Illumina data, Fungi: Q0 Summarize diversity

Marissa Lee

5/26/2020

Table of contents

Load packages, paths

A. Load data and pre-process ASV matrix

1. Load phyloseq object from Illum_makeASVmatrix.Rmd, add environmental data [commented out]

```
ps <- readRDS(file = file.path(merged_path, "phyloseq_samps.RData"))
ps # 3811 ASVs
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 3811 taxa and 332 samples ]
## sample_data() Sample Data:       [ 332 samples by 28 sample variables ]
## tax_table()   Taxonomy Table:    [ 3811 taxa by 9 taxonomic ranks ]
## refseq()      DNAStringSet:      [ 3811 reference sequences ]
```

```
# ps.df <- data.frame(sample_data(ps))
# dim(ps.df)
```

```
#
# # load site data
# path1 <- "data_intermediates/dataCleaningProducts/DOE-NC-FIELD_SiteData_Rcompiled.xlsx"
# siteData<- read_excel(path1, sheet = "data")
# colnames(siteData)
#
# # load sample data
# path2 <- "data_intermediates/dataCleaningProducts/DOE-NC-FIELD_SampleData_Rcompiled.xlsx"
# sampleData <- read_excel(path2, sheet = "data")
# colnames(sampleData)
# # # combine sequence sample matrix, site data, and sample data into 1 df
# dim(ps.df); dim(siteData); dim(sampleData)
# ps.df %>%
#   select(sample.name.match, sample.type, SiteSamp, Site, Tissue) %>%
#   left_join(siteData) %>%
#   left_join(sampleData) -> sampleMat
# