date_formatted = as.Date(Date),
                            Date_formatted_vs = as.Date(Date_vs))

ABUND_B_loss_long <- ABUND_B_loss_mtx %>%
                     as.data.frame() %>%
                     rownames_to_column("Date") %>%
                     pivot_longer(names_to = "Date_vs",
                                  values_to = "B_loss",
                                  cols = 2:(dim(ABUND_B_loss_mtx)[2] +1)) %>%
                     filter(is.na(B_loss) != T) %>%
                     mutate(Date_formatted = as.Date(Date),
                            Date_formatted_vs = as.Date(Date_vs))

ABUND_C_gain_long <- ABUND_C_gain_mtx %>%
                     as.data.frame() %>%
                     rownames_to_column("Date") %>%
                     pivot_longer(names_to = "Date_vs",
                                  values_to = "C_gain",
                                  cols = 2:(dim(ABUND_C_gain_mtx)[2] +1)) %>%
                     filter(is.na(C_gain) != T) %>%
                     mutate(Date_formatted = as.Date(Date),
                            Date_formatted_vs = as.Date(Date_vs))
```

6. Merge tables

```
ABUND_dist_long_ext <- left_join(x = ABUND_D_diff_long,
                                 y = ABUND_B_loss_long,
                                 by = c("Date","Date_vs")) %>%
                       left_join(x = .,
```

```
                                  y = ABUND_C_gain_long,
                                  by = c("Date","Date_vs")) %>%
                       mutate(days = abs(as.numeric((difftime(Date_vs, Date, units = "days"))))) %>%
                       mutate(BC_value = abs(B_loss - C_gain),
                              gains = if_else(Date < Date_vs & C_gain > B_loss, 1, 0),
                              gains = if_else(Date > Date_vs & C_gain < B_loss, 1, gains),
                              losses = if_else(gains == 1, 0, 1)) %>%
                      left_join(x = ., y = DATE2SEASON2COMMUITY %>%
                                           mutate(Date = as.character(Date)) %>%
                                           select(Date, Community),
                                by = "Date") %>%
                      left_join(x = ., y = DATE2SEASON2COMMUITY %>%
                                          mutate(Date = as.character(Date)) %>%
                                          select(Date, Community),
                                by = c("Date_vs" = "Date"), suffix = c("", "_vs")) %>%
                      mutate(s_comparison = paste(Community,
                                                  Community_vs,
                                                  sep = " vs "),
                             s_comparison =  factor(s_comparison,
                                                    levels = c("S1 vs S2", "S2 vs S1",
                                                               "S1 vs S1", "S2 vs S2")))  %>%
                     filter(Date != Date_vs)
```

7. Plot losses bar plot

```
text_size  <- 12

ABUND_dist_long_ext$s_comparison %>% table()
```

```
## .
## S1 vs S2 S2 vs S1 S1 vs S1 S2 vs S2
##      117      117       72      156
```

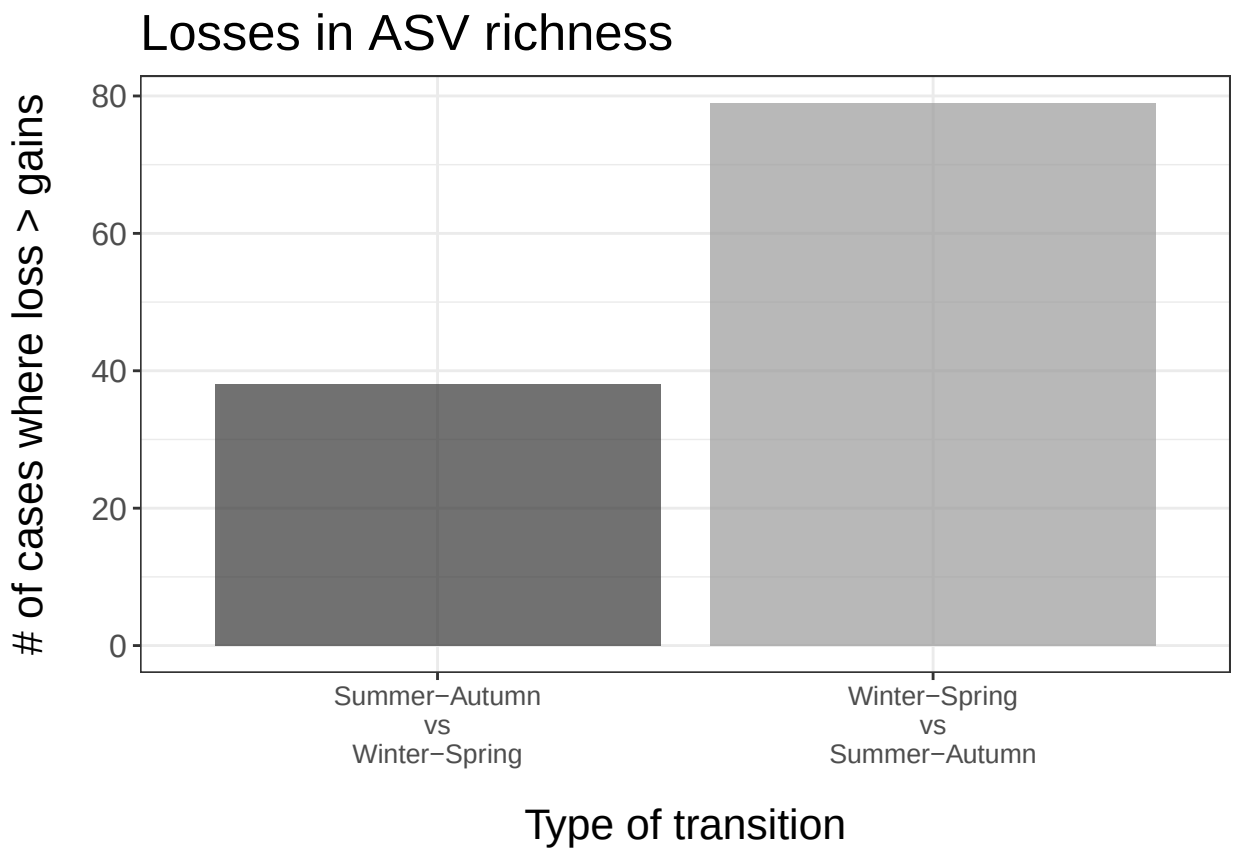
```
# gains_counts <- ABUND_dist_long_ext %>%
#   filter(s_comparison == "S1 vs S2" | s_comparison == "S2 vs S1") %>%
#   group_by(s_comparison) %>%
#   summarise(gains = sum(gains))

