Figure2

Emiliano Pereira

2025-08-06

Intro

Space-time analysis comparing SAMO and TARA Oceans samples, representing microbial communities from the Brazil and Malvinas ocean currents, to reproduce the analysis shown in Figure 2 of the original publication Seasonal dynamics of the coastal microbiome and its association with environmental factors.

1. Set the environment

```
library(tidyverse)
library(vegan)
library(maps)
library(fANCOVA)
```

2. Load data

```
ABUND_workable_long <- read_tsv("../data/samo_vs_tara_workable.tsv")
```

```
## Rows: 11470 Columns: 4
## -- Column specification ------------------------------------------------------------------------
## Delimiter: "\t"
## chr (3): sample_name, tax, dataset
## dbl (1): abund
##
## 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.
```

```
SAMO_METADATA <- read_tsv("../data/samo_metadata_workable.tsv")
```

```
## Rows: 26 Columns: 15
## -- Column specification ------------------------------------------------------------------------
## Delimiter: "\t"
## chr   (2): sample_name, Season
## dbl  (12): Z_m, Secchi_m, Temperature_C, pH, k_mS_cm_1, Turbidity_NTU, DO, Percent_DO, TDS, Salinity
## date  (1): Date
##
## 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.
```

```
TARA_METADATA <- read_tsv("../data/tara_metadata_workable.tsv")
```

```
## Rows: 139 Columns: 55
## -- Column specification ------------------------------------------------------------------------
## Delimiter: "\t"
## chr  (14): pangea_id, Mean_Date [YY/MM/DD hh:mm]*, sample, INSDC sample accession number(s), INSDC r
```

```
## dbl  (40): Mean_Lat*, Mean_Long*, Mean_Depth [m]*, Mean_Temperature [deg C]*, Mean_Salinity [PSU]*, I
## dttm  (1): Date/Time [yyyy-mm-ddThh:mm]
##
## 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.
```

3. Split datasets and convert to wide format

This section selects SAMO and TARA sample IDs of interest and converts the long-format abundance table into a wide matrix (samples × taxa). It then subsets the matrix into SAMO and TARA blocks and reports their dimensions.

```
samo_samples <- SAMO_METADATA$sample_name

tara_samples <- TARA_METADATA %>%
  filter(region == "(SAO) South Atlantic Ocean [MRGID:1914]" | region == "(SO) Southern Ocean [MRGID:19
  filter(level == "SRF") %>%
  filter(lat < 0 & lon < -5) %>%
  pull(sample)

ABUND_workable <- ABUND_workable_long %>%
                  select(-dataset) %>%
                  pivot_wider(names_from = tax, values_from = abund, values_fill = 0) %>%
                  column_to_rownames("sample_name")

ABUND_workable_samo <- ABUND_workable[samo_samples,]
ABUND_workable_tara <- ABUND_workable[tara_samples,]

dim(ABUND_workable_samo)
```

```
## [1]  26 718
```

```
dim(ABUND_workable_tara)
```

```
## [1]   6 718
```

4. Compute dissimilarity

This section computes Bray–Curtis dissimilarity for every SAMO–TARA sample pair and returns a tidy data frame with one row per comparison.

```
samo_vs_tara_diss <- list()

for (i in samo_samples) {
  samo_sample_i <- ABUND_workable_samo[i,]

  for (j in tara_samples) {
    tara_sample_j <- ABUND_workable_tara[j,]

    diss <- vegdist(rbind(samo_sample_i, tara_sample_j), method = "bray")
    samo_vs_tara_diss[[paste(i, j, sep = "|")]] <- as.numeric(diss)
  }
}

samo_vs_tara_diss_df <- samo_vs_tara_diss %>%
                        do.call(rbind, .) %>%
                        as.data.frame %>%
```

```
                        rownames_to_column(var = "comparison") %>%
                        mutate(samo_sample = str_remove(comparison, "\\|.*"),
                               tara_sample = str_remove(comparison, ".*\\|")) %>%
                        select(-comparison) %>%
                        rename(diss = V1)
```

5. Add metadata

Simply add the metadata to the dissimilarity table `samo_vs_tara_diss_df`.

```
samo_vs_tara_diss_df_ext <- samo_vs_tara_diss_df %>%
  left_join(SAMO_METADATA %>%
  select(Date, sample_name, Season, Temperature_C),
         by = c("samo_sample" = "sample_name")) %>%
  left_join(TARA_METADATA %>% select(sample, lat, lon, region),
         by = c("tara_sample" = "sample"))
```

