

Community ecology and multivariate analyses

Ramiro Logares (ICM-CSIC, Barcelona)



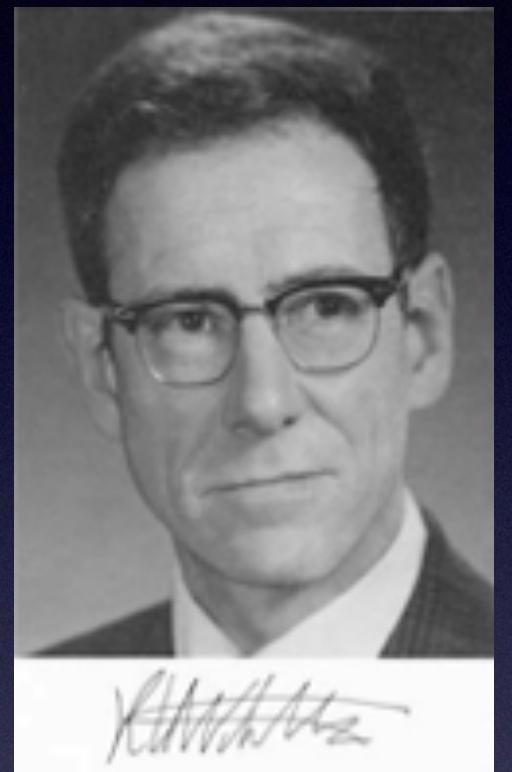
Metabarcoding projects

- 1. Sampling and wet-lab
- 2. Sequence processing
- 3. Ecological analyses : what questions do we want to answer?

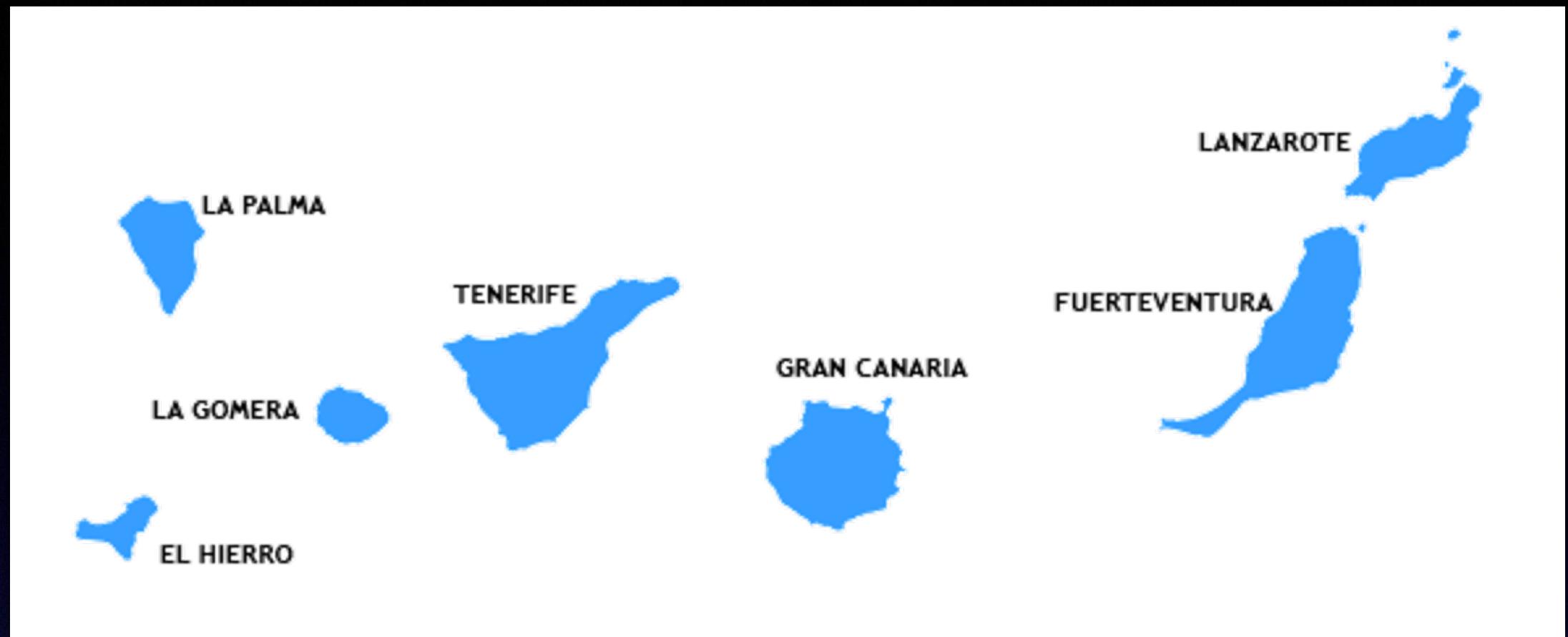
- Diversity analyses

Diversity

- Alpha
 - Richness: number of species in a location/sample
 - Evenness: relative species abundance in a location/sample
- Beta
 - Species turnover across locations/timepoints/samples
- Gamma
 - Species in all analysed locations/samples



Robert Whittaker

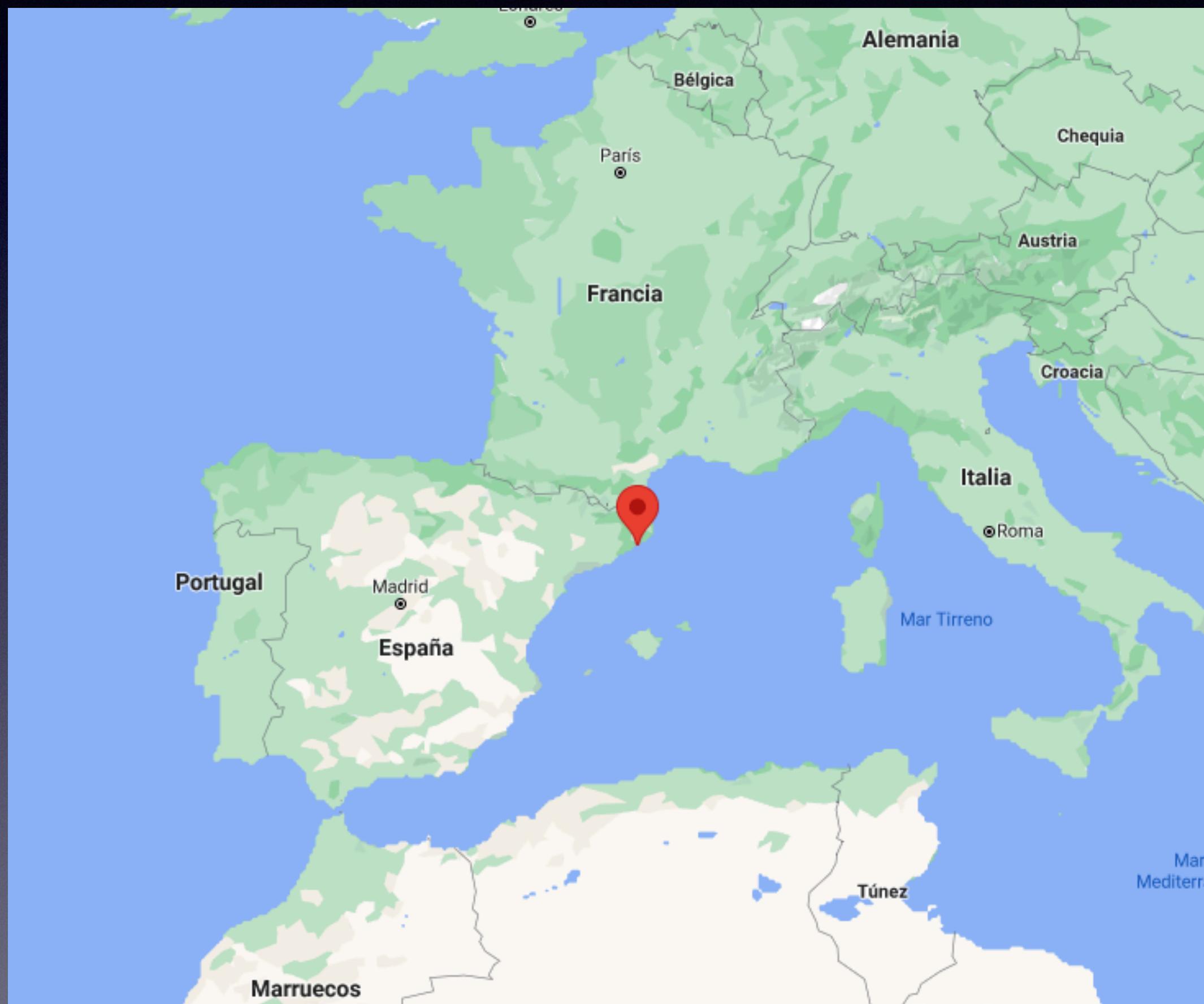


- Alpha diversity: number of species in each island
- Beta diversity: species change between islands
- Gamma diversity: species in all islands

Toy dataset

- Samples of the marine microbiome
 - Blanes Bay Microbial Observatory
 - Community 18S rRNA gene
 - 8 samples
 - January, April, July & October of 2004 and 2005

Blanes Bay Microbial Observatory



```
1 #####  
2 ## Community ecology  
3 #####  
4  
5 # Install packages (in case you didn't before)  
6  
7 install.packages("vegan")      # Community ecology functions  
8 library(vegan)  
9  
  
10 # Read dada2 output  
11  
12 otu.tab<-read_tsv("https://raw.githubusercontent.com/krabberod/BIO9905MERG1_v21/main/Dada2_Pipeline/  
13 dada2_results/OTU_table.tsv")  
14 head(otu.tab)  
15 names(otu.tab)  
  
16 dim(otu.tab) # 2107    26
```

```
18 #Let's reorder the table
19 otu.tab<-otu.tab[,c(17,19:26,1:16,18)]
20
21 #We assign to rownames the OTU names
22
23 otu.tab <- column_to_rownames(otu.tab, var = "OTUNumber") # %>% as_tibble()
24
25 rownames(otu.tab)
26 dim(otu.tab) # 2107    25
27
28 otu.tab.simple<-otu.tab[,1:8] # We'll need this table for community ecology analyses
29
30 #We transpose the table, as this is how Vegan likes it
31
32 otu.tab.simple<-t(otu.tab.simple)
33 otu.tab.simple[1:5,1:5]
34
35 #          OTU_00001 OTU_00002 OTU_00004 OTU_00005 OTU_00006
36 # BL040126      4996     12348     11426         0     3958
37 # BL040419      739       684       97     16605     4702
38 # BL040719      0         0       166         0     806
39 # BL041019      78        74         0     184     286
40 # BL050120    30697    12885     5417         0     3739
41
```

Alpha diversity

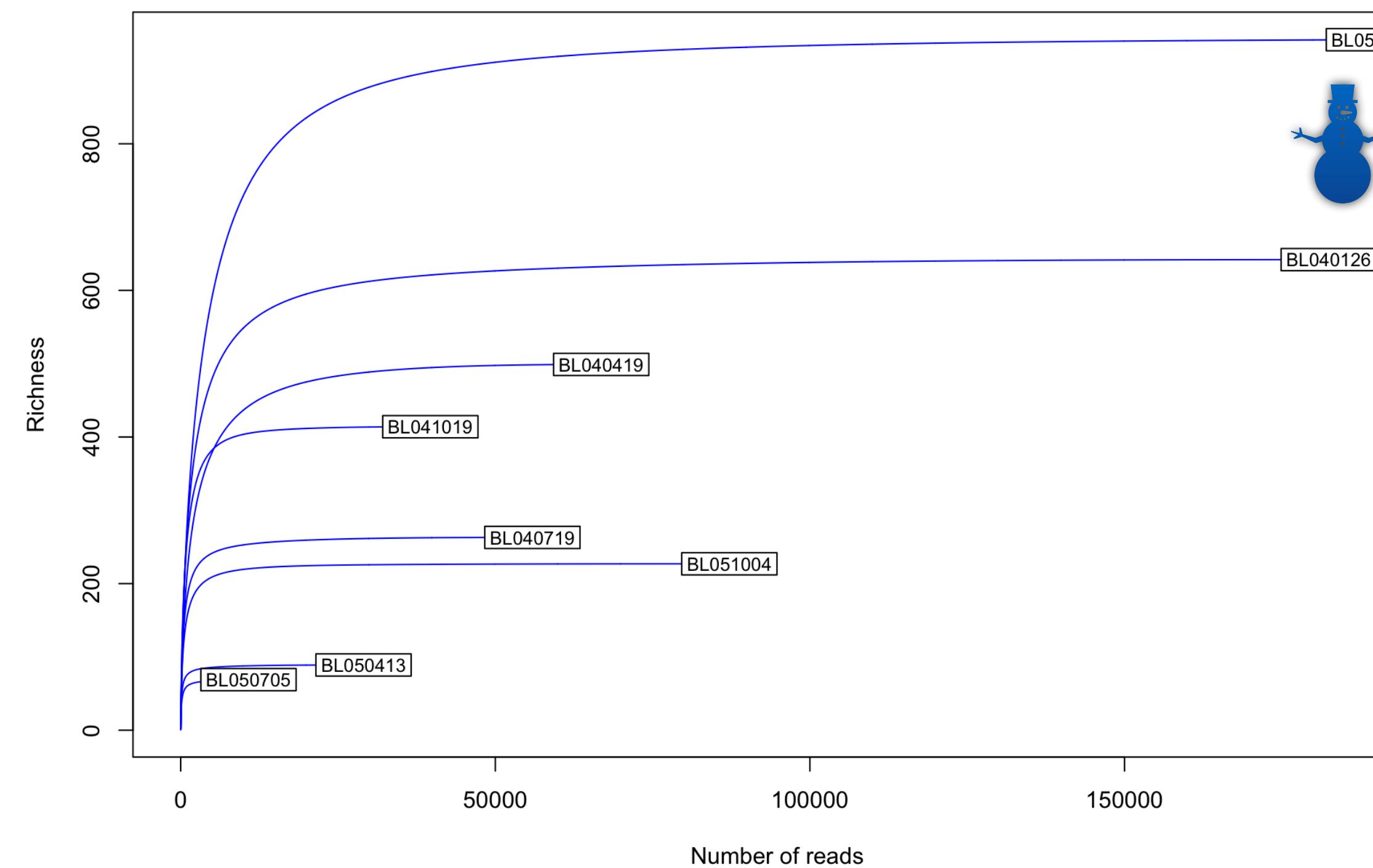
Number of species in specific samples/location

