

# 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

## Shared taxa and Venn Diagram

```
#### load packages ####
library(dplyr)
library(phyloseq)
library(tidyverse)
library(ggvenn)

#### importing files ####
phylo_merge_16S_nr <- readRDS("data/phylo_merge_not_rarefied_16S_paper.rds")
phylo_merge_16S_nr #1206 taxa

## 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 ]

#### subset taxa based on pres/abs threshold per sample group ####
# make presence absence table
project_data.shared <- phylo_merge_16S_nr # duplicate raw counts phyloseq object
otu <- as.data.frame(otu_table(project_data.shared)) #get OTU table
#set all positive values in OTU table of project_data.pres_abs to '1'
otu_table(project_data.shared)[otu >= 1] <- 1
```

## samples types with whole dataset *excluding artificial*

```
#### subset based on groups you're interested in (sample types) ####
seawater = subset_samples(project_data.shared, sample_growth == "seawater")
zostera_new = subset_samples(project_data.shared, sample_growth == "zostera_new")
zostera_old = subset_samples(project_data.shared, sample_growth == "zostera_old")
artificial = subset_samples(project_data.shared, sample_growth == "artificial")

#### remove all OTUs not found at threshold (N samples) ####
# do sums from presence absence OTU table
taxa_sums_sea <- as.data.frame(filter_taxa(seawater,
```

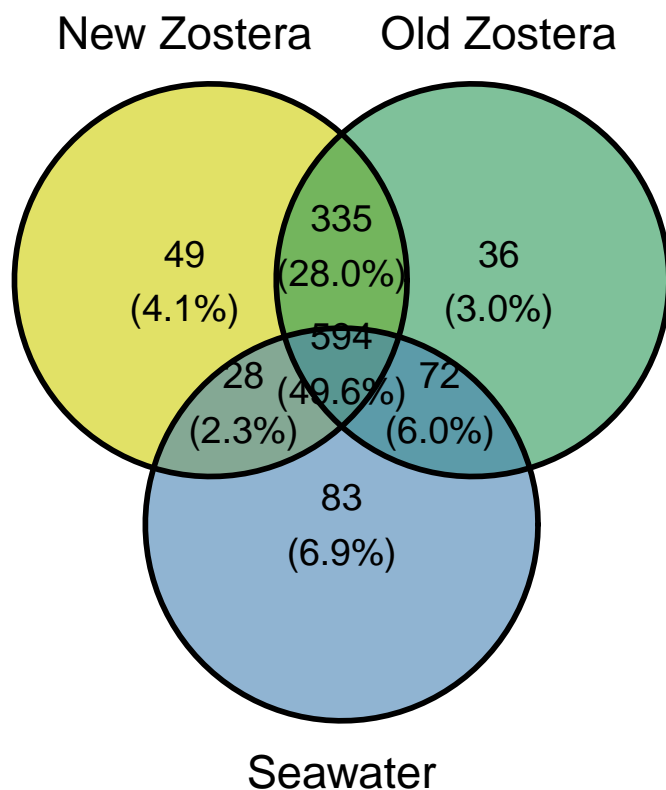
```

                                function(x) sum(x)))
# do sums from presence absence OTU table
taxa_sums_zos_new <- as.data.frame(filter_taxa(zostera_new,
                                function(x) sum(x)))
# do sums from presence absence OTU table
taxa_sums_zos_old <- as.data.frame(filter_taxa(zostera_old,
                                function(x) sum(x)))
# do sums from presence absence OTU table
taxa_sums_artificial<- as.data.frame(filter_taxa(artificial,
                                function(x) sum(x)))

#### select OTUs present in at least 2 samples ####
#select OTUs with sample count over your threshold
keep_sea <- row.names(
  taxa_sums_sea )[which(taxa_sums_sea [,1] >= 2)]
#select OTUs with sample count over your threshold
keep_zos_new <- row.names(
  taxa_sums_zos_new)[which(taxa_sums_zos_new[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_zos_old <- row.names(
  taxa_sums_zos_old)[which(taxa_sums_zos_old[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_artificial <- row.names(
  taxa_sums_artificial)[which(taxa_sums_artificial[,1] >= 2)]

#### Venn diagram ####
count = dplyr::count
all <- list("New Zostera"=keep_zos_new,
           "Old Zostera"=keep_zos_old,
           "Seawater"=keep_sea)
venn_all <- ggvenn(
  all,
  fill_color = c("yellow3", "#2a9958","steelblue"),
  fill_alpha = 0.6,
  stroke_color = "black",
  stroke_alpha = 0.7,
  stroke_size = 1,
  stroke_linetype = "solid",
  set_name_color = "black",
  set_name_size = 6,
  text_color = "black",
  text_size = 5
)
venn_all

