

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

Indicator Species Analysis (IndVal)

core taxa

```
### load packages ###

# load packages
library(dplyr)
library(vegan)
library(labdsv)
library(indicspecies)
library(phyloseq)
library(tidyverse)
library(reshape2)
library(stringr)

### set.seed for reproducibility ###
set.seed(333)

### importing files ###
phylo_merge_16S_nr <- readRDS("data/phylo_merge_not_rarefied_16S_paper.rds")
phylo_merge_16S_nr

## phyloseq-class experiment-level object
## otu_table() OTU Table: [ 1206 taxa and 149 samples ]
## sample_data() Sample Data: [ 149 samples by 4 sample variables ]
## tax_table() Taxonomy Table: [ 1206 taxa by 7 taxonomic ranks ]
## phy_tree() Phylogenetic Tree: [ 1206 tips and 1204 internal nodes ]

### Saving tables to perform analyses ###
project_data.otu <- as.data.frame(otu_table(phylo_merge_16S_nr))

project_data.tax <- as.data.frame(tax_table(phylo_merge_16S_nr))

project_data.sam <- as.data.frame(unclass(sample_data(phylo_merge_16S_nr)))
```

```

project_data.otu <- project_data.otu %>%
  rownames_to_column(var = "OTU")

project_data.tax <- project_data.tax %>%
  rownames_to_column(var = "OTU")

write.csv(project_data.otu, file="data/species_final_NONRAREFIED.otu.csv",
  quote=F, row.names=F)

write.csv(project_data.tax, file="data/species_final_NONRAREFIED.tax.csv",
  quote=F, row.names=F)

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

### transpose the otu table so taxa are columns and samples are rows ###
Bact_nonrarefied <- read.csv("data/species_final_NONRAREFIED.otu.csv", header=T)

Bact_nonrarefied <- data.frame(Bact_nonrarefied[,-1], row.names=Bact_nonrarefied[,1])

### transpose it while keeping column names ###
Bact_nonrarefied_trans <- as.data.frame(t(Bact_nonrarefied[,-1]))
colnames(Bact_nonrarefied) <- Bact_nonrarefied_trans$SampleID

Bact_nonrarefied_trans <- Bact_nonrarefied_trans %>%
  rownames_to_column(var = "SampleID")

### Read metadata ###
metadata <- read.csv(file="data/species_final_NONRAREFIED.sam.csv", header=T)
bact_metadata <- inner_join(metadata, Bact_nonrarefied_trans, by = "SampleID")

```

IndVal - Indicator Species Analysis

```

### Creating an object to store abundances only ###
nonrarefied_16S_abund <- bact_metadata %>% dplyr::select(-(1:4))

### Factors
host <-bact_metadata$sample_growth
class(host)
levels(host)

### Multipatt analysis: new and old growth, artificial and water ###
set.seed(333)
multipatt.seagrass_nonrarefied <- multipatt(nonrarefied_16S_abund, host,
  control = how(nperm=10000))

summary(multipatt.seagrass_nonrarefied)
multipatt_output <- capture.output(summary(multipatt.seagrass_nonrarefied, indvalcomp=TRUE))
write.csv(as.data.frame(multipatt_output), file = "data/INDVAL_core_16S.csv",
  row.names = F)