6. Compute similarity-weighted coordinates

This section converts dissimilarities to similarities, normalizes them per SAMO sample, and computes similarity-weighted average latitude and longitude for the matched TARA locations. It also estimates a north–south asymmetry for context.

```
scale_lat <- 3
scale_lon <- 1

samo_vs_tara_diss_df_ext_wa <- samo_vs_tara_diss_df_ext %>%
  mutate(sim = 1 -diss) %>%
  group_by(samo_sample, Date, Season, Temperature_C) %>%
  mutate(sim_norm = sim/sum(sim)) %>%
  summarize(wlat = sum(sim_norm*(-lat)^(1/scale_lat)),
            wlon = sum(sim_norm*(-lon)^(1/scale_lon))) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'samo_sample', 'Date', 'Season'. You can override using the `.gr
```

```
X_north <- TARA_METADATA %>%
  filter(sample %in% tara_samples) %>%
  filter(`Mean_Lat*` > -34.71) %>%
  mutate(dist2samo = abs(34.71 - abs(`Mean_Lat*`))) %>%
  pull(dist2samo) %>%
  mean()


X_south <- TARA_METADATA %>%
  filter(sample %in% tara_samples) %>%
  filter(`Mean_Lat*` < -34.71) %>%
  mutate(dist2samo = abs(34.71 - abs(`Mean_Lat*`))) %>%
  pull(dist2samo) %>%
  mean()


lat_diff <- X_south -X_north
```

7. Fit a LOESS function

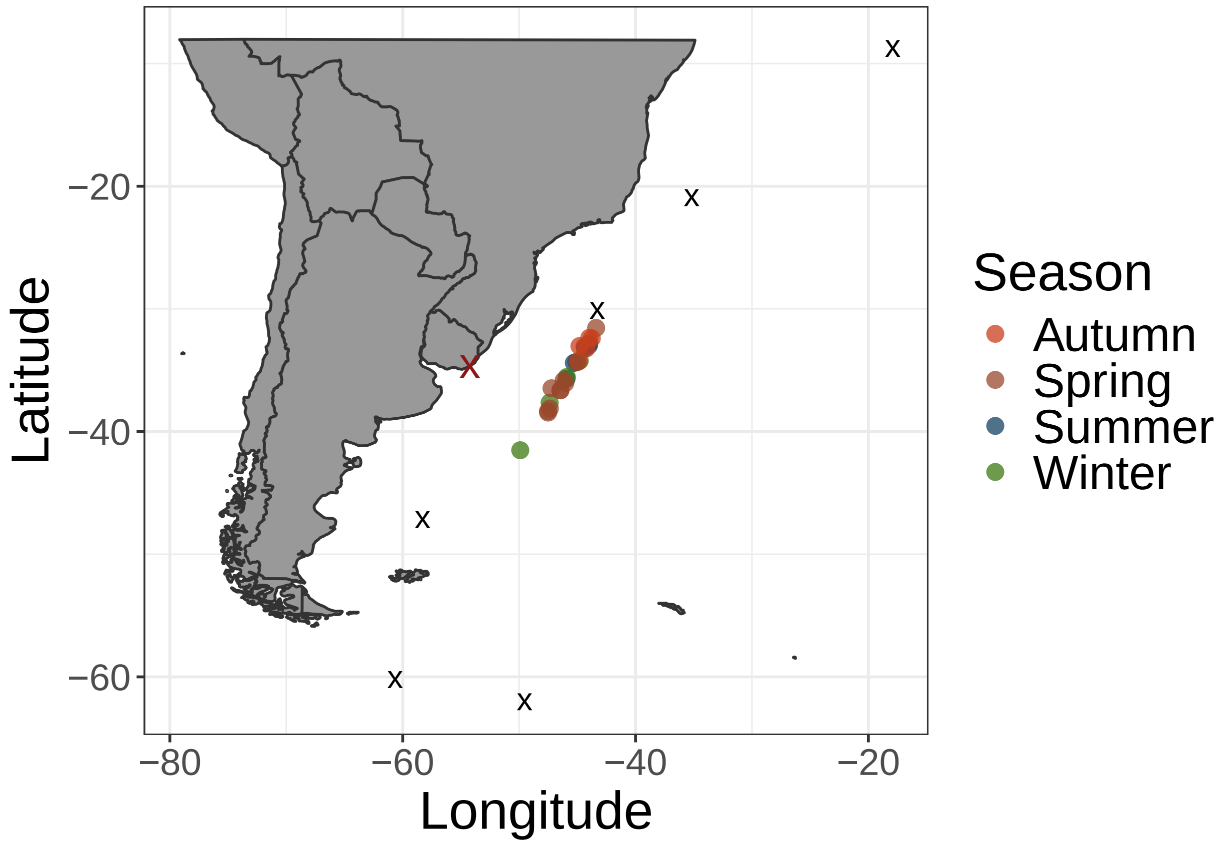
This section encodes dates as ordered numeric levels and selects an optimal LOESS span (via AICc) to smooth the time series of weighted latitude.

```
samo_vs_tara_diss_df_ext_wa$Date_levels <- as.numeric(as.factor(samo_vs_tara_diss_df_ext_wa$Date))

d_diff_loess <- loess.as(x = samo_vs_tara_diss_df_ext_wa$Date_levels,
                         y = -(samo_vs_tara_diss_df_ext_wa$wlat^scale_lat), degree = 2,
                         criterion = c("aicc", "gcv")[1], user.span = NULL, plot = F)


samo_vs_tara_diss_df_ext_wa$fit <-  predict(d_diff_loess,
                                            data.frame(days=samo_vs_tara_diss_df_ext_wa$Date_levels))
```

8. Create map

```
text_size <- 18
season_colors <- c("#c93f1b","#98482b", "#154360", "#3c7810")
world_map <- map_data("world")

# Define South American countries
south_america_countries <- c("Argentina", "Bolivia", "Brazil", "Chile",
                             "Colombia", "Ecuador", "Guyana", "Paraguay",
                             "Peru", "Suriname", "Uruguay", "Venezuela",
                             "French Guiana")

# Filter the map data for South America
south_america_map <- world_map %>% # filter(region %in% south_america_countries) %>%
                     filter(long > -80 & long < -10 & lat > -60 & lat < -8)

# Create the base map
base_map <- ggplot() +
  geom_polygon(data = south_america_map, aes(x = long, y = lat, group = group),
               fill = "gray60", color = "gray20") +
  theme_bw() +
  ylab("Longitude") +
  ylab("Latitude")

samo_coords <- data.frame(lat = -34.712056, lon = -54.235722)

# Add the points
map_with_points <- base_map +
  geom_point(data = samo_vs_tara_diss_df_ext_wa,
             aes(y = -(wlat^scale_lat), x = -(wlon)^scale_lon, color = Season), size = 2.5, alpha = 0.75
  scale_color_manual(values = c(season_colors)) +
  geom_point(data = samo_coords, aes(x = lon, y = lat), color = "#8d1616", size = 3, shape = "x", stroke
  # add dashed line for SAMO
 # geom_hline(yintercept = samo_coords$lat) +
  theme_bw() +
  labs(x = "Longitude", y = "Latitude") +
  geom_point(data = TARA_METADATA %>% filter(sample %in% tara_samples),
              aes(x = lon , y = lat), color = "black", size = 3, shape = "x", stroke = 2) +
  theme(
```

```r
    axis.text.x = element_text(size = text_size -4),
    axis.text.y = element_text(size = text_size -4),
    axis.title.x = element_text(size = text_size +2),
    axis.title.y = element_text(size = text_size +2),
    legend.text = element_text(size = text_size),
    legend.title = element_text(size = text_size +2)
  )

map_with_points
```

