Host-specificity and core taxa of seagrass leaf microbiome identified across tissue age and geographical regions | *Sanders-Smith, R. & Segovia, B.T.(joint contribution)*, Forbes, C., Hessing-Lewis, M., Morien, E., Lemay, M.A., O'Connor, M. I., Parfrey, L.W.

Bianca Trevizan Segovia

20/11/2020

Phyloseq pipeline filtering with tree

```
#### load packages ####
library(phyloseq)
library(tidyverse)
library(vegan)
library(dplyr)
library(QsRutils)

#### importing files ####
mapping_file <- import_qiime_sample_data("data/mapping_seagrass_2015_final.txt")
otu_tax_table <- import_biom(file.path("data/16s_seagrass_OTU_Table.filtered.2015quadrats.wtaxa.1000c.bi
otu_table_txt <- read.table(
  "data/16s_seagrass_OTU_Table.filtered.2015quadrats.wtaxa.1000c.txt")
tree_file <- read_tree("data/16s_makephylo_fasttree.tre")

#### merge otu data, mapping_file, and tree data into single phyloseq object ####
phylo_merge <- merge_phyloseq(otu_tax_table, mapping_file, tree_file)
```

```
#### Quality filtering taxa data ####
# 1. Remove mitochondrial and chloroplast OTUs
phylo_merge <- phylo_merge %>%
  subset_taxa(Rank5 != "Mitochondria"| is.na(Rank5)) %>%
  subset_taxa(Rank3 != "Chloroplastida"| is.na(Rank3)) %>%
  subset_taxa(Rank1 != "Unassigned"| is.na(Rank1))

# Filtering per sample
# 2. Remove OTUs with fewer than three reads per sample/noise
otu <- as.data.frame(otu_table(phylo_merge))
otu_table(phylo_merge)[otu <= 3] <- 0 #free of noise, I set to 3 otus/sample
otu2 <- as.data.frame(otu_table(phylo_merge)) #free of noise

# Filtering overall
# 3. Remove OTUs with less than 250 total reads
phylo_merge <- prune_taxa(taxa_sums(phylo_merge) >= 250, phylo_merge)
```