```

Data wrangle - prepare IndVal output to combine with taxonomy

```
### load output and wrangle data to separate sample types ###
indval_raw <- read.csv("data/INDVAL_core_16S.csv", sep=" ", header=TRUE)
names(indval_raw)

## [1] "multipatt_output"

# artificial #
#View(as.data.frame(indval_raw[c(11:212), ]))
indval_artificial <- as.data.frame(indval_raw[c(13:212), ])
colnames(indval_artificial)[1] <- "artificial"
indval_artificial$artificial <- gsub(' +', ' ', indval_artificial$artificial)
indval_artificial_split <- indval_artificial %>%
  separate(artificial, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_artificial_split$indval_group <- 'artificial'

# seawater #
#View(as.data.frame(indval_raw[c(214:396), ]))
indval_seawater <- as.data.frame(indval_raw[c(216:396), ])
colnames(indval_seawater)[1] <- "seawater"
indval_seawater$seawater <- gsub(' +', ' ', indval_seawater$seawater)
indval_seawater_split <- indval_seawater %>%
  separate(seawater, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_seawater_split$indval_group <- 'seawater'

# zostera_new #
#View(as.data.frame(indval_raw[c(397:470), ]))
indval_new_growth <- as.data.frame(indval_raw[c(399:469), ])
colnames(indval_new_growth)[1] <- "new_growth"
indval_new_growth$new_growth <- gsub(' +', ' ', indval_new_growth$new_growth)
indval_new_growth_split <- indval_new_growth %>%
  separate(new_growth, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_new_growth_split$indval_group <- 'zostera_new'

# zostera_old #
#View(as.data.frame(indval_raw[c(470:672), ]))
indval_old_growth <- as.data.frame(indval_raw[c(472:671), ])
colnames(indval_old_growth)[1] <- "old_growth"
indval_old_growth$old_growth <- gsub(' +', ' ', indval_old_growth$old_growth)
indval_old_growth_split <- indval_old_growth %>%
  separate(old_growth, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_old_growth_split$indval_group <- 'zostera_old'

# artificial+seawater #
#View(as.data.frame(indval_raw[c(673:706), ]))
indval_art_sea <- as.data.frame(indval_raw[c(675:706), ])
colnames(indval_art_sea)[1] <- "art_sea"
indval_art_sea$art_sea <- gsub(' +', ' ', indval_art_sea$art_sea)
indval_art_sea_split <- indval_art_sea %>%
  separate(art_sea, c("OTU", "A(specificity)", "B(fidelity)",
```

```

      "indval.stat", "p.value"), " ")
indval_art_sea_split$indval_group <- 'artificial+seawater'

# artificial+zostera_new #
#View(as.data.frame(indval_raw[c(707:720), ]))
indval_art_new <- as.data.frame(indval_raw[c(709:721), ])
colnames(indval_art_new)[1] <- "art_new"
indval_art_new$art_new <- gsub(' +', ' ', indval_art_new$art_new)
indval_art_new_split <- indval_art_new %>%
  separate(art_new, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_art_new_split$indval_group <- 'artificial+zostera_new'

# artificial+zostera_old (marine surfaces) #
#View(as.data.frame(indval_raw[c(721:879), ]))
indval_art_old <- as.data.frame(indval_raw[c(724:881), ])
colnames(indval_art_old)[1] <- "art_old"
indval_art_old$art_old <- gsub(' +', ' ', indval_art_old$art_old)
indval_art_old_split <- indval_art_old %>%
  separate(art_old, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_art_old_split$indval_group <- 'artificial+zostera_old'

# seawater+zostera_new #
#View(as.data.frame(indval_raw[c(880:884), ]))
indval_sea_new <- as.data.frame(indval_raw[c(884:886), ])
colnames(indval_sea_new)[1] <- "sea_new"
indval_sea_new$sea_new <- gsub(' +', ' ', indval_sea_new$sea_new)
indval_sea_new_split <- indval_sea_new %>%
  separate(sea_new, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_sea_new_split$indval_group <- 'seawater+zostera_new'

# seawater+zostera_old #
#View(as.data.frame(indval_raw[c(885:914), ]))
indval_sea_old <- as.data.frame(indval_raw[c(889:916), ])
colnames(indval_sea_old)[1] <- "sea_old"
indval_sea_old$sea_old <- gsub(' +', ' ', indval_sea_old$sea_old)
indval_sea_old_split <- indval_sea_old %>%
  separate(sea_old, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_sea_old_split$indval_group <- 'seawater+zostera_old'

# zostera_new+zostera_old (Zostera marina) #
#View(as.data.frame(indval_raw[c(915:944), ]))
indval_new_old <- as.data.frame(indval_raw[c(919:947), ])
colnames(indval_new_old)[1] <- "new_old"
indval_new_old$new_old <- gsub(' +', ' ', indval_new_old$new_old)
indval_new_old_split <- indval_new_old %>%
  separate(new_old, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_new_old_split$indval_group <- 'zostera_new+zostera_old'