Richness estimates

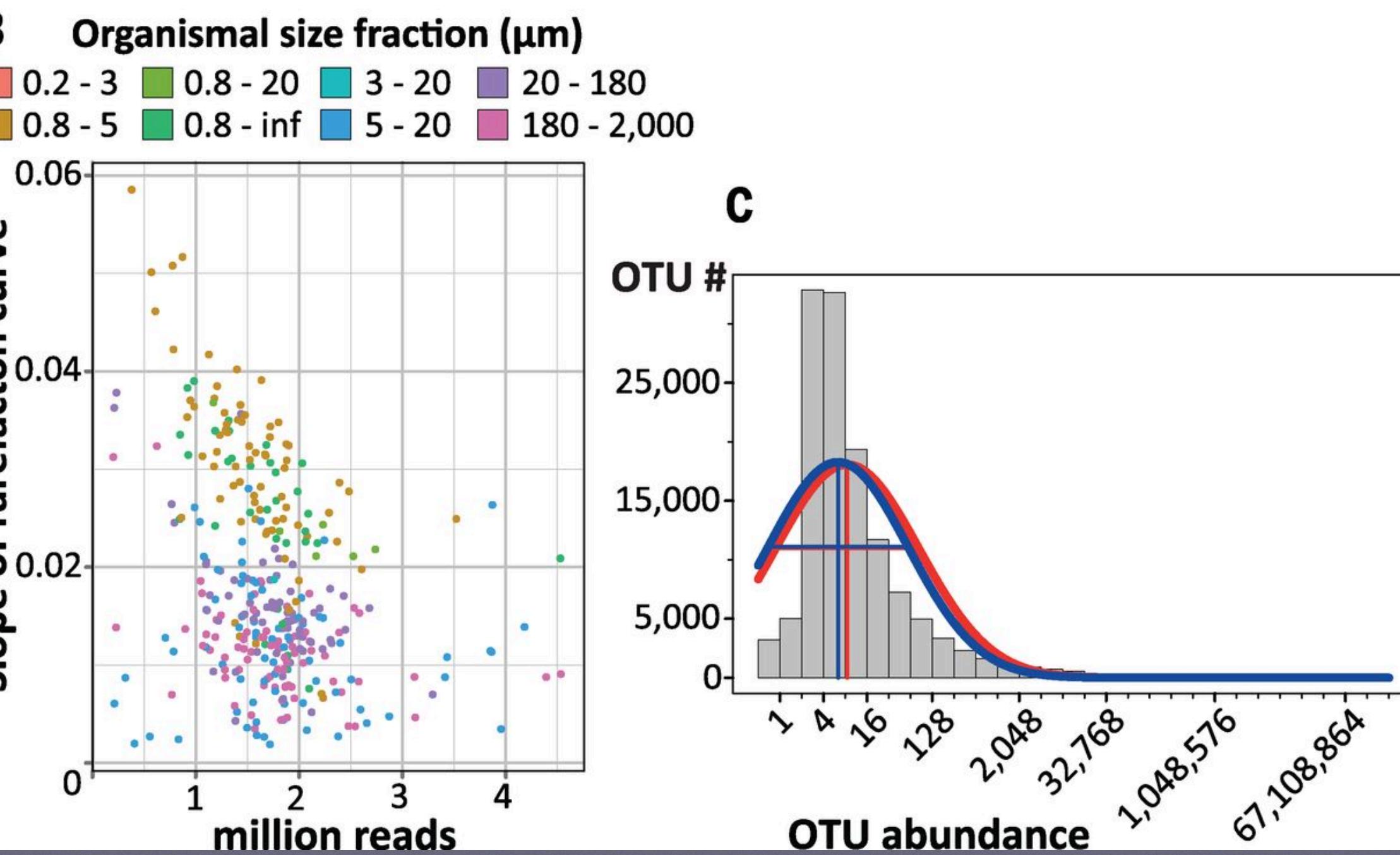
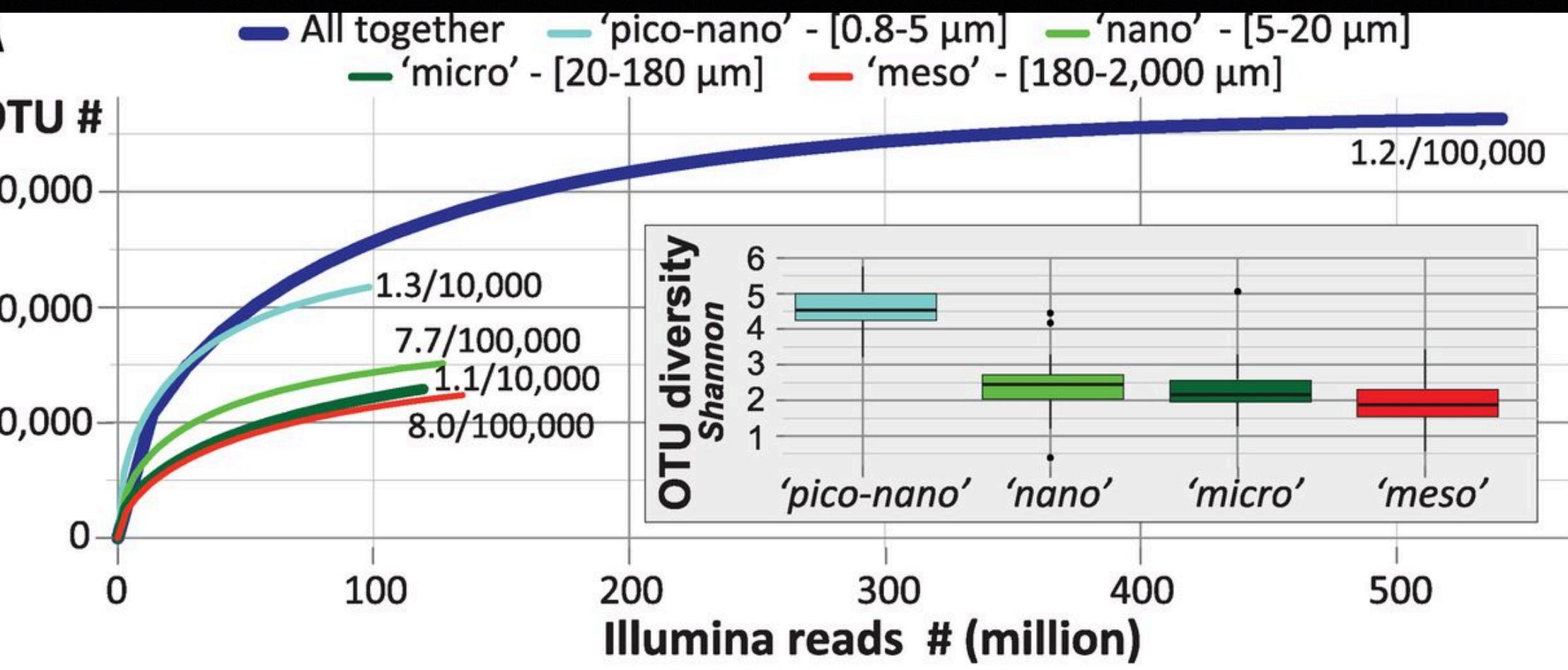
```
1 richness<-estimateR(otu.tab.simple)
2
3 # BL040126    BL040419    BL040719    BL041019    BL050120    BL050413    BL050705    BL051004
4 # S.obs       642.000000 499.000000 263.000000 414.000000 942.000000 89.000000 69.000000 227.000000
5 # S.chao1     642.000000 499.000000 263.000000 414.000000 943.250000 89.000000 69.000000 227.000000
6 # se.chao1   0.000000  0.000000  0.000000  0.000000  1.621617  0.000000  0.000000  0.000000
7 # S.ACE      642.000000 499.000000 263.000000 414.000000 943.399653 89.000000 69.000000 227.000000
8 # se.ACE     7.091415  9.573887  4.198497  6.011262  10.391615  2.539574  2.797514  2.604638
9
10 # Above are the estimators Chao and ACE and the species number.
11
```

Are we recovering all diversity?

```
1 # Rarefaction  
2  
3 #Let's calculate the number of reads per sample  
4  
5 rowSums(otu.tab.simple)  
6  
7 # BL040126 BL040419 BL040719 BL041019 BL050120 BL050413 BL050705 BL051004  
8 # 182462 66827 55896 39672 189636 29053 10771 87192  
9  
10  
11 rarecurve (otu.tab.simple, step=100, xlab= "Number of reads", ylab="Richness", col="blue")  
12
```



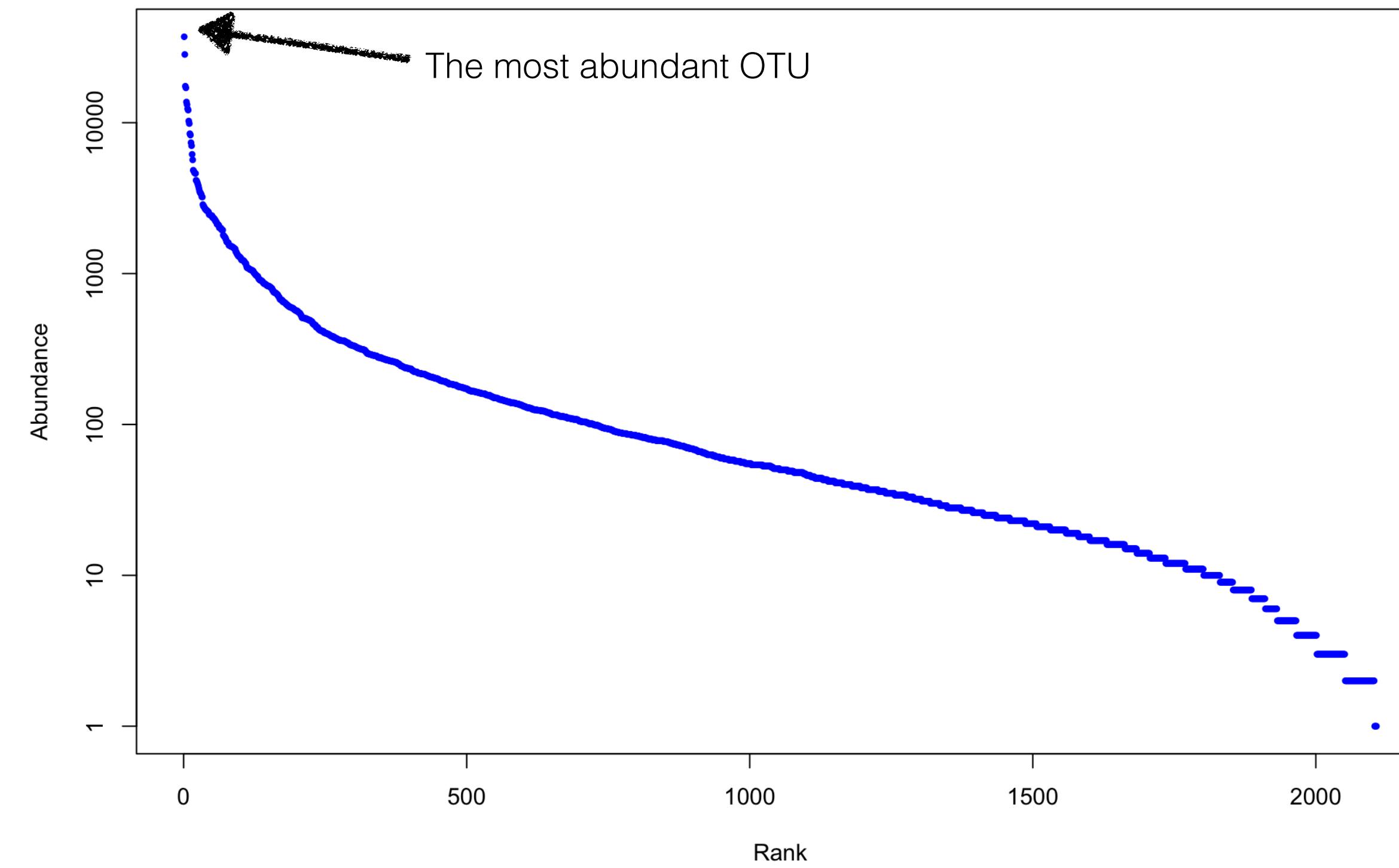
What are these results telling us?



18S V9 (Swarm)

Evenness

```
1 #Evenness  
2  
3 plot(colSums(otu.tab.simple), log="y", xlab="Rank", ylab="Abundance", pch=19, cex=0.5, col="blue")  
4  
5
```



Few species highly abundant, while most species have a low abundance

Characteristic of microbiota
Why?