```



samples types with choked dataset *including artificial*

```
### Select data from Choked region to compare to artificial seagrass ###
phylo_choked <- subset_samples(project_data.shared, region=="choked")
### Remove any OTUs that are absent from Choked samples ###
phylo_choked <- prune_taxa((taxa_sums(phylo_choked) > 0), phylo_choked)

#### subset based on groups you're interested in (sample types) ####
seawater_choked = subset_samples(phylo_choked, sample_growth == "seawater")
zostera_new_choked = subset_samples(phylo_choked, sample_growth == "zostera_new")
zostera_old_choked = subset_samples(phylo_choked, sample_growth == "zostera_old")
artificial_choked = subset_samples(phylo_choked, sample_growth == "artificial")

#### remove all OTUs not found at threshold (N samples) ####
# do sums from presence absence OTU table
taxa_sums_sea_choked <- as.data.frame(filter_taxa(seawater_choked,
                                                  function(x) sum(x)))

# do sums from presence absence OTU table
taxa_sums_zos_new_choked <- as.data.frame(filter_taxa(zostera_new_choked,
                                                       function(x) sum(x)))

# do sums from presence absence OTU table
taxa_sums_zos_old_choked <- as.data.frame(filter_taxa(zostera_old_choked,
                                                       function(x) sum(x)))

# do sums from presence absence OTU table
taxa_sums_artificial_choked <- as.data.frame(filter_taxa(artificial_choked,
                                                         function(x) sum(x)))

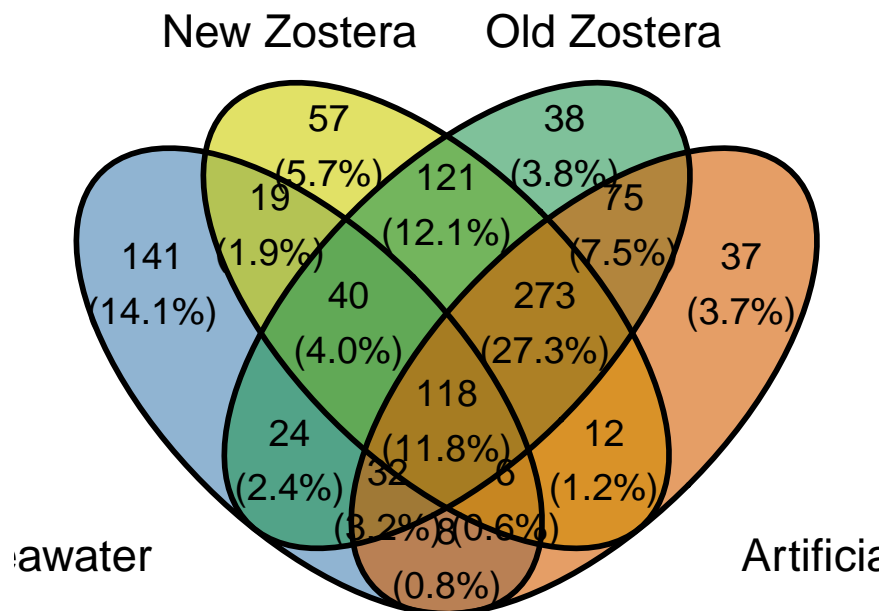
### select OTUs present in at least 2 samples ###
#select OTUs with sample count over your threshold
```

```

keep_sea_choked <- row.names(
  taxa_sums_sea_choked)[which(taxa_sums_sea_choked[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_zos_new_choked <- row.names(
  taxa_sums_zos_new_choked)[which(taxa_sums_zos_new_choked[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_zos_old_choked <- row.names(
  taxa_sums_zos_old_choked)[which(taxa_sums_zos_old_choked[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_artificial_choked <- row.names(
  taxa_sums_artificial_choked)[which(taxa_sums_artificial_choked[,1] >= 2)]

### Venn diagram ###
choked <- list("Seawater"=keep_sea_choked,
              "New Zostera"=keep_zos_new_choked,
              "Old Zostera"=keep_zos_old_choked,
              "Artificial"=keep_artificial_choked)
venn_choked <- ggvenn(
  choked,
  fill_color = c("steelblue", "yellow3", "#2a9958", "#D55E00"),
  fill_alpha = 0.6,
  stroke_color = "black",
  stroke_alpha = 0.7,
  stroke_size = 1,
  stroke_linetype = "solid",
  set_name_color = "black",
  set_name_size = 6,
  text_color = "black",
  text_size = 5
)
venn_choked