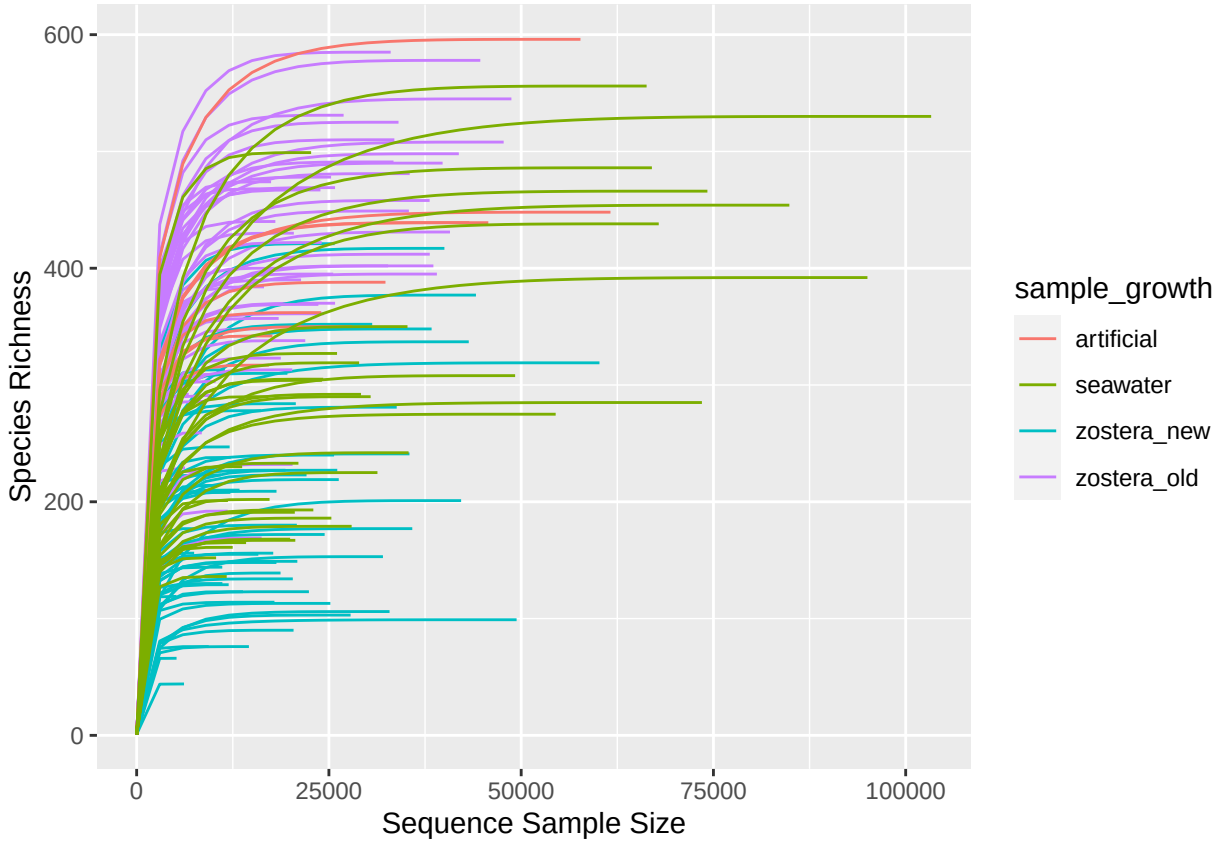
```
# 4. Remove samples with low read count (less than 1,000 reads)
phylo_merge <- prune_samples(sample_sums(phylo_merge) >= 1000, phylo_merge)

# 5. look at minimum, mean, and maximum sample counts, if desired
summary(sample_sums(phylo_merge))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3130   16249   22972   27201   34050  103313
```

```
# save non rarefied data as RDS to run shared_taxa_Venn_diagram_16S.R script
saveRDS(phylo_merge, "data/phylo_merge_not_rarefied_16S_paper.rds")
```

```
#### plot rarefaction curves ####
# found rarefaction curve function here:
# https://rdrr.io/github/gauravsk/ranacapa/src/R/ggrare.R
#download this script
#to use this you have to load the ggrare() function in from ggrare.R
source("ggrare.R") # load in ggrare function
p <- ggrare(phylo_merge, step = 3000, color = "sample_growth", se = FALSE)
```




```
ggsave("final_figures/rarefaction_curves.jpeg",plot = p,
       width=300, height=250, units="mm",dpi=300)
```

```
#### Rarefy data ####
# no samples below 3000 sequences per sample
# so we retained the maximum number of samples in our analyses
phylo_merge_rare <- rarefy_even_depth(phylo_merge,
                    sample.size = 3000,
```

```
                    rngseed = 7, # set seed for reproducibility
                    replace = FALSE)# sample without replacement; slower but more accurate
```

```
## `set.seed(7)` was used to initialize repeatable random subsampling.

## Please record this for your records so others can reproduce.

## Try `set.seed(7); .Random.seed` for the full vector

## ...
```

```
# save rarefied data as RDS to run NMDS
saveRDS(phylo_merge, "data/phylo_merge_rarefied_16S_paper.rds")

### Saving tables to perform analyses ###
project_data.3000.otu <- as.data.frame(otu_table(phylo_merge_rare))

project_data.3000.tax <- as.data.frame(tax_table(phylo_merge_rare))

project_data.3000.sam <- as.data.frame(unclass(sample_data(phylo_merge_rare)))

write.csv(project_data.3000.otu, file="data/species_final.otu.csv")

write.csv(project_data.3000.tax, file="data/species_final.tax.csv")

write.csv(project_data.3000.sam, file="data/species_final.sam.csv", row.names = F)

### Read otu table and transpose it so samples are rows and taxa are columns ###
bact3000 <- read.csv("data/species_final.otu.csv", header=T)
bact3000_trans <- setNames(data.frame(t(bact3000[,-1])), bact3000[,1])
bact3000_final <- tibble::rownames_to_column(bact3000_trans, "SampleID")

### Read metadata
metadata <- read.csv(file="data/species_final.sam.csv", header=T)
bact_metadata <- left_join(metadata, bact3000_final, by = "SampleID")

write.csv(bact_metadata, file="data/Bact_3000_metadata_clean.csv",
          quote=F, row.names=F)
```