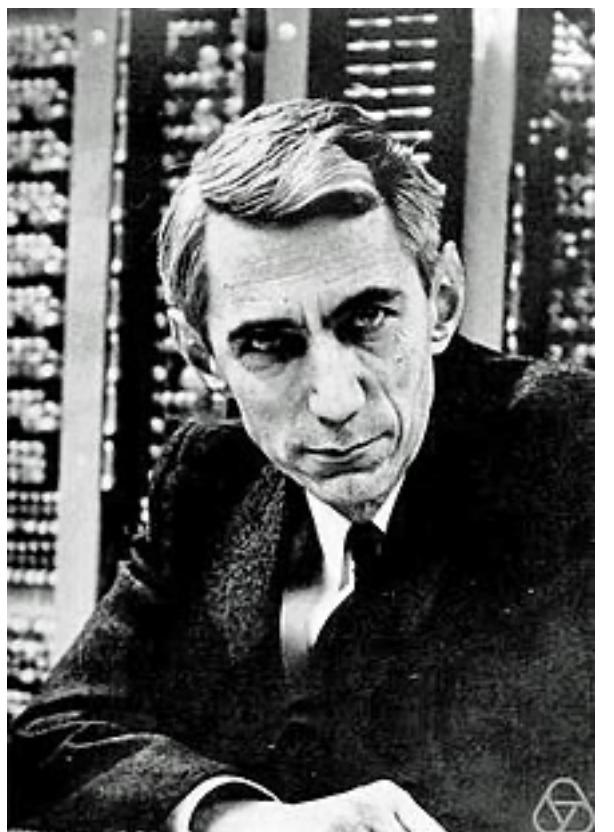
Shannon H index

- Considers richness and evenness
- Originally proposed by Claude Shannon in 1948 to quantify the entropy in strings of text.

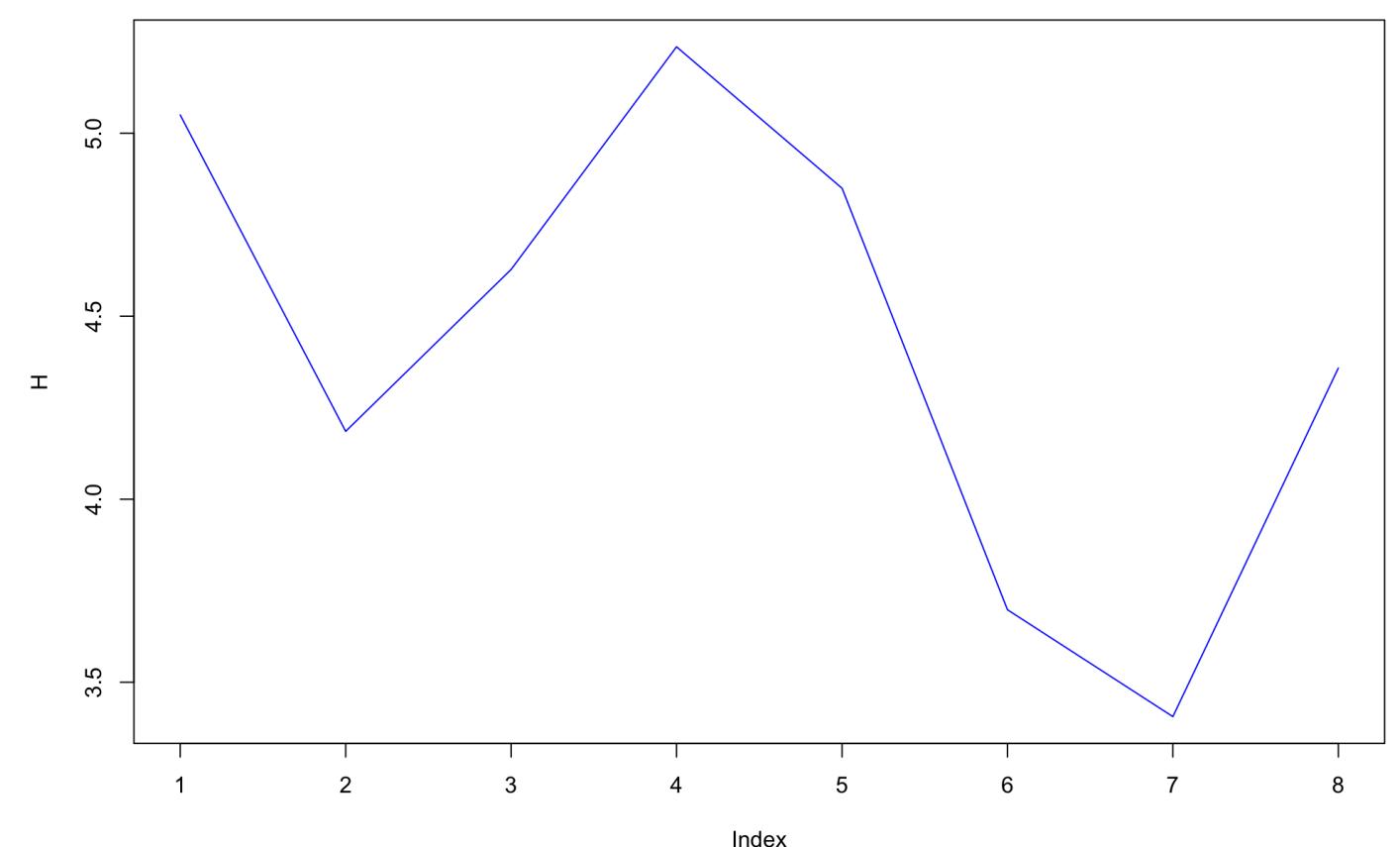
```
1 #Shannon H index (considers richness and evenness)
2
3 H<-diversity(otu.tab.simple, index="shannon")
4
5 # BL040126 BL040419 BL040719 BL041019 BL050120 BL050413 BL050705 BL051004
6 # 5.049747 4.185494 4.627698 5.236017 4.849669 3.698185 3.406164 4.358232
7
8 plot(H, type="l", col="blue")
```

$$H' = - \sum_{i=1}^R p_i \ln p_i$$

p_i is the relative abundance of the i th species



Claude Shannon



Pielou's index of evenness



$$J' = \frac{H'}{H'_{\max}}$$

E.C. Pielou

```
1 #Pielou's index of evenness (range 0-1, 1 = maximum evenness)

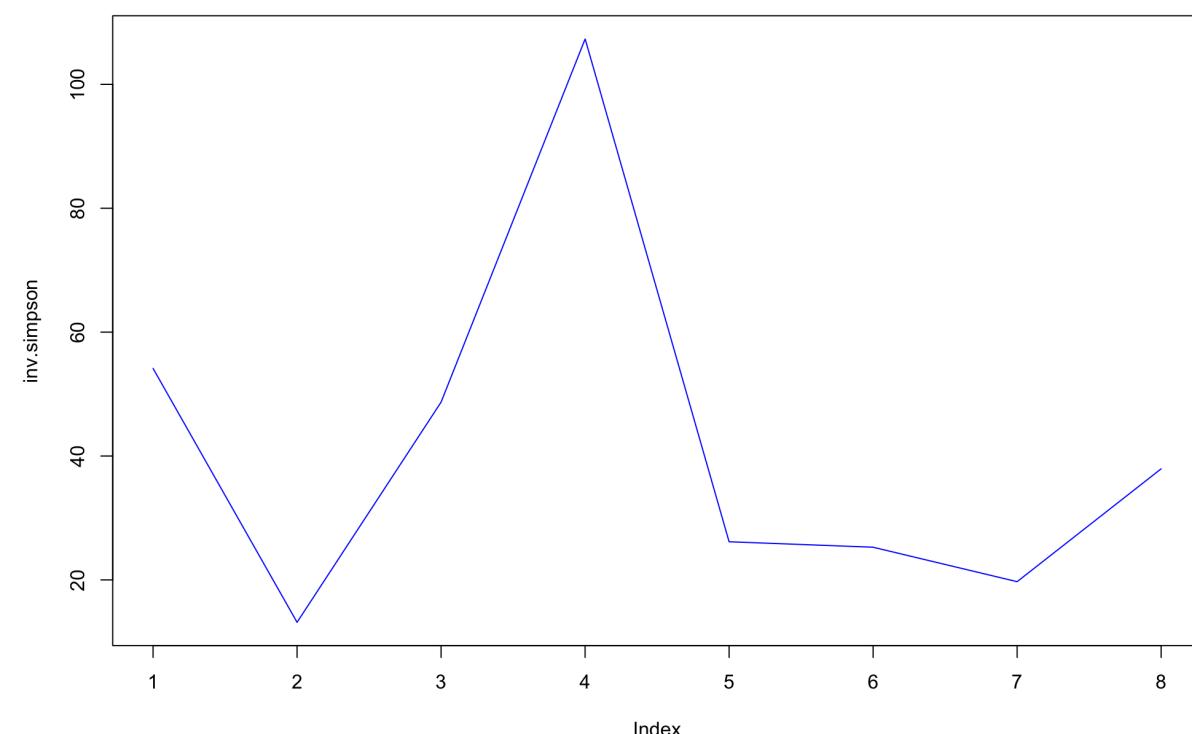
2 # J=H/Hmax
3 # J=Shannon (H) / log(S=species richness)
4
5 J=H/log(rowSums(otu.tab.simple>0))
6
7 # BL040126 BL040419 BL040719 BL041019 BL050120 BL050413 BL050705 BL051004
8 # 0.7811398 0.6737098 0.8305043 0.8689236 0.7081871 0.8238995 0.8044587 0.8033681
9
10 # Inverse Simpson's D index (richness+evenness. Larger values, larger diversity)
11
12 inv.simpson<-diversity(otu.tab.simple, "invsimpson")
13 plot(inv.simpson, type="l", col="blue")
14
15 # BL040126 BL040419 BL040719 BL041019 BL050120 BL050413 BL050705 BL051004
16 # 54.13768 13.15796 48.69382 107.30411 26.16040 25.27907 19.71550 37.93128
17
```

Inverse Simpson's D index



$$\frac{1}{\lambda} = \frac{1}{\sum_{i=1}^R p_i^2} = {}^2D$$

Edward Simpson



Beta diversity

Species turnover between samples/location

- Beta diversity analyses will investigate how communities change over different samples (timepoints, locations, etc.)
- Analyses can be biased if different samples have different sequencing efforts

- Different sequencing depths may bias the calculation of distances for multivariate analyses
 - One way to mitigate this is to subsample or “rarefy” samples to the same sequencing depth
 - But, it has been criticised due to loss of information and precision
 - Anyways, let’s try rarefying the samples to the same sequencing depth

```
1 #We rarefy all samples to the same sequencing depth to reduce biases
2 min(rowSums(otu.tab.simple)) # We calculate the sample with the minimum
amount of reads
3 # [1] 10771
4
5 otu.tab.simple.ss<-rrarefy(otu.tab.simple, 10771) #Samples are rarefied to
10771 reads per sample
6
7 rowSums(otu.tab.simple.ss) # We check the number of reads per sample
8 # BL040126 BL040419 BL040719 BL041019 BL050120 BL050413 BL050705 BL051004
9 # 10771      10771      10771      10771      10771      10771      10771      10771
10
11 #Check the dimensions of the tables
12 dim(otu.tab.simple)
13 # [1] 8 2107
14 dim(otu.tab.simple.ss)
15 # [1] 8 2107
```

```
17 #Tables have the same size, but after removing reads, several OTUs are  
18 length(which(colSums(otu.tab.simple)==0))  
19 # [1] 0 #No OTU has an abundance sum that is 0, as expected  
20  
21 length(which(colSums(otu.tab.simple.ss)==0))  
22 # [1] 273 # A total of 273 OTUs were found in the rarefied table with cero  
abundance. Let's corroborate  
23  
24 which(colSums(otu.tab.simple.ss)==0) # Show the OTUs and the position in  
the table that have 0 abundance  
25 # A small subsample of them  
26 # OTU_00814 OTU_01076 OTU_01077 OTU_01232 OTU_01242  
27 #    772        1020        1021        1166        1176
```

```
29 otu.tab.simple[,772] # This gives the abundance of the OTU_00814 across the different samples in the  
table that is NOT subsampled  
  
30 # BL040126 BL040419 BL040719 BL041019 BL050120 BL050413 BL050705 BL051004  
31 # 0 0 0 0 88 0 0 0  
32  
  
33 otu.tab.simple.ss[,772] # This gives the abundance of the OTU_00814 across the different samples in  
the table that IS subsampled  
  
34 # BL040126 BL040419 BL040719 BL041019 BL050120 BL050413 BL050705 BL051004  
35 # 0 0 0 0 0 0 0 0  
36
```

```
37 otu.tab.simple.ss.nocero<-otu.tab.simple.ss[, -(which(colSums(otu.tab.simple.ss)==0))]  
# Removes OTUs with cero abundance  
  
38 length(which(colSums(otu.tab.simple.ss.nocero)==0)) # Check that no cero abundance OTUs are left  
39 # [1] 0 # correct  
  
40 # Let's check dimensions  
  
41 dim(otu.tab.simple.ss)  
42 # [1] 8 2107  
  
43 dim(otu.tab.simple.ss.nocero)  
44 # [1] 8 1834  
  
45 # 2107-1834 = 273, This is the number of OTUs that we expected to be removed.  
46
```

Distance metrics

- Statistical distance: distance between variables
- *Distance metrics in ecology: allow measuring the dissimilarity between communities composed by several species (OTUs)*
- Several distance metrics available in R
- Often used: Bray Curtis, Euclidean, Jaccard, Sorensen, Simpson

1 Distance metrics available in Vegan

2 "manhattan", "euclidean", "canberra", "clark", "bray", "kulczynski", "jaccard", "gower",
3 "altGower", "morisita", "horn", "mountford", "raup", "binomial", "chao", "cao", "mahalanobis",
4 "chisq" or "chord".