```



## regions for seawater

```
### Select seawater data ###
project_data_water = subset_samples(project_data.shared,
                                     sample_growth == "seawater")

### subset based on groups you're interested in (regions) ###
water_groupC = subset_samples(project_data_water, region == "choked")
water_groupG = subset_samples(project_data_water, region == "goose")
water_groupM = subset_samples(project_data_water, region == "mcmullin")
water_groupT = subset_samples(project_data_water, region == "triquet")

### remove all OTUs not found at threshold (N samples) ###
# do sums from presence absence OTU table
water_taxa_sums_grC <- as.data.frame(filter_taxa(water_groupC,
                                                  function(x) sum(x)))

# do sums from presence absence OTU table
water_taxa_sums_grG <- as.data.frame(filter_taxa(water_groupG,
                                                  function(x) sum(x)))

# do sums from presence absence OTU table
water_taxa_sums_grM <- as.data.frame(filter_taxa(water_groupM,
                                                  function(x) sum(x)))

# do sums from presence absence OTU table
water_taxa_sums_grT <- as.data.frame(filter_taxa(water_groupT,
                                                  function(x) sum(x)))

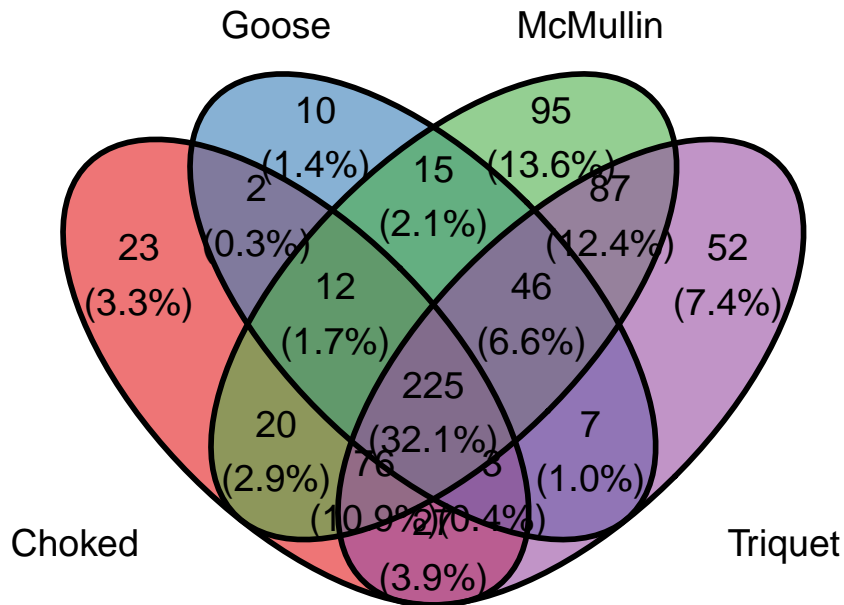
### select OTUs present in at least 2 samples ###
#select OTUs with sample count over your threshold
keep_water_C <- row.names(
  water_taxa_sums_grC[which(water_taxa_sums_grC[,1] >= 2)])
#select OTUs with sample count over your threshold
keep_water_G <- row.names(
  water_taxa_sums_grG[which(water_taxa_sums_grG[,1] >= 2)])
#select OTUs with sample count over your threshold
keep_water_M <- row.names(
  water_taxa_sums_grM[which(water_taxa_sums_grM[,1] >= 2)])
#select OTUs with sample count over your threshold
keep_water_T <- row.names(
  water_taxa_sums_grT[which(water_taxa_sums_grT[,1] >= 2)])

### Venn Diagram ###
sea_regions <- list("Choked"=keep_water_C,
                   "Goose"=keep_water_G,
                   "McMullin"=keep_water_M,
                   "Triquet"=keep_water_T)
venn_water <- ggvenn(
  sea_regions,
  fill_color = c("#e41a1c", "#377eb8", "#4daf4a", "#984ea3"),
  fill_alpha = 0.6,
  stroke_color = "black",
  stroke_alpha = 0.7,
  stroke_size = 1,
  stroke_linetype = "solid",
  set_name_color = "black",
```

```

set_name_size = 5,
text_color = "black",
text_size = 5
)
venn_water