```

```

# artificial+seawater+zostera_new #
#View(as.data.frame(indval_raw[c(945:949), ]))
indval_art_sea_new <- as.data.frame(indval_raw[c(950:952), ])
colnames(indval_art_sea_new)[1] <- "art_sea_new"
indval_art_sea_new$art_sea_new <- gsub(' +', ' ', indval_art_sea_new$art_sea_new)
indval_art_sea_new_split <- indval_art_sea_new %>%
  separate(art_sea_new, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_art_sea_new_split$indval_group <- 'artificial+seawater+zostera_new'

# artificial+seawater+zostera_old #
#View(as.data.frame(indval_raw[c(950:993), ]))
indval_art_sea_old <- as.data.frame(indval_raw[c(955:996), ])
colnames(indval_art_sea_old)[1] <- "art_sea_old"
indval_art_sea_old$art_sea_old <- gsub(' +', ' ', indval_art_sea_old$art_sea_old)
indval_art_sea_old_split <- indval_art_sea_old %>%
  separate(art_sea_old, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_art_sea_old_split$indval_group <- 'artificial+seawater+zostera_old'

# artificial+zostera_new+zostera_old #
#View(as.data.frame(indval_raw[c(994:1024), ]))
indval_art_new_old <- as.data.frame(indval_raw[c(999:1029), ])
colnames(indval_art_new_old)[1] <- "art_new_old"
indval_art_new_old$art_new_old <- gsub(' +', ' ', indval_art_new_old$art_new_old)
indval_art_new_old_split <- indval_art_new_old %>%
  separate(art_new_old, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_art_new_old_split$indval_group <- 'artificial+zostera_new+zostera_old'

# seawater+zostera_new+zostera_old #
#View(as.data.frame(indval_raw[c(1025:1028), ]))
indval_sea_new_old <- as.data.frame(indval_raw[c(1032:1033), ])
colnames(indval_sea_new_old)[1] <- "sea_new_old"
indval_sea_new_old$sea_new_old <- gsub(' +', ' ', indval_sea_new_old$sea_new_old)
indval_sea_new_old_split <- indval_sea_new_old %>%
  separate(sea_new_old, c("OTU", "A(specificity)", "B(fidelity)",
    "indval.stat", "p.value"), " ")
indval_sea_new_old_split$indval_group <- 'seawater+zostera_new+zostera_old'

indval_final_table <- as.data.frame(bind_rows(indval_artificial_split,
indval_seawater_split,
indval_new_growth_split,
indval_old_growth_split,
indval_art_sea_split,
indval_art_new_split,
indval_art_old_split,
indval_sea_new_split,
indval_sea_old_split,
indval_new_old_split,
indval_art_sea_new_split,
indval_art_sea_old_split,
indval_art_new_old_split,

```

```
indval_sea_new_old_split))
```

Combine IndVal and taxonomy information

```
### load tax table from saved phyloseq object ###
tax_table <- read.csv("data/species_final_NONRAREFIED.tax.csv", header=T)
tax_table$OTU <- as.character(tax_table$OTU)
### join indval_final_table and tax_table ###
indval_tax_table <- inner_join(indval_final_table, tax_table, by = "OTU")
```