Bray Curtis distances for the rarefied datasets

```
1 # Distance metrics
2 # We calculate the Bray Curtis dissimilarities for the rarefied dataset
3 otu.tab.simple.ss.nozero.bray<-vegdist(otu.tab.simple.ss.nozero, method="bray")
4 as.matrix(otu.tab.simple.ss.nozero.bray)[1:5,1:5]

5 #          BL040126  BL040419  BL040719  BL041019  BL050120
6 # BL040126  0.0000000  0.8087457  0.9264692  0.8720639  0.5661498
7 # BL040419  0.8087457  0.0000000  0.9017733  0.8754062  0.8352985
8 # BL040719  0.9264692  0.9017733  0.0000000  0.7490484  0.9118002
9 # BL041019  0.8720639  0.8754062  0.7490484  0.0000000  0.8183084
10 # BL050120  0.5661498  0.8352985  0.9118002  0.8183084  0.0000000
```

Ordination

In ecological terms: ordination serves to summarise community data (such as species abundance data) by producing a low-dimensional ordination space in which *similar species and samples are plotted close together, and dissimilar species and samples are placed far apart.*

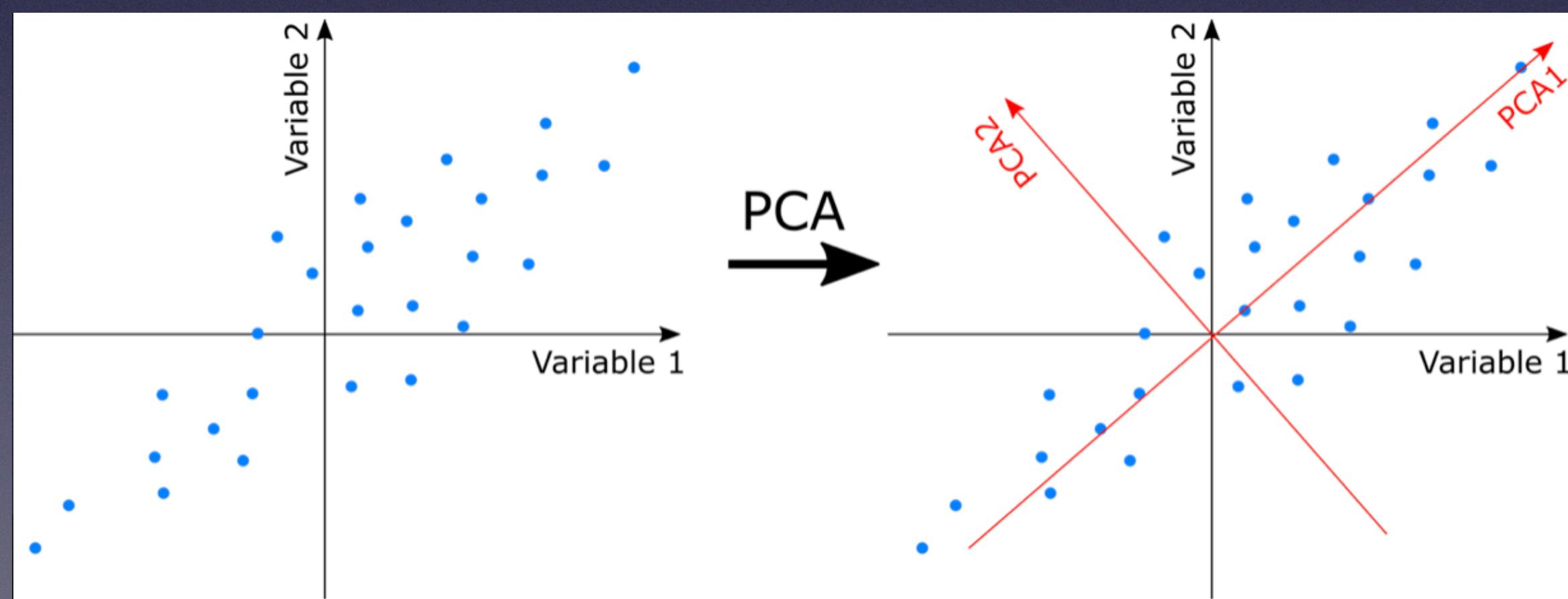
Ordination approaches

Two commonly used unconstrained techniques

- Principal Component Analysis (PCA)
- Non-metric Multidimensional Scaling (NMDS)

PCA

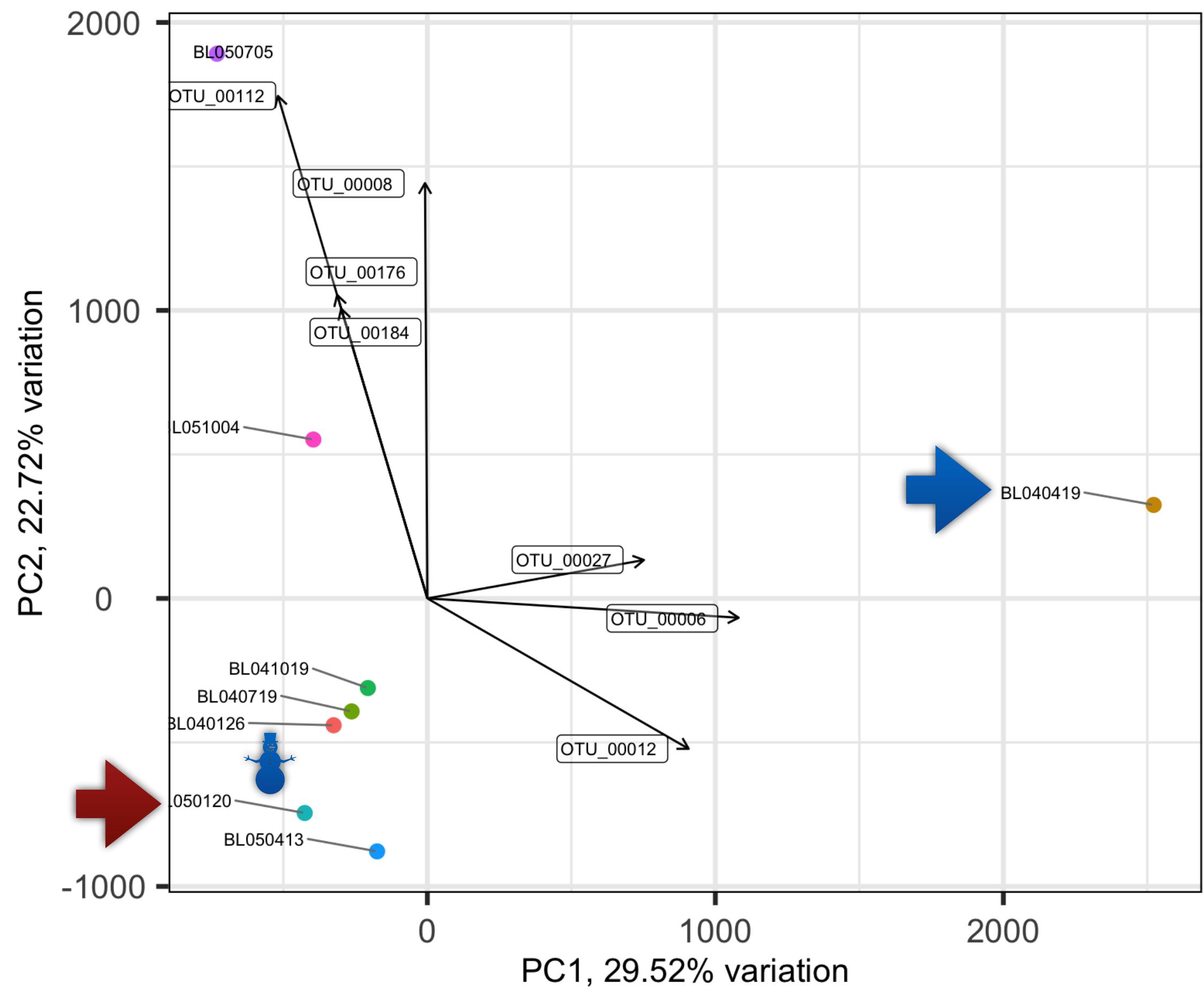
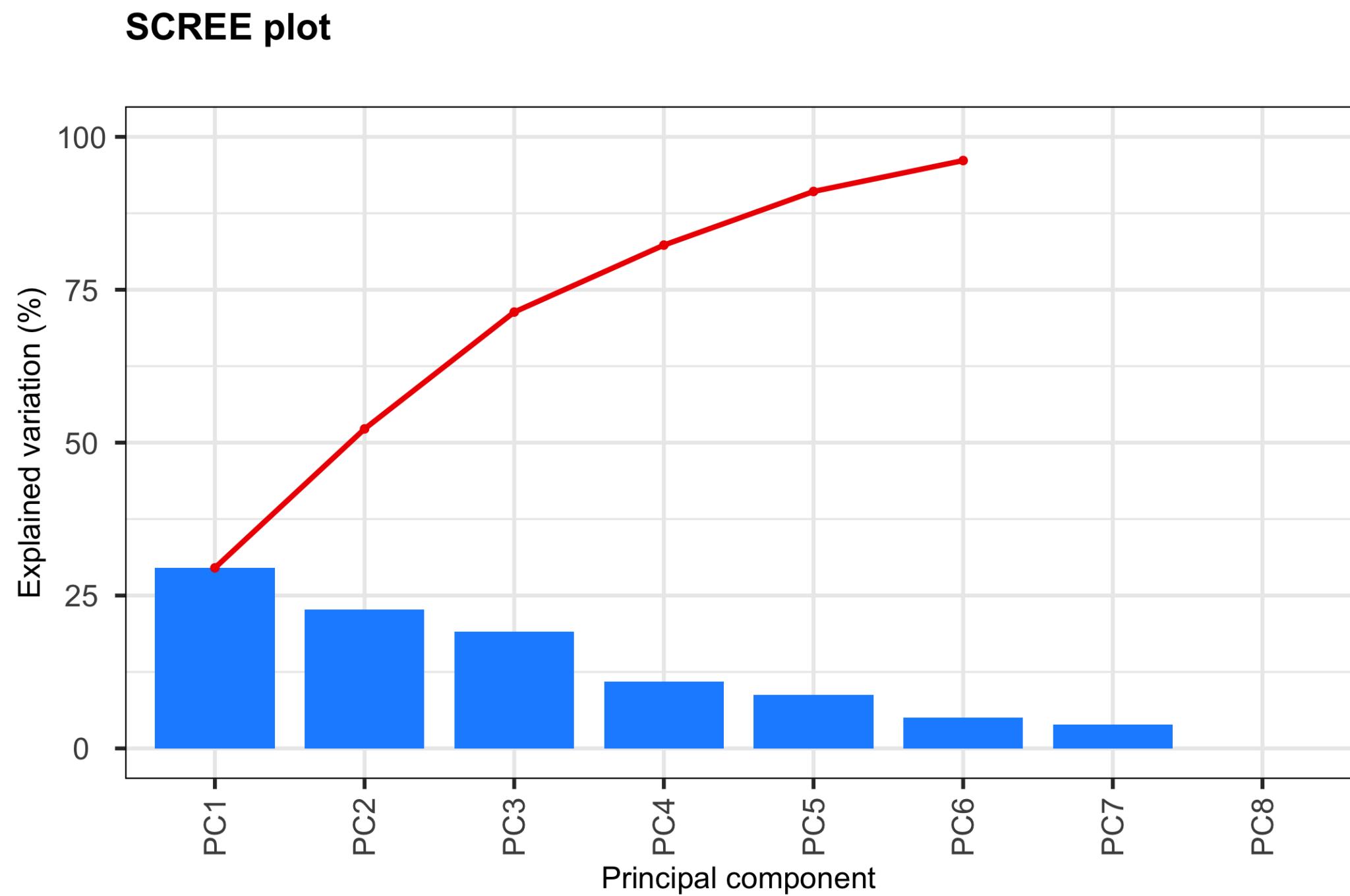
Rotates the original axes in order to maximise the 2D variability. The first principal component (PC) will be placed in the direction of the maximum variability and subsequent PCs will be generated in the same manner



PCA

```
1 #Ordination and clustering
2
3 #PCA
4
5 # We install PCAtools
6 if (!requireNamespace('BiocManager', quietly = TRUE))
7   install.packages('BiocManager')
8
9 BiocManager::install('PCAtools')
10
11 library(PCAtools)
12
13 #PCA rarefied table
14
15 # Run PCA
16 otu.tab.simple.ss.nozero.pca<-pca(t(otu.tab.simple.ss.nozero), scale=FALSE) #
17
18 biplot(otu.tab.simple.ss.nozero.pca, showLoadings = T,
19         lab=rownames(otu.tab.simple.ss.nozero)) # Plots de PCA
20
21 screeplot(otu.tab.simple.ss.nozero.pca, axisLabSize = 18, titleLabSize = 22)
22
23 # We plot the percentage of variance explained by each axis
```

PCA



Percentage of variance explain by each PC

Samples and OTUs are plotted. The arrows indicate the weight of each OTU in the different directions

Non-metric Multidimensional Scaling (NMDS)

- NMDS is more robust than PCA (e.g., is not affected by the arch effect)
- NMDS attempts to represent the pairwise dissimilarity between objects in a low-dimensional space
- Any distance metric can be used to build the distance matrix
- NMDS is a rank approach, meaning that ranks replace distances
- The stress value indicates how well the ordination summarises the observed distances among the samples

```
1 # We calculate NMDS for k(dimensions)=2  
2 # Rarefied table (we use the dataframe to have access to sample and OTU names)  
3 otu.tab.simple.ss.nozero.bray.nmds<-metaMDS(otu.tab.simple.ss.nozero, k=2,  
      trymax=100, trace=F, autotransform = F, distance="bray")  
4 # Check Stress <0.2
```