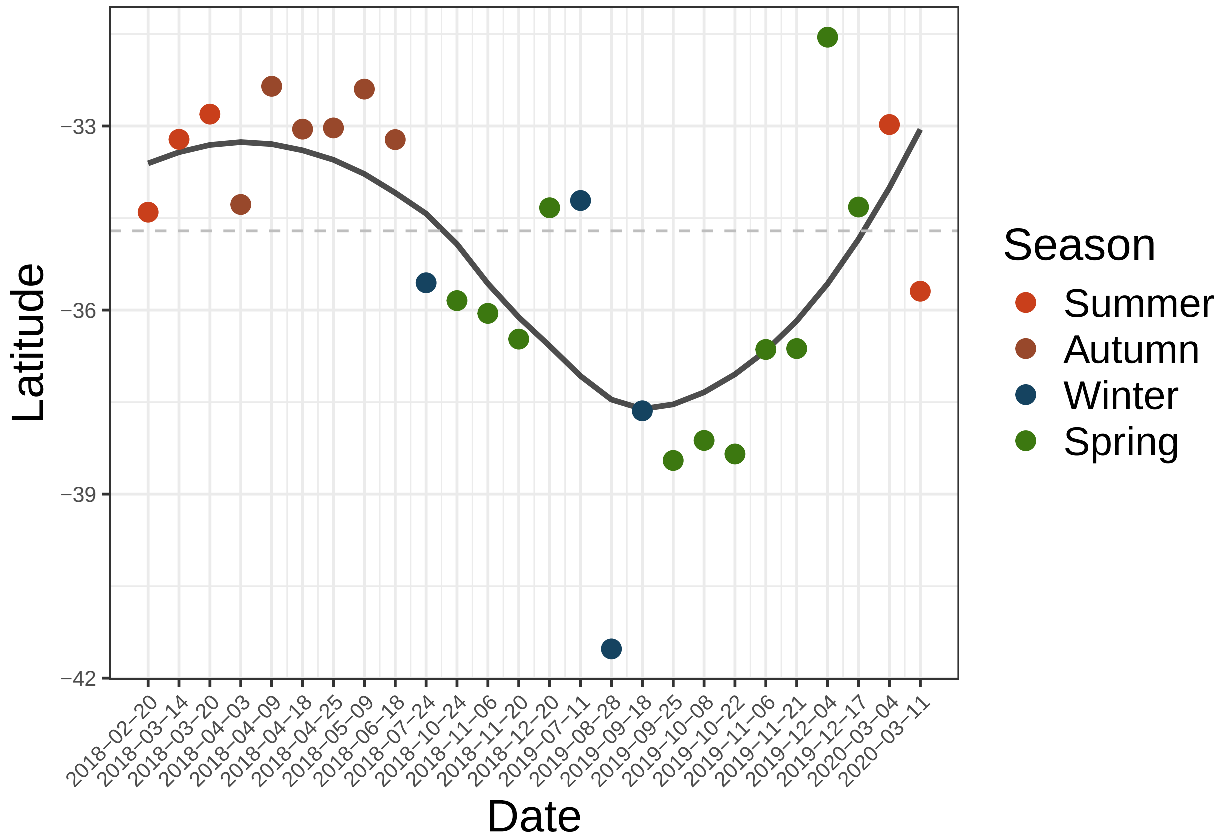


9. Plot date vs latitude

This section plots (i) the LOESS-smoothed trajectory of weighted latitude versus time and (ii) the individual points colored by season. It formats axes and draws a dashed line at the SAMO latitude.

```r
samo_vs_tara_diss_df_ext_wa$Date <- as.factor(samo_vs_tara_diss_df_ext_wa$Date)
samo_vs_tara_diss_df_ext_wa$Season <- factor(samo_vs_tara_diss_df_ext_wa$Season,
                                             levels = c("Summer", "Autumn", "Winter", "Spring" ))

text_size <- 15

wlat_plot <- samo_vs_tara_diss_df_ext_wa %>%
  ggplot(aes(x = Date_levels, y = -wlat^scale_lat, color = Season)) +
  geom_line(aes(x = Date_levels, y = fit), color = "black", linewidth = 1, alpha = 0.7) +
  geom_point(size = 3) +
  scale_x_continuous(breaks = samo_vs_tara_diss_df_ext_wa$Date_levels,
```

```
                             labels = samo_vs_tara_diss_df_ext_wa$Date) +
  geom_hline(yintercept =-34.71,
             linetype = "dashed", color = "grey") +
  theme_bw() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1, size = text_size-7),
    axis.text.y = element_text(size = text_size -7),
    axis.title.x = element_text(size = text_size +2),
    axis.title.y = element_text(size = text_size +2),
    legend.text = element_text(size = text_size),
    legend.title = element_text(size = text_size +2)
  ) +
  scale_color_manual(values = season_colors) +
  labs(x = "Date", y = "Latitude", color = "Season")

wlat_plot
```



10. 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] grid      stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] fANCOVA_0.6-1      maps_3.4.2         corrgram_1.14      ggpubr_0.6.0       ggcorrplot_0.1.4.1
##  [7] lattice_0.22-6     permute_0.9-7      lubridate_1.9.3    forcats_1.0.0      stringr_1.5.1
## [13] purrr_1.0.2        readr_2.1.5        tidyr_1.3.1        tibble_3.2.1       ggplot2_3.5.1
##
## loaded via a namespace (and not attached):
##  [1] gtable_0.3.5       xfun_0.48          rstatix_0.7.2      tzdb_0.4.0         vctrs_0.6.5        tools
##  [7] generics_0.1.3     parallel_4.4.2     fansi_1.0.6        highr_0.11         cluster_2.1.6      pkgco
## [13] Matrix_1.7-0       lifecycle_1.0.4    compiler_4.4.2     farver_2.1.2       tinytex_0.53       munse
## [19] carData_3.0-5      htmltools_0.5.8.1  yaml_2.3.10        Formula_1.2-5      pillar_1.9.0       car_3
## [25] crayon_1.5.3       MASS_7.3-61        abind_1.4-8        nlme_3.1-166       digest_0.6.37      tidys
## [31] stringi_1.8.4      labeling_0.4.3     splines_4.4.2      fastmap_1.2.0      cowplot_1.1.3      color
## [37] cli_3.6.3          magrittr_2.0.3     utf8_1.2.4         broom_1.0.7        withr_3.0.1        scale
## [43] backports_1.5.0    bit64_4.5.2        timechange_0.3.0   rmarkdown_2.28     bit_4.5.0          ggsig
## [49] hms_1.1.3          evaluate_1.0.1     knitr_1.48         mgcv_1.9-1         rlang_1.1.4        glue_
## [55] rstudioapi_0.16.0  vroom_1.6.5        R6_2.5.1
```