losses_counts <- ABUND_dist_long_ext %>%
#   filter(Date_formatted < Date_formatted_vs) %>%
  filter(s_comparison == "S1 vs S2" | s_comparison == "S2 vs S1") %>%
  group_by(s_comparison) %>%
  summarise(losses = sum(losses))

barplot_losses <-  losses_counts %>%
  ggplot(mapping = aes(x = s_comparison, y = losses, fill = s_comparison)) +
  geom_bar(alpha = 0.7, linewidth = 3, stat = "identity") +
  scale_x_discrete(labels = c("S1 vs S2"= "Summer-Autumn\nvs\nWinter-Spring",
                              "S2 vs S1" = "Winter-Spring\nvs\nSummer-Autumn")) +
  # scale_fill_manual(values =  c("#FF6347", "#237230")) +
  scale_fill_manual(values =  c("gray20", "gray60")) +
  ggtitle("Losses in ASV richness") +
  theme_bw() +
  xlab("Type of transition") +
```

```
  ylab("# of cases where loss > gains") +
  theme(
    # axis.text.x = element_text(angle = 45, hjust = 1, size = text_size),
    axis.text.x = element_text(size = text_size -2),
    axis.text.y = element_text(size = text_size),
    axis.title.x = element_text(size =  text_size +4, margin = margin(t = 15)),
    axis.title.y = element_text(size =  text_size +4, margin = margin(r = 15)),
    strip.text = element_text(size = text_size+4),
    title = element_text(size = text_size + 4, hjust = 0.5, margin = margin(b = 15)),
    strip.background = element_blank()
  ) +
  guides(fill = "none")

barplot_losses
```



8. Print session info

```
sessionInfo()
```

```
## R version 4.4.2 (2024-10-31)
## Platform: x86_64-pc-linux-gnu
## Running under: Ubuntu 20.04.6 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.9.0
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.9.0
```

```
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C               LC_TIME=en_US.UTF-8        LC_COLLATE=en_U
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8    LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C             LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATI
##
## time zone: Etc/UTC
## tzcode source: system (glibc)
##
## attached base packages:
## [1] parallel  stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] doParallel_1.0.17 iterators_1.0.14  foreach_1.5.2     vegan_2.6-8       lattice_0.22-6    permu
##  [7] lubridate_1.9.3   forcats_1.0.0     stringr_1.5.1     dplyr_1.1.4       purrr_1.0.2       readr_
## [13] tidyr_1.3.1       tibble_3.2.1      ggplot2_3.5.1     tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] utf8_1.2.4        generics_0.1.3    stringi_1.8.4     digest_0.6.37     hms_1.1.3         magrit
##  [7] evaluate_1.0.1    grid_4.4.2        timechange_0.3.0  fastmap_1.2.0     Matrix_1.7-0      tinyte
## [13] mgcv_1.9-1        fansi_1.0.6       scales_1.3.0      codetools_0.2-20  cli_3.6.3         rlang_
## [19] crayon_1.5.3      bit64_4.5.2       munsell_0.5.1     splines_4.4.2     yaml_2.3.10       withr_
## [25] tools_4.4.2       tzdb_0.4.0        colorspace_2.1-1  vctrs_0.6.5       R6_2.5.1          lifecy
## [31] bit_4.5.0         vroom_1.6.5       MASS_7.3-61       cluster_2.1.6     pkgconfig_2.0.3   pillar
## [37] gtable_0.3.5      glue_1.8.0        highr_0.11        xfun_0.48         tidyselect_1.2.1  rstudi
## [43] knitr_1.48        farver_2.1.2      htmltools_0.5.8.1 nlme_3.1-166      labeling_0.4.3    rmarko
## [49] compiler_4.4.2
```