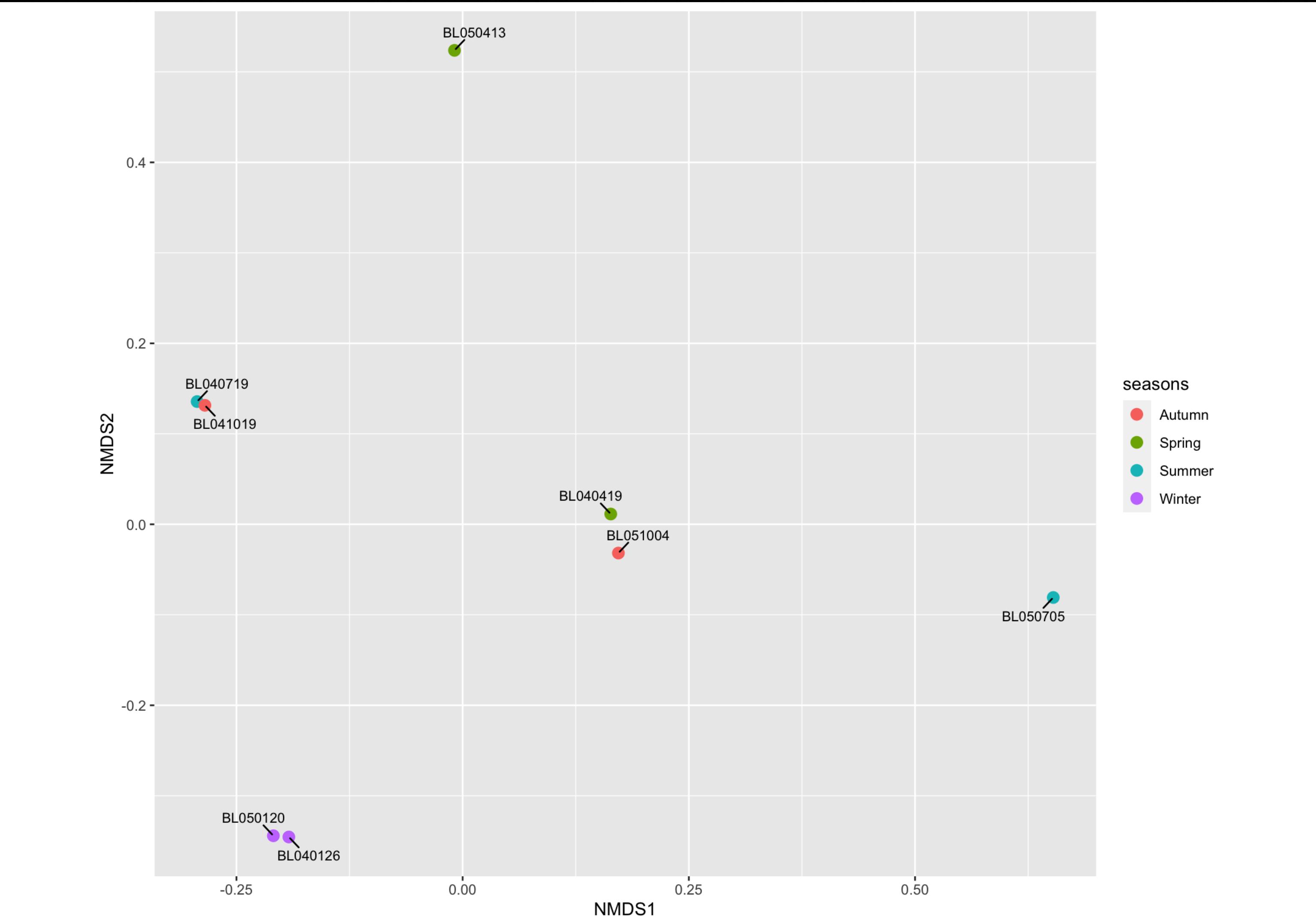
```
1 # Simple plotting  
  
2 # Rarefied table  
3 plot(otu.tab.simple.ss.nozero.bray.nmds, display="sites", type="n")  
4 points(otu.tab.simple.ss.nozero.bray.nmds, display = "sites", col = "red", pch=19)  
5 text(otu.tab.simple.ss.nozero.bray.nmds, display ="sites")  
6  
  
11  
12 # Let's make nicer plots  
13 # We define seasons for samples  
14 seasons<-c("Winter","Spring","Summer","Autumn","Winter","Spring","Summer","Autumn")  
15 months<-c("January","April","July","October","January","April","July","October")  
16  
17 library(ggplot2) # Generates nice plots  
18 library(ggrepel) # Adds in to ggplot  
19
```

```

20 # Rarefied table
21 # We generate a table of nmds scores and other features
22 otu.tab.simple.ss.nozero.bray.nmds.scores<-as.data.frame(scores(otu.tab.simple.ss.nozero.bray.nmds))
23 otu.tab.simple.ss.nozero.bray.nmds.scores$seasons<-seasons
24 otu.tab.simple.ss.nozero.bray.nmds.scores$months<-months
25 otu.tab.simple.ss.nozero.bray.nmds.scores$samples<-rownames(otu.tab.simple.ss.nozero.bray.nmds.scores)
26
27 #          NMDS1      NMDS2 seasons months samples
28 # BL040126 -0.192087931 -0.34552707 Winter January BL040126
29 # BL040419  0.163687487  0.01138097 Spring April  BL040419
30 # BL040719 -0.293448084  0.13565597 Summer July   BL040719
31 # BL041019 -0.284857321  0.13150682 Autumn October BL041019
32 # BL050120 -0.209189049 -0.34417159 Winter January BL050120
33 # BL050413 -0.009003643  0.52375809 Spring April  BL050413
34 # BL050705  0.652757387 -0.08086158 Summer July   BL050705
35 # BL051004  0.172141153 -0.03174161 Autumn October BL051004
36
37
38 # Create the plot
39 p <- ggplot(otu.tab.simple.ss.nozero.bray.nmds.scores) +
40   geom_point(mapping = aes(x = NMDS1, y = NMDS2, colour = seasons), size=3) +
41   coord_fixed()## need aspect ratio of 1!
42   geom_text_repel(box.padding = 0.5, aes(x = NMDS1, y = NMDS2, label = samples),
43                 size = 3)

```

NMDS plots



What ordination axis corresponds to the largest gradient in our dataset (i.e. the gradient explaining most of the variance)?

Incorporating environmental data

- We aim at investigating whether environmental variability could explain community variance
- Environmental variables are standardised to have comparable ranges of variation
- For each datapoint:

$$z = \frac{x - \mu}{\sigma}$$

Data point
↓
 x

Mean of all observations
← μ

Standard deviation of all observations
← σ

```

2 #Analyses using environmental variation

3 # We aim to investigate the environmental variation that may explain community variance.
4 # Read the environmental table
5 bbmo.metadata.course<-read_tsv("https://raw.githubusercontent.com/krabberod/BIO9905MERG1_V21/main/
community.ecology/bbmo.metadata.course.tsv", col_names = T)

7 bbmo.metadata.course<-as.data.frame(bbmo.metadata.course)
8 rownames(bbmo.metadata.course)<-bbmo.metadata.course[,1]
9 bbmo.metadata.course<-bbmo.metadata.course[,-1]
10
11 #
12 # ENV_Temp
13 # ENV_SECCHI
14 # ENV_SAL_CTD
15 # ENV_CHL_total
16 # ENV_PO4
17 # ENV_NH4
18 # ENV_NO2
19 # ENV_NO3
20 # ENV_SI
21 # ENV_BACTERIA
22 # ENV_SYNCHROS
23 # ENV_Micromonas
24 # ENV_PNF_tot
25 # ENV_HNF_tot
26 # ENV_Day_length_Hours_light
27 # Month
28 # Season
29 # Season_corr
30 # Year

BL040126 BL040419 BL040719 BL041019 BL050120 BL050413 BL050705 BL051004
14 12.6 24 19.2 13 13 24 21.5
14 6 24 12 19 18 22 17
37.9 35.9 36.9 37.5 37 37.7 37.35 35.1
1.1 1.4 0.4 0.3 0.5 2 0.1 0.6
0.2 0.2 0.1 0.1 0.2 0.3 0.2 0.2
0.3 1.5 1 0.5 1.1 2.1 1.4 1.5
0.3 0.4 0.2 0.1 0.2 0.4 0.1 0.1
1.5 2.5 0.1 0.4 1.1 3.3 0.2 2.4
1.8 6.1 1.4 1.4 2.6 3.4 1.8 1.6
854356 1046779 1654834 1083724 582655 788163 1127596 885144
5927 1411 38741 30915.5 8253 4169 24823 33866
9258 1424 203 730 4414 1543 505 573
11451 2266 1228 2811 5853 2506 1699 2052
329 1793 1357 822 420 669 1528 837
9.8 13.51 14.81 10.94 9.61 13.2 15.12 11.67
01_jan 04_apr 07_jul 10_oct 01_jan 04_apr 07_jul 10_oct
win spr sum aut win spr sum aut
win spr sum aut win spr sum aut
2004 2004 2004 2004 2005 2005 2005 2005

```

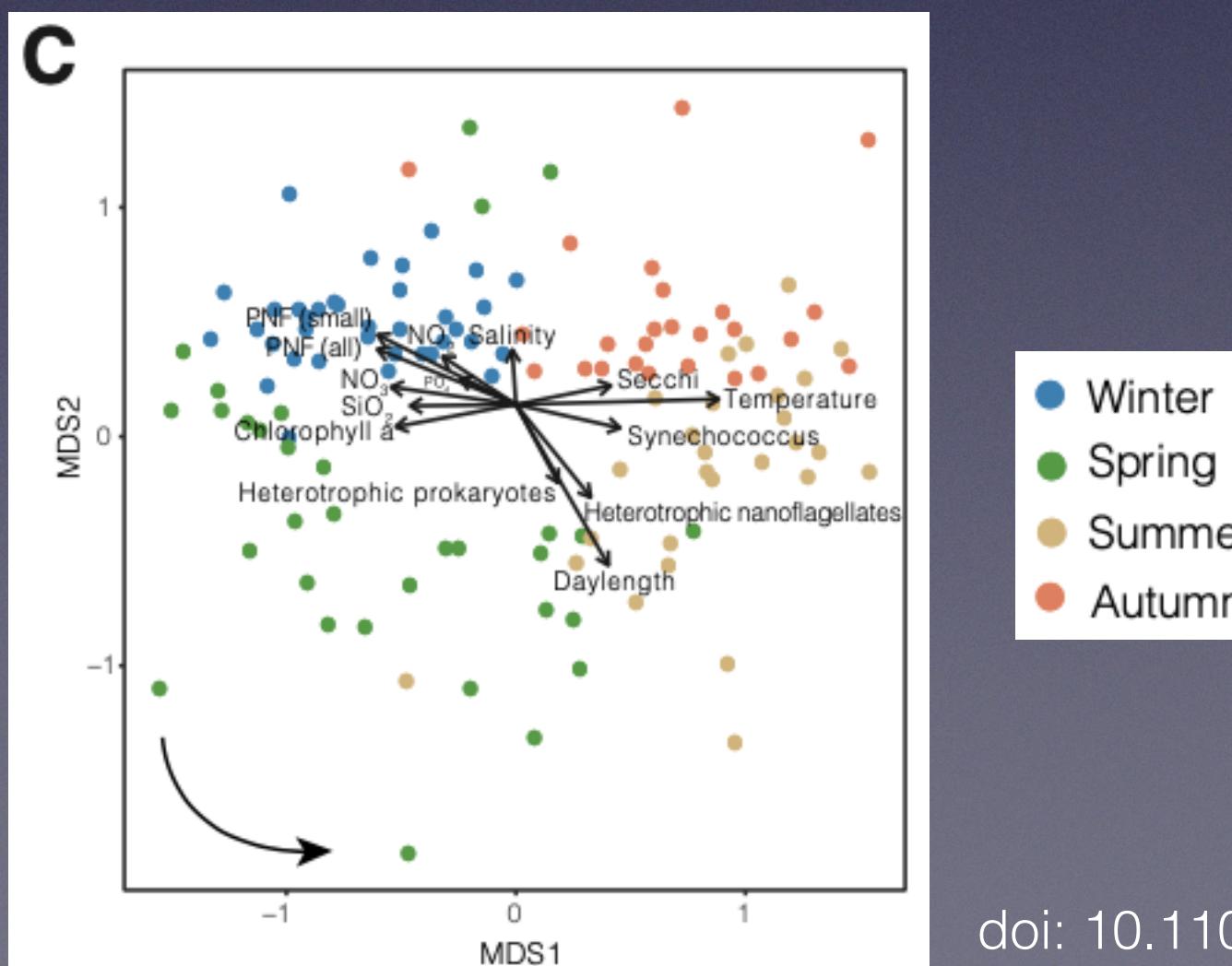
```
35 #We transform variables 1:15 using z-scores to have comparable ranges of variation  
  
36 bbmo.metadata.course.15vars<-bbmo.metadata.course[1:15,] #We select continuous variables  
37 bbmo.metadata.course.15vars[]<- lapply(bbmo.metadata.course.15vars, as.character) #We transform  
the datatype to characters  
38 bbmo.metadata.course.15vars[]<- lapply(bbmo.metadata.course.15vars, as.numeric) #We transform to  
numeric  
39 #lapply: applies a function to all elements  
  
  
  
  
40 bbmo.metadata.course.15vars.zscores<-scale(t(bbmo.metadata.course.15vars), center = T, scale = T)  
41 bbmo.metadata.course.15vars.zscores[,1:5]  
  
  
  
  
42  
43 #          ENV_Temp  ENV_SECCHI ENV_SAL_CTD ENV_CHL_total      ENV_PO4  
44 # BL040126 -0.7223777 -0.43425521  1.02225526       0.4644927  0.1950474  
45 # BL040419 -0.9985084 -1.82387188 -1.06132234       0.9289853  0.1950474  
46 # BL040719  1.2499845  1.30276563 -0.01953354      -0.6193235 -1.3653316  
47 # BL041019  0.3032507 -0.78165938  0.60553974      -0.7741544 -1.3653316  
48 # BL050120 -0.9196139  0.43425521  0.08464534      -0.4644927  0.1950474  
49 # BL050413 -0.9196139  0.26055313  0.81389750       1.8579706  1.7554264  
50 # BL050705  1.2499845  0.95536146  0.44927142      -1.0838162  0.1950474  
51 # BL051004  0.7568940  0.08685104 -1.89475338      -0.3096618  0.1950474
```

Unconstrained vs. Constrained ordination

- In unconstrained ordination, we first find the major compositional variation and then relate this variation to observed environmental variation (envfit e.g.)
- In constrained ordination we do not want to display all or even most of the compositional variation, but only the variation that the used environmental variables, or constraints can explain

Unconstrained ordination: Fitting vectors

- We correlate environmental variables with ordination axes
- The arrow points to the direction of most rapid change in the environmental variable. Often this is called the direction of the gradient
- The length of the arrow is proportional to the correlation between ordination and environmental variable. Often this is called the strength of the gradient



doi: 10.1101/2021.03.18.435965

vegan (version 2.4-2)
envfit: Fits an Environmental Vector or Factor onto an Ordination

Description
The function fits environmental vectors or factors onto an ordination. The projections of points onto vectors have maximum correlation with corresponding environmental variables, and the factors show the averages of factor levels.