Get relative abundance tables

```
phylo_merge_16S_nr <- readRDS("data/phylo_merge_not_rarefied_16S_paper.rds")

# get relative abundance table
phylo_merge_16S_nr_rel_abund <- transform_sample_counts(phylo_merge_16S_nr,
                                                         function(x) x/sum(x))

project_data.rel_abund.otu <- as.data.frame(otu_table(phylo_merge_16S_nr_rel_abund))
project_data.rel_abund.tax <- as.data.frame(tax_table(phylo_merge_16S_nr_rel_abund))
project_data.rel_abund.sam <- as.data.frame(sample_data(phylo_merge_16S_nr_rel_abund))

project_data.rel_abund.otu <- project_data.rel_abund.otu %>%
  rownames_to_column(var = "OTU")

project_data.rel_abund.tax <- project_data.rel_abund.tax %>%
  rownames_to_column(var = "OTU")

write.csv(project_data.rel_abund.otu, file="data/species_final_REL_ABUND.otu.csv",
          quote=F, row.names=F)

write.csv(project_data.rel_abund.tax, file="data/species_final_REL_ABUND.tax.csv",
          quote=F, row.names=F)

write.csv(project_data.rel_abund.sam, file="data/species_final_REL_ABUND.sam.csv",
          quote=F, row.names=F)

### load the otu table ###
Bact_rel_abund <- read.csv("data/species_final_REL_ABUND.otu.csv", header=T)

#### transpose it so taxa are columns and samples are rows, while keeping column names ###
Bact_rel_abund <- data.frame(Bact_rel_abund[,-1], row.names=Bact_rel_abund[,1])
Bact_rel_abund_trans <- as.data.frame(t(Bact_rel_abund[,-1]))
colnames(Bact_rel_abund) <- Bact_rel_abund_trans$SampleID
Bact_rel_abund_trans <- Bact_rel_abund_trans %>%
  rownames_to_column(var = "SampleID")

### load metadata ###
env <- read.csv(file="data/species_final_REL_ABUND.sam.csv", header=T)
```

```

### add metadata according to #SampleID labels ###
bact_env_relatabund <- inner_join(env, Bact_rel_abund_trans, by = "SampleID")

write.csv(bact_env_relatabund, file="data/Bact_metadata_REL_ABUND.csv",
          quote=F, row.names=F)

### Now get the mean relative abundance per region for each sample type ###
### (average relative abundance of all samples within that region for each habitat) ###

#### load relative abundance table ###
relat_abund_16S <- read.csv(file="data/Bact_metadata_REL_ABUND.csv", header=T)

### create host_region column and bring together with other metadata ###
relat_abund_16S <- relat_abund_16S %>% dplyr::mutate(
  host_region = paste(sample_growth, region, sep = "."))
relat_abund_16S <- relat_abund_16S %>%
  dplyr::select(host_region, everything())

### get mean relative abundances per region ###
relat_abund_16S_regions <- relat_abund_16S %>%
  dplyr::select(-c(2:5)) %>%
  group_by(host_region) %>%
  summarise_all(list(mean = mean))

### transpose it while keeping column names ###
final_relat_abund_16S <- as.data.frame(t(relat_abund_16S_regions[, -1]))
colnames(final_relat_abund_16S) <- relat_abund_16S_regions$host_region

final_relat_abund_16S <- final_relat_abund_16S %>%
  rownames_to_column(var = "OTU")

### remove extra labeling automatically added from OTU labels ###
### remove first character and replace with nothing ###
final_relat_abund_16S$OTU <- gsub("^.{0,1}", "", final_relat_abund_16S$OTU)
### remove last five characters and replace with nothing ###
final_relat_abund_16S$OTU <- gsub(".{0,5}$", "", final_relat_abund_16S$OTU)

```

Combine IndVal taxonomy table with relative abundance table

```

indval_tax_table$OTU <- as.character(indval_tax_table$OTU)
final_bubble_plot_table <- dplyr::inner_join(indval_tax_table,
                                              final_relat_abund_16S,
                                              by = "OTU")

### replace - with . so it doesn't interfere with knitting to pdf ###
final_bubble_plot_table$Rank5 <- gsub("-", ".", final_bubble_plot_table$Rank5)
final_bubble_plot_table$Rank6 <- gsub("-", ".", final_bubble_plot_table$Rank6)
final_bubble_plot_table$Rank7 <- gsub("-", ".", final_bubble_plot_table$Rank7)

### this is the final table to be used in the bubble plot script ###
write.csv(final_bubble_plot_table, file="data/IndVal_taxonomy_relat_abund.csv",
          quote=F, row.names=F)

```