```



regions for new growth leaves

```

### Select zostera new data ###
project_data_zostera_new = subset_samples(project_data.shared,
                                           sample_growth == "zostera_new")
project_data_zostera_new

## phyloseq-class experiment-level object
## otu_table() OTU Table: [ 1206 taxa and 54 samples ]
## sample_data() Sample Data: [ 54 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 ]

### subset based on groups you're interested in (regions) ###
zostera_new_groupC = subset_samples(project_data_zostera_new, region == "choked")
zostera_new_groupG = subset_samples(project_data_zostera_new, region == "goose")
zostera_new_groupM = subset_samples(project_data_zostera_new, region == "mcmullin")
zostera_new_groupT = subset_samples(project_data_zostera_new, region == "triquet")

### remove all OTUs not found at threshold (N samples) ###
# do sums from presence absence OTU table
zostera_new_taxa_sums_grC <- as.data.frame(filter_taxa(zostera_new_groupC,
                                                       function(x) sum(x)))

# do sums from presence absence OTU table
zostera_new_taxa_sums_grG <- as.data.frame(filter_taxa(zostera_new_groupG,
                                                       function(x) sum(x)))

# do sums from presence absence OTU table
zostera_new_taxa_sums_grM <- as.data.frame(filter_taxa(zostera_new_groupM,

```

```

function(x) sum(x))

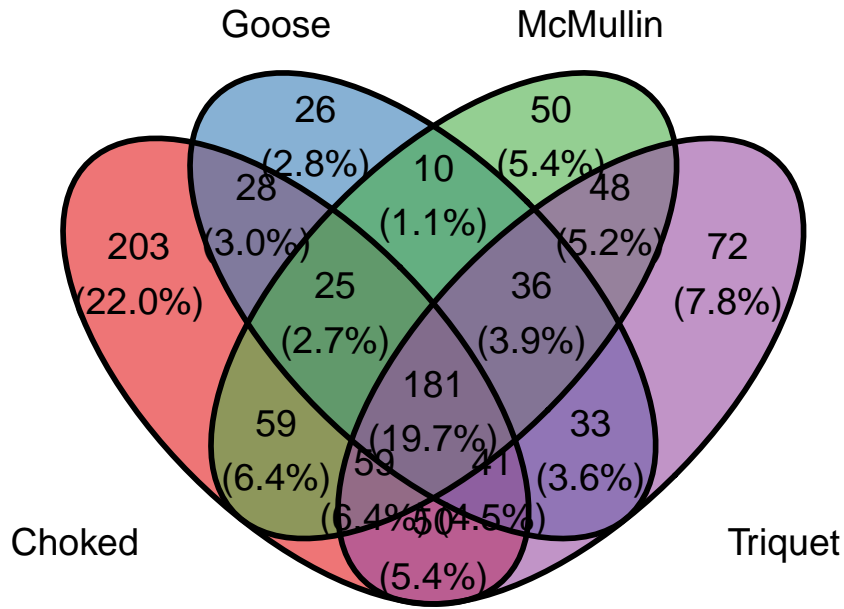
# do sums from presence absence OTU table
zostera_new_taxa_sums_grT <- as.data.frame(filter_taxa(zostera_new_groupT,
function(x) sum(x)))

### select OTUs present in at least 2 samples ###
#select OTUs with sample count over your threshold
keep_new_zostera_C <- row.names(
  zostera_new_taxa_sums_grC)[which(zostera_new_taxa_sums_grC[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_new_zostera_G <- row.names(
  zostera_new_taxa_sums_grG)[which(zostera_new_taxa_sums_grG[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_new_zostera_M <- row.names(
  zostera_new_taxa_sums_grM)[which(zostera_new_taxa_sums_grM[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_new_zostera_T <- row.names(
  zostera_new_taxa_sums_grT)[which(zostera_new_taxa_sums_grT[,1] >= 2)]

new_regions <- list("Choked"=keep_new_zostera_C,
                    "Goose"=keep_new_zostera_G,
                    "McMullin"=keep_new_zostera_M,
                    "Triquet"=keep_new_zostera_T)

venn_new <- ggvenn(
  new_regions,
  fill_color = c("#e41a1c", "#377eb8", "#4daf4a", "#984ea3"),
  fill_alpha = 0.6,
  stroke_color = "black",
  stroke_alpha = 0.7,
  stroke_size = 1,
  stroke_linetype = "solid",
  set_name_color = "black",
  set_name_size = 5,
  text_color = "black",
  text_size = 5
)
venn_new

```



regions for old growth leaves

```
### Select zostera old data ###
project_data_zostera_old = subset_samples(project_data.shared,
                                           sample_growth == "zostera_old")
project_data_zostera_old

## phyloseq-class experiment-level object
## otu_table() OTU Table: [ 1206 taxa and 51 samples ]
## sample_data() Sample Data: [ 51 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 ]

### subset based on groups you're interested in (regions) ###
zostera_old_groupC = subset_samples(project_data_zostera_old, region == "choked")
zostera_old_groupG = subset_samples(project_data_zostera_old, region == "goose")
zostera_old_groupM = subset_samples(project_data_zostera_old, region == "mcmullin")
zostera_old_groupT = subset_samples(project_data_zostera_old, region == "triquet")

### remove all OTUs not found at threshold (N samples) ###
# do sums from presence absence OTU table
zostera_old_taxa_sums_grC <- as.data.frame(filter_taxa(zostera_old_groupC,
                                                       function(x) sum(x)))

# do sums from presence absence OTU table
zostera_old_taxa_sums_grG <- as.data.frame(filter_taxa(zostera_old_groupG,
                                                       function(x) sum(x)))

# do sums from presence absence OTU table
zostera_old_taxa_sums_grM <- as.data.frame(filter_taxa(zostera_old_groupM,
                                                       function(x) sum(x)))

# do sums from presence absence OTU table
zostera_old_taxa_sums_grT <- as.data.frame(filter_taxa(zostera_old_groupT,
                                                       function(x) sum(x)))

### select OTUs present in at least 2 samples ###
```



```

#select OTUs with sample count over your threshold
keep_old_zostera_C <- row.names(
  zostera_old_taxa_sums_grC)[which(zostera_old_taxa_sums_grC[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_old_zostera_G <- row.names(
  zostera_old_taxa_sums_grG)[which(zostera_old_taxa_sums_grG[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_old_zostera_M <- row.names(
  zostera_old_taxa_sums_grM)[which(zostera_old_taxa_sums_grM[,1] >= 2)]
#select OTUs with sample count over your threshold
keep_old_zostera_T <- row.names(
  zostera_old_taxa_sums_grT)[which(zostera_old_taxa_sums_grT[,1] >= 2)]

old_regions <- list("Choked"=keep_old_zostera_C,
                    "Goose"=keep_old_zostera_G,
                    "McMullin"=keep_old_zostera_M,
                    "Triquet"=keep_old_zostera_T)

venn_old <- ggvenn(
  old_regions,
  fill_color = c("#e41a1c", "#377eb8", "#4daf4a", "#984ea3"),
  fill_alpha = 0.6,
  stroke_color = "black",
  stroke_alpha = 0.7,
  stroke_size = 1,
  stroke_linetype = "solid",
  set_name_color = "black",
  set_name_size = 5,
  text_color = "black",
  text_size = 5
)
venn_old

```