Usage

```
"envfit"(ord, env, permutations = 999, strata = NULL, choices=c(1,2), display = "sites", w = weights(ord), na.rm = FALSE, ...)  
"envfit"(formula, data, ...)  
"plot"(x, choices = c(1,2), labels, arrow.mul, at = c(0,0), axis = FALSE, p.max = NULL, col = "blue", bg, add = TRUE, ...)  
"scores"(x, display, choices, ...)  
"vectorfit"(X, P, permutations = 0, strata = NULL, w, ...)  
"factorfit"(X, P, permutations = 0, strata = NULL, w, ...)
```

Constrained ordination

Redundancy Analysis (RDA) can be considered as a constrained version of PCA

- Distance based RDA: allows calculating RDA with a chosen distance matrix

Selecting environmental variables that explain most community variance

- *Forward selection*: begins with an empty model and adds variables one by one. In each step forward, it adds one variable that gives the single best improvement to the model
- *Backwards elimination*: starts with a model that includes all variables and eliminates variables with low explanatory power one by one

- **Ordistep** (Vegan): Performs step-wise selection of environmental variables based on two criteria:
 - If their inclusion into the model leads to a significant increase of the explained variance
 - If the AIC (Akaike Information Criterion) of the new model is lower than the AIC of the more simple model
 - AIC: estimates the quality of models relative to other models (model selection). It is an estimator of prediction error

```
1 #Constrained Ordination
2 # Selection of the most important variables for distance-based redundancy analyses
3
4 #Rarefaction table
5 mod0.rarefaction<-capscale(otu.tab.simple.ss.nozero.bray~1, as.data.frame(bbmo.metadata.course.15vars.zscores)) # model containing
       only species matrix and intercept
7 mod1.rarefaction<-capscale(otu.tab.simple.ss.nozero.bray~ ., as.data.frame(bbmo.metadata.course.15vars.zscores)) # # model including
       all variables from env matrix (the dot after tilde (~) means ALL!)
9 ordistep(mod0.rarefaction, scope = formula(mod1.rarefaction), perm.max = 1000, direction="forward")
10
11 # Start: otu.tab.simple.ss.nozero.bray ~ 1
12 #          Df      AIC      F Pr(>F)
13 # + ENV_PNF_tot        1 9.2535 1.3702  0.050 *
14 # + ENV_Day_length_Hours_light  1 9.1702 1.4474  0.055 .
15 # + ENV_Micromonas     1 9.2311 1.3909  0.055 .
16 # + ENV_BACTERIA      1 9.4129 1.2248  0.110
17 # + ENV_Temp           1 9.3168 1.3121  0.170
18 # + ENV_PO4            1 9.4892 1.1562  0.195
19 # + ENV_HNF_tot         1 9.4548 1.1870  0.240
20 # + ENV_SYNCHOS        1 9.4613 1.1812  0.255
21 # + ENV_NH4             1 9.5305 1.1193  0.285
22 # + ENV_NO3             1 9.5767 1.0784  0.350
23 # + ENV_NO2             1 9.6558 1.0087  0.380
24 # + ENV_CHL_total       1 9.5684 1.0857  0.385
25 # + ENV_SAL_CTD         1 9.6782 0.9891  0.485
26 # + ENV_SI              1 9.7718 0.9078  0.590
27 # + ENV_SECCHI         1 9.8076 0.8770  0.675
28 # ---
29 #   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

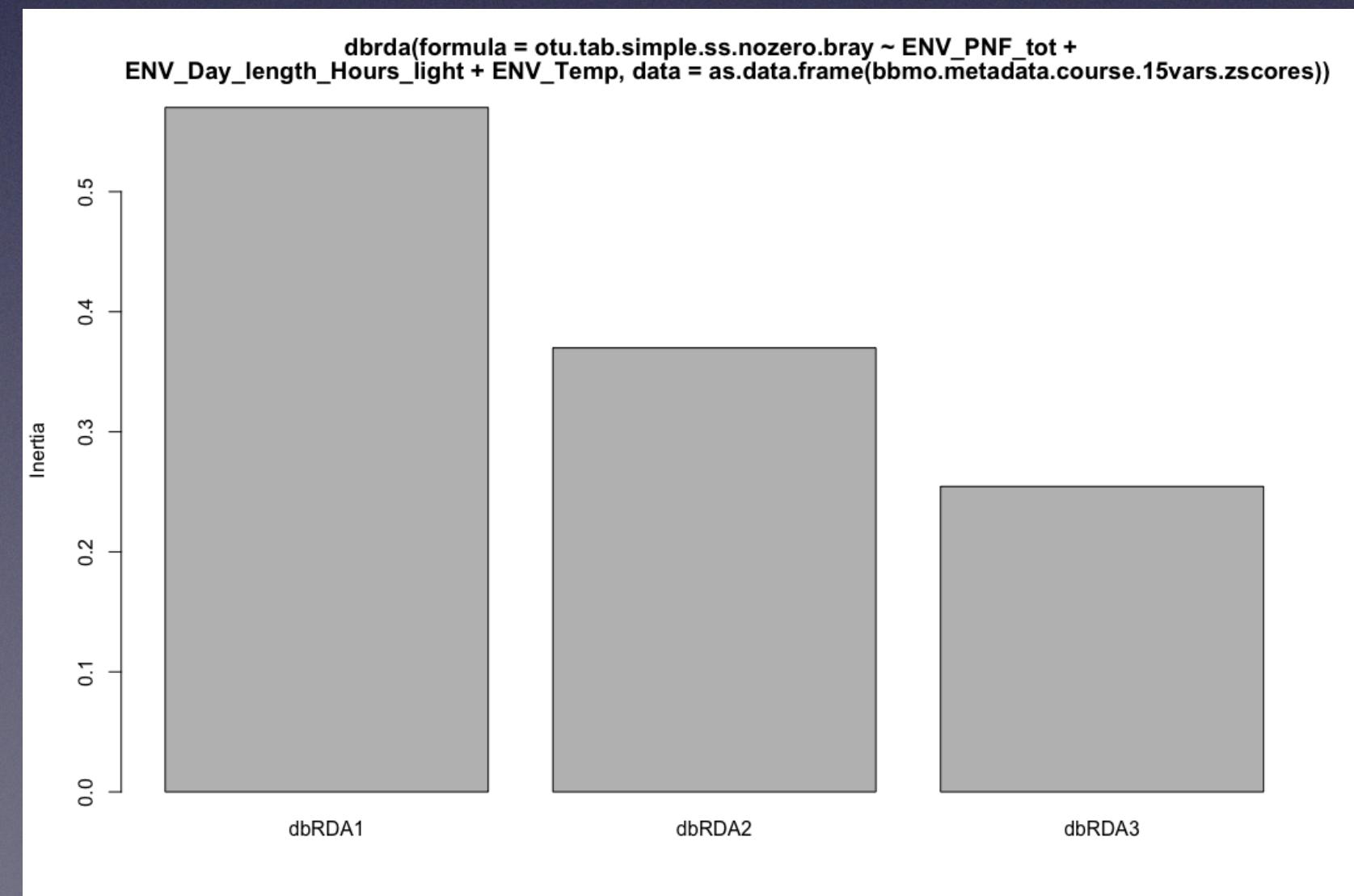
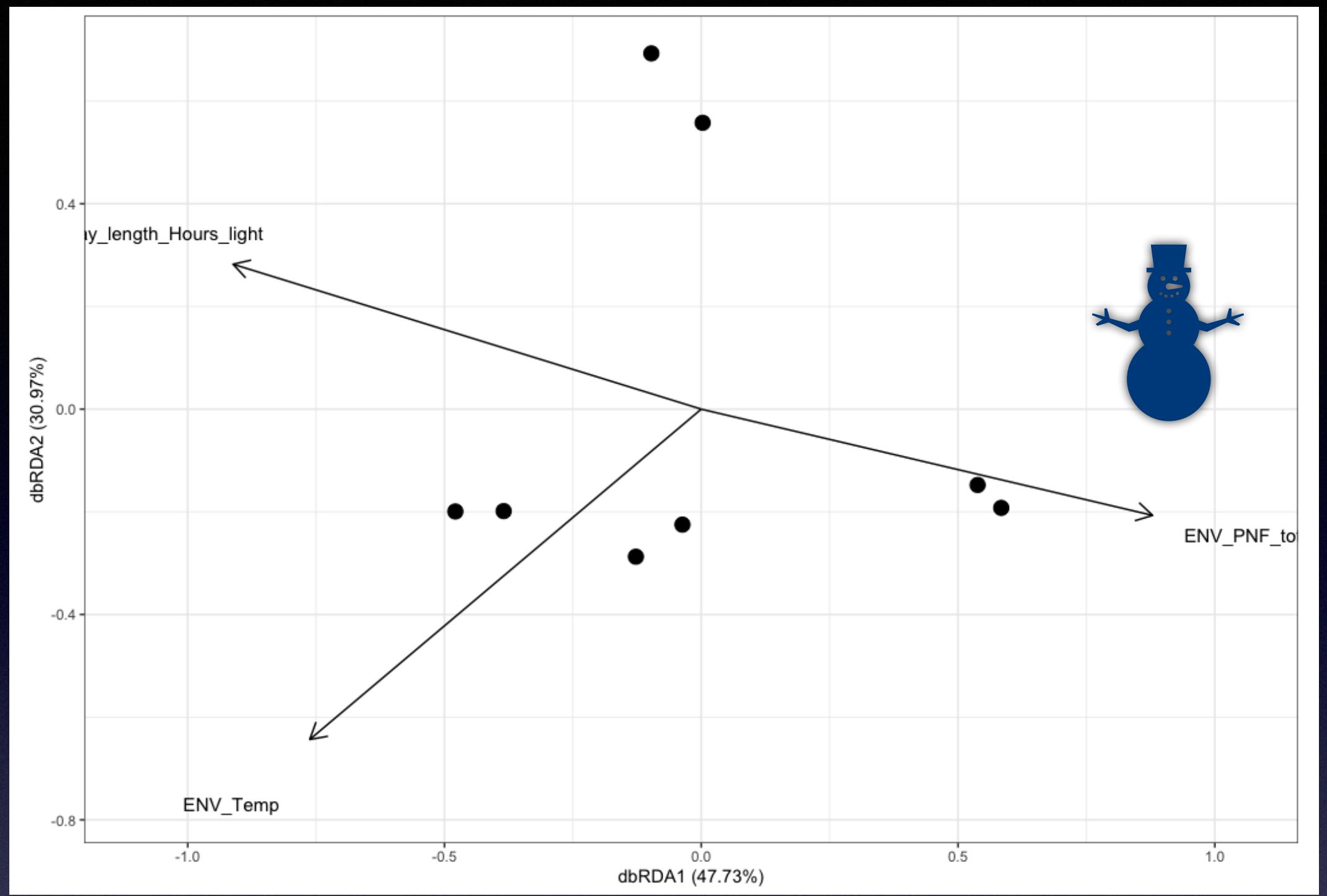
dbRDA

```
30
31 # Step: otu.tab.simple.ss.nozero.bray ~ ENV_PNF_tot
32 #          Df      AIC      F Pr(>F)
33 # + ENV_SAL_CTD      1 9.5007 1.2248 0.225
34 # + ENV_PO4         1 9.4897 1.2333 0.235
35 # + ENV_CHL_total    1 9.5584 1.1800 0.285
36 # + ENV_NO3         1 9.6004 1.1477 0.285
37 # + ENV_NH4         1 9.6031 1.1456 0.330
38 # + ENV_NO2         1 9.7120 1.0625 0.370
39 # + ENV_Temp        1 9.7212 1.0555 0.380
40 # + ENV_SYNCHOS     1 9.7690 1.0195 0.465
41 # + ENV_SI          1 9.7931 1.0013 0.600
42 # + ENV_BACTERIA    1 9.8945 0.9258 0.620
43 # + ENV_HNF_tot      1 9.9316 0.8983 0.645
44 # + ENV_SECCHI       1 9.9584 0.8786 0.675
45 # + ENV_Day_length_Hours_light 1 10.0502 0.8115 0.720
46 # + ENV_Micromonas   1 10.0363 0.8216 0.745
47
48 # Call: capscale(formula = otu.tab.simple.ss.nozero.bray ~ ENV_PNF_tot, data = as.data.frame(bbmo.metadata.course.15vars.zscores))
49 # NB: the variables in this model are the ones that were selected.
50
51 # Inertia Proportion Rank
52 # Total      2.7072  1.0000
53 # Constrained 0.5033  0.1859  1
54 # Unconstrained 2.2039  0.8141  6
55 # Inertia is squared Bray distance
56
57 # Eigenvalues for constrained axes:
58 #   CAP1
59 # 0.5033
60
61 # Eigenvalues for unconstrained axes:
62 #   MDS1   MDS2   MDS3   MDS4   MDS5   MDS6
63 # 0.5958 0.4353 0.3912 0.2791 0.2778 0.2246
```

We use two more variables that are known to be important drivers of community variance

1. Day-length
2. Temperature

```
1 #Generate the ordination
2 # We will use two more variables that we know they are important in this dataset
3
4 #We install ggord for nicer plots
5 library(devtools)
6 install_github('fawda123/ggord')
7 library(ggord)
8 library(ggplot2)
9
10 #rarefied table
11 ggord(dbrda(formula = otu.tab.simple.ss.nozero.bray ~ ENV_PNF_tot+ENV_Day_length_Hours_light+ENV_Temp, data = as.data.frame(bbmo.metadata.course.
12           15vars.zscores)))
13 screeplot(dbrda(formula = otu.tab.simple.ss.nozero.bray ~ ENV_PNF_tot+ENV_Day_length_Hours_light+ENV_Temp, data = as.data.frame(bbmo.metadata.
14           course.15vars.zscores)))
15 dbrda(formula = otu.tab.simple.ss.nozero.bray ~ ENV_PNF_tot+ENV_Day_length_Hours_light+ENV_Temp, data = as.data.frame(bbmo.metadata.course.15vars.
16           zscores))
17
18 # Call: dbrda(formula = otu.tab.simple.ss.nozero.bray ~ ENV_PNF_tot + ENV_Day_length_Hours_light + ENV_Temp, data =
19 #                   as.data.frame(bbmo.metadata.course.15vars.zscores))
20 #           Inertia Proportion Rank
21 # Total      2.7072    1.0000
22 # Constrained 1.1945    0.4412    3  # Community variation constrained by the used variables
23 # Unconstrained 1.5127    0.5588    4
24 # Inertia is squared Bray distance # Inertia = variance in species abundances
25
26 # Eigenvalues for constrained axes:
27 #   dbRDA1 dbRDA2 dbRDA3
28 #   0.5701 0.3699 0.2545
29
30 # Eigenvalues for unconstrained axes:
31 #   MDS1   MDS2   MDS3   MDS4
32 #   0.5818 0.3960 0.2997 0.2353
33
```



Core microbiota BBMO: example

- What taxa constitute the interconnected core microbiota over 10 years at one marine location?
- How the diversity of core taxa changes over time? What are the seasonal patterns?
- What are their potential ecological interactions?

Krabberød et al. *Environmental Microbiome* (2022) 17:22
<https://doi.org/10.1186/s40793-022-00417-1>

Environmental Microbiome

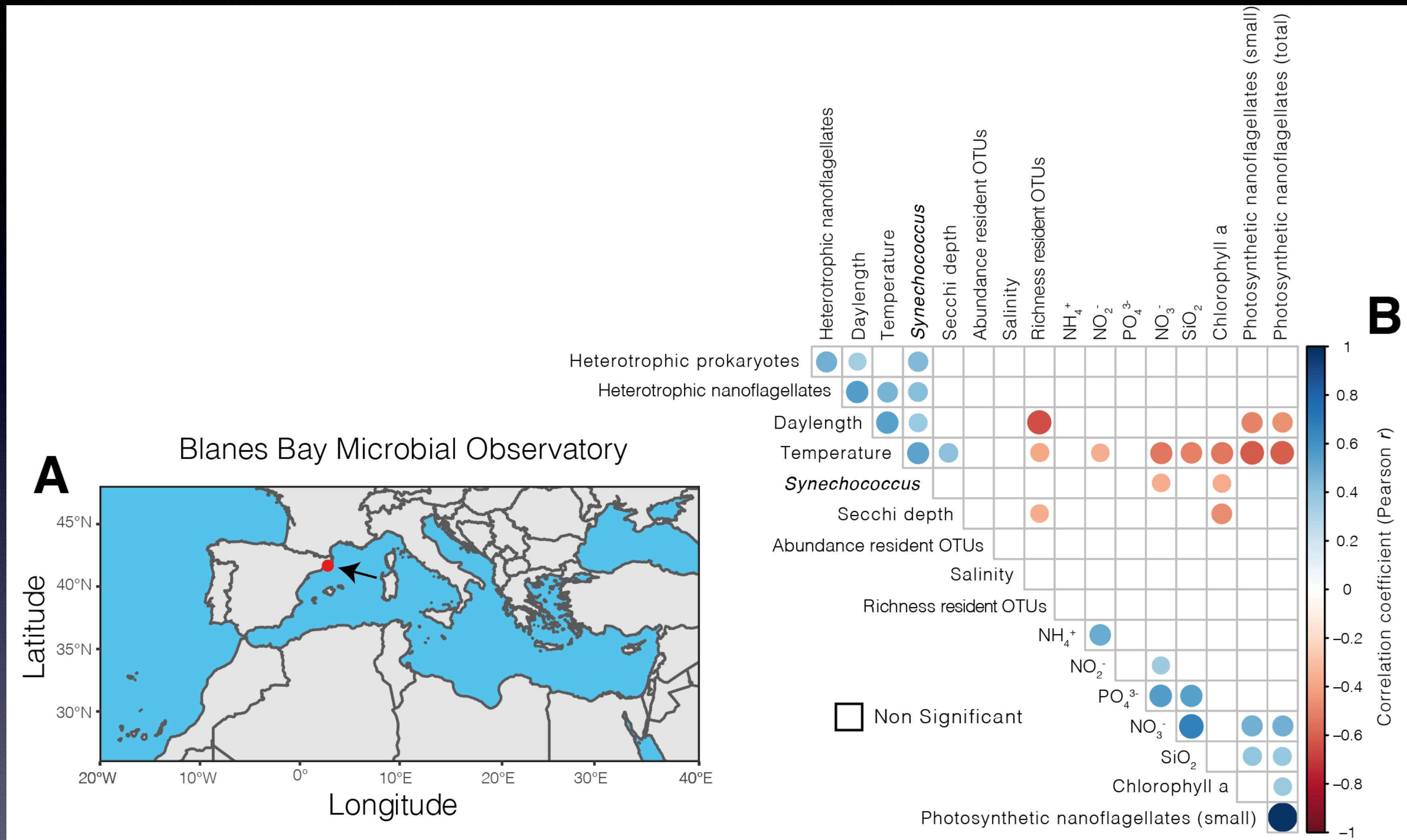
RESEARCH ARTICLE

Open Access

Long-term patterns of an interconnected core marine microbiota

Anders K. Krabberød^{1*}, Ina M. Deutschmann², Marit F. M. Bjorbækmo¹, Vanessa Balagué², Caterina R. Giner², Isabel Ferrera^{2,3}, Esther Garcés², Ramon Massana², Josep M. Gasol^{2,4} and Ramiro Logares^{1,2*} 





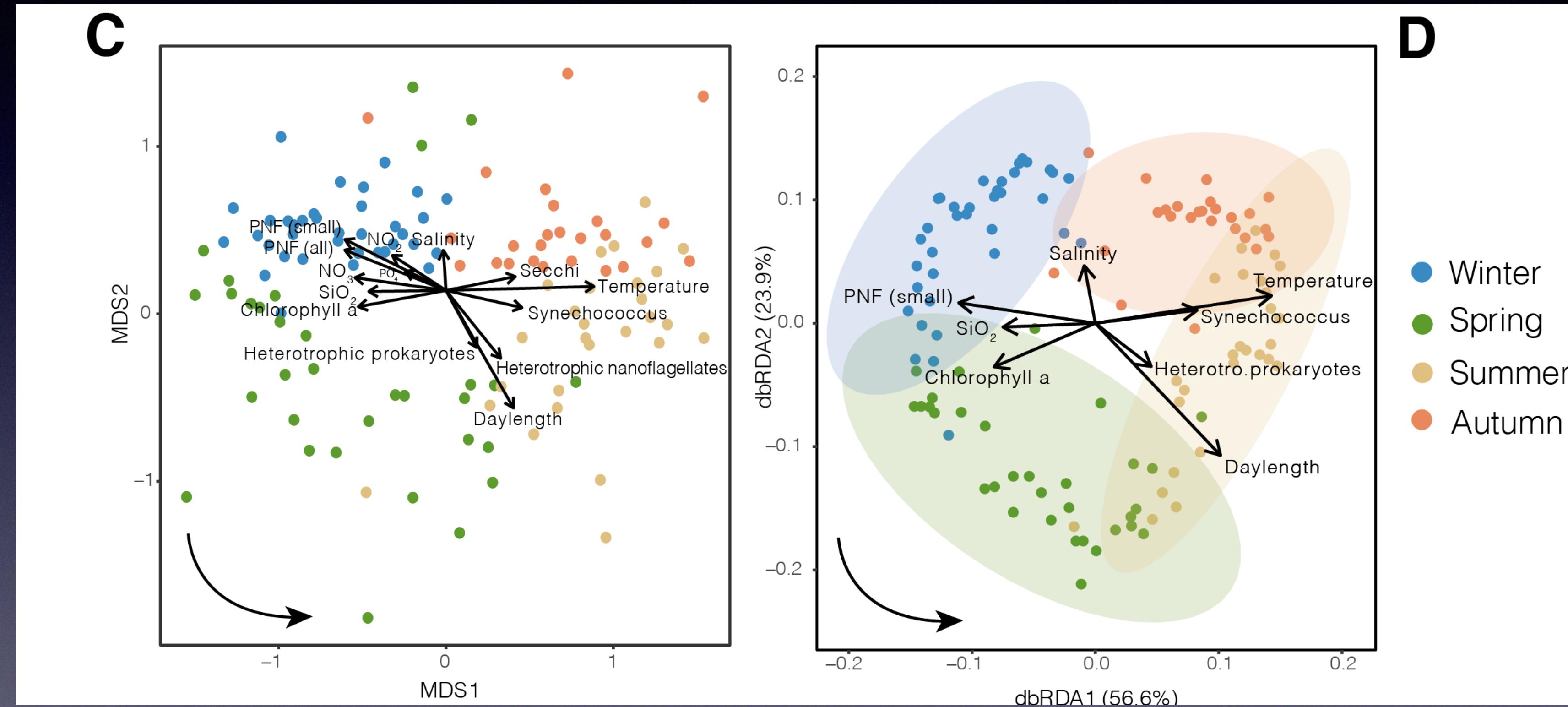
zscores

Pearson correlations



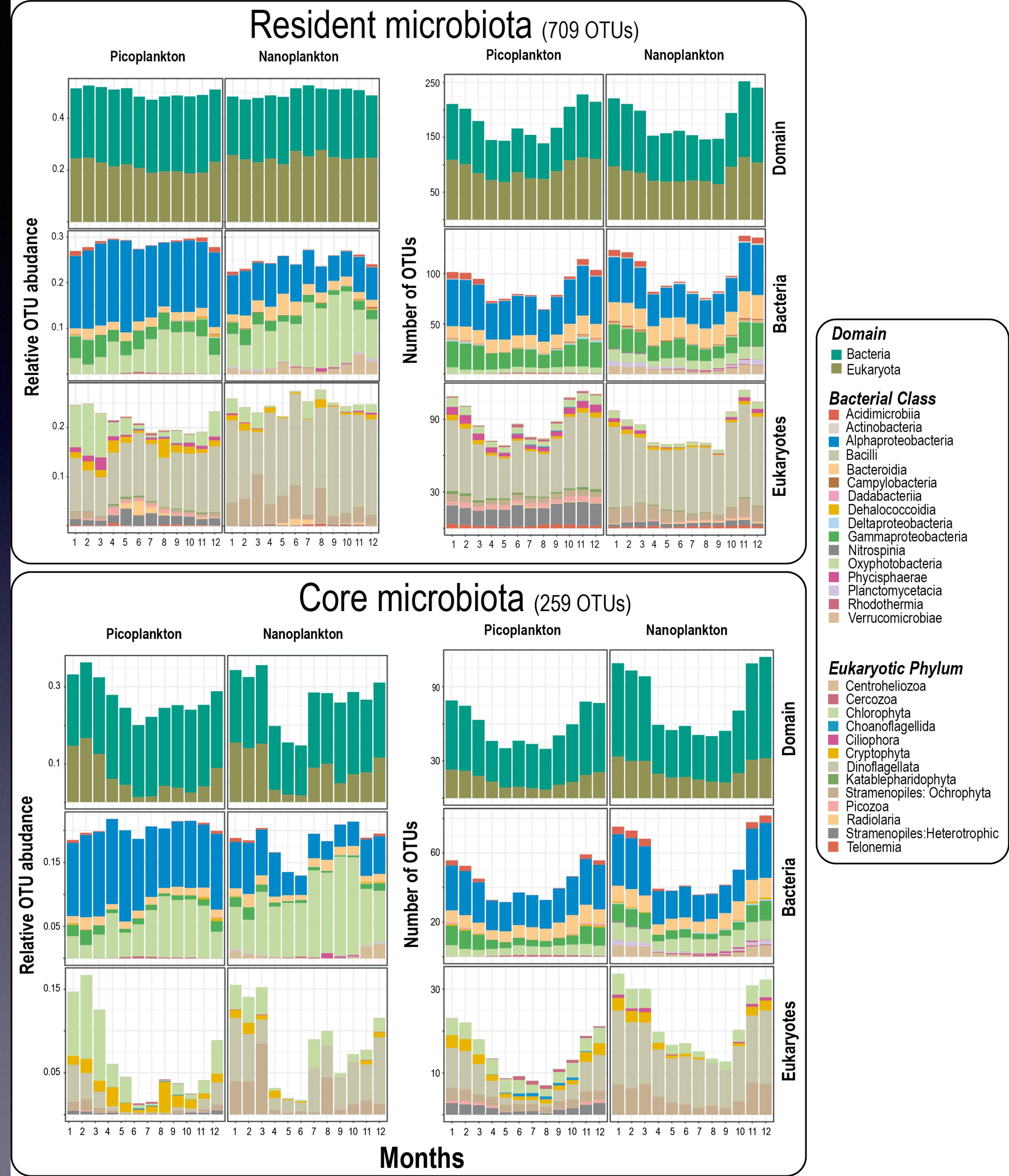


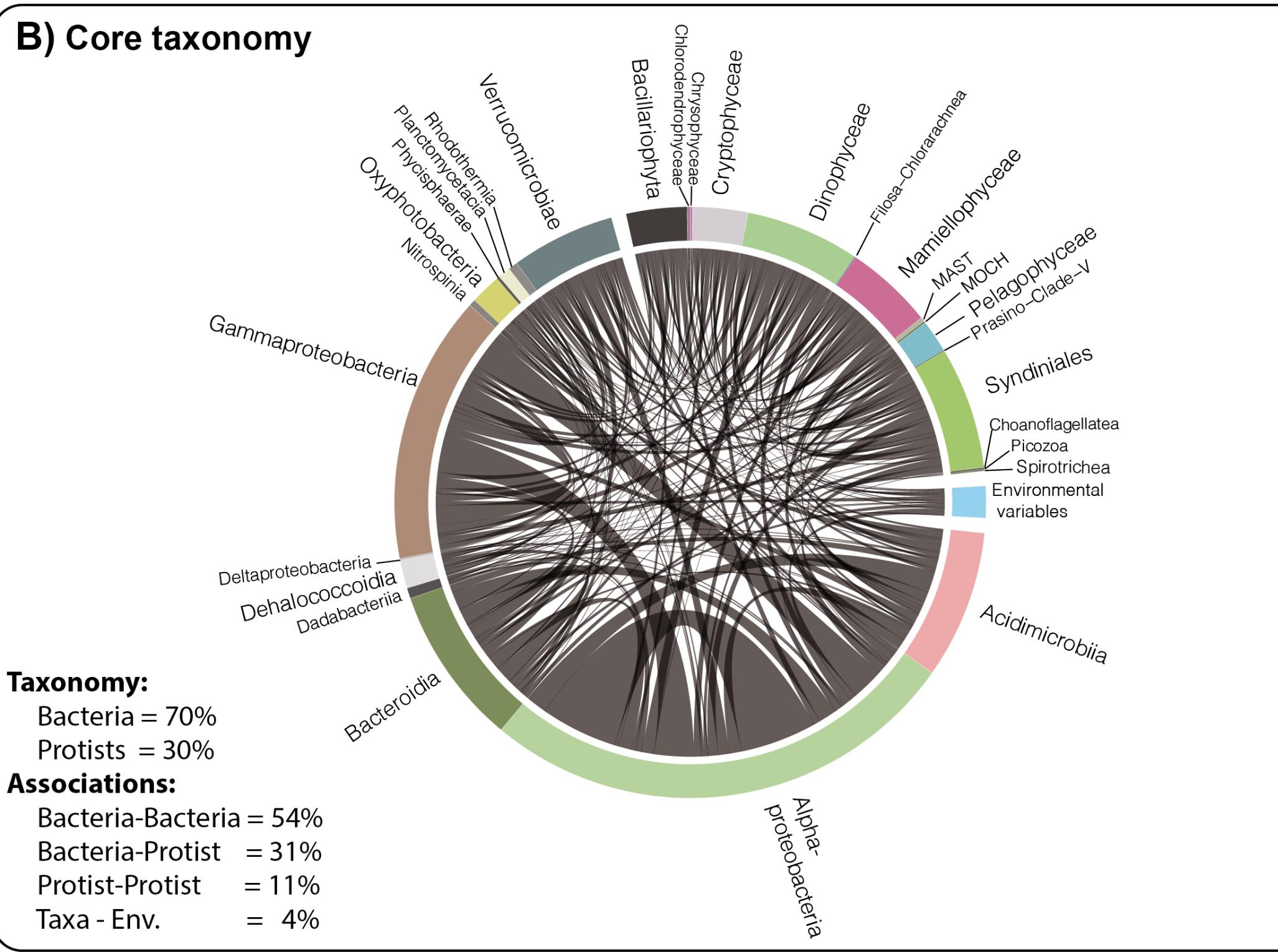
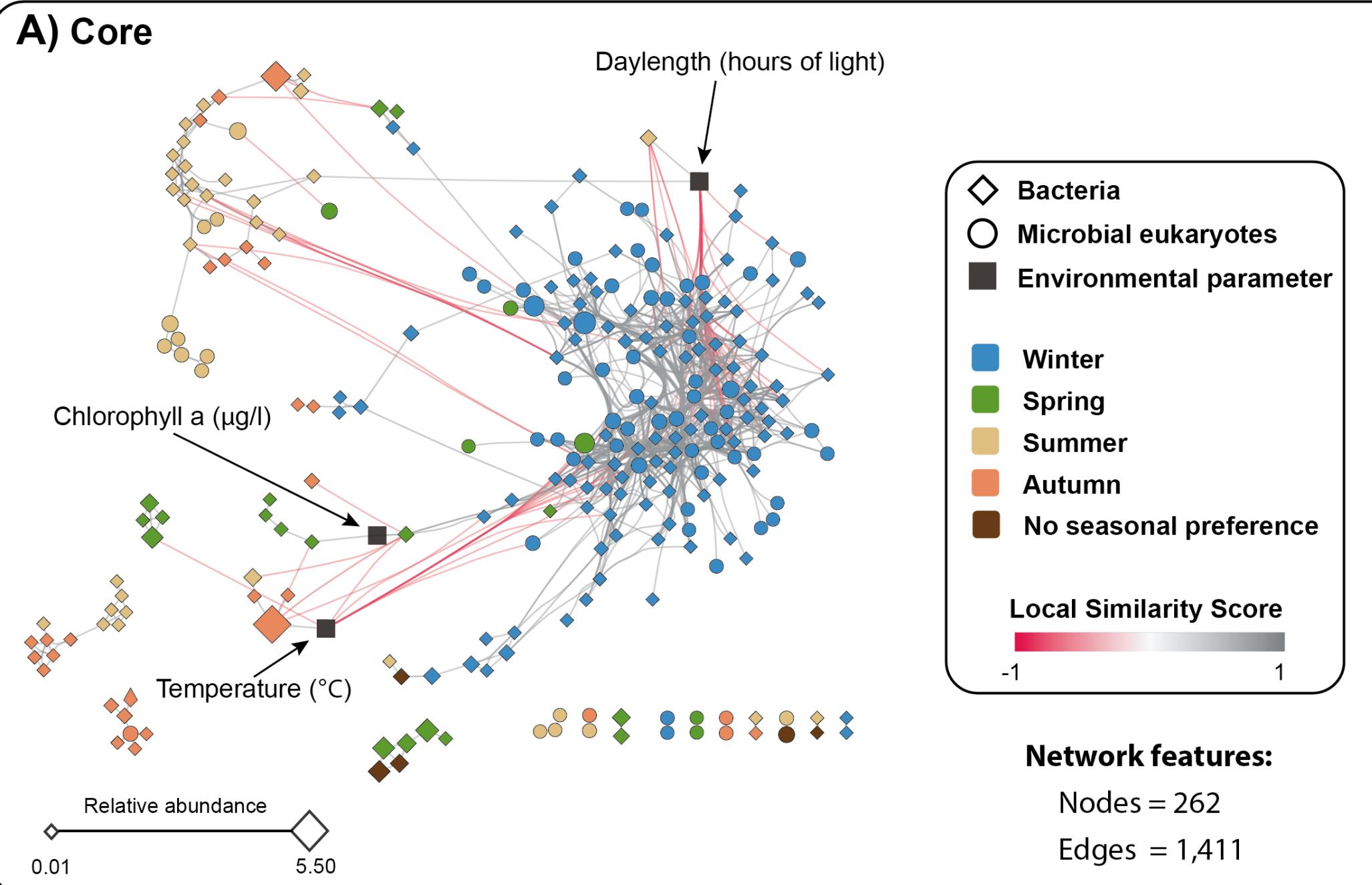
NMDS
envfit



dbRDA
Forward selection

Richness 😐
Relative abundance





Conclusions of the study

- The core microbiota included 259 Operational Taxonomic Units (OTUs) including 182 bacteria, 77 protists, and 1411 strong and mostly positive (~ 95%) associations.
- The richness and abundance of core OTUs varied annually, decreasing in stratified warmers waters and increasing in colder mixed waters.
- Most core OTUs had a preference for one season, mostly winter, which featured subnetworks with the highest connectivity.

Other things you could explore

- The relative importance of the main processes structuring microbiotas
- Selection
- Dispersal
- Drift

The ISME Journal (2013) 7, 2069–2079
© 2013 International Society for Microbial Ecology All rights reserved 1751-7362/13
www.nature.com/ismej 

ORIGINAL ARTICLE
Quantifying community assembly processes and identifying features that impose them

James C Stegen¹, Xueju Lin^{1,2}, Jim K Fredrickson¹, Xingyuan Chen³, David W Kennedy¹, Christopher J Murray⁴, Mark L Rockhold³ and Allan Konopka¹
¹*Biological Sciences Division, Pacific Northwest National Laboratory, Richland, WA, USA; ²School of Biology, Georgia Institute of Technology, Atlanta, GA, USA; ³Hydrology Group, Pacific Northwest National Laboratory, Richland, WA, USA and ⁴Department of Geosciences, Pacific Northwest National Laboratory, Richland, WA, USA*

R code
https://github.com/stegen/Stegen_etal_ISME_2013

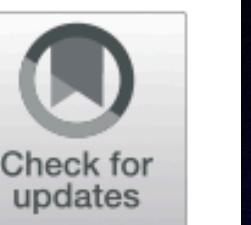
RESEARCH

Microbiome

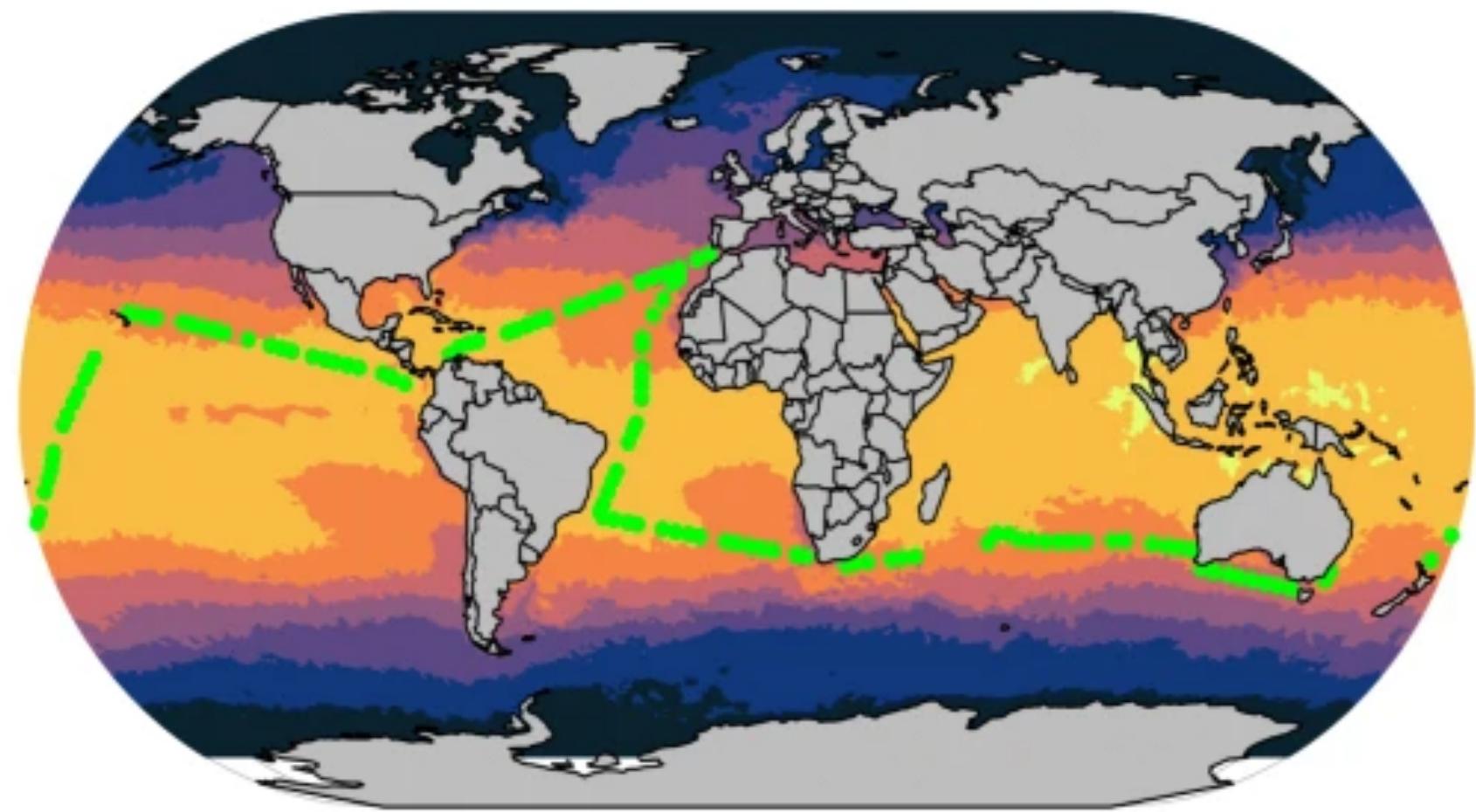
Open Access

Disentangling the mechanisms shaping the surface ocean microbiota

Ramiro Logares^{1,2*}, Ina M. Deutschmann¹, Pedro C. Junger³, Caterina R. Giner^{1,4}, Anders K. Krabberød², Thomas S. B. Schmidt⁵, Laura Rubinat-Ripoll⁶, Mireia Mestre^{1,7,8}, Guillem Salazar^{1,9}, Clara Ruiz-González¹, Marta Sebastián^{1,10}, Colomban de Vargas⁶, Silvia G. Acinas¹, Carlos M. Duarte¹¹, Josep M. Gasol^{1,12} and Ramon Massana¹



A



Sea Surface Temperature (°C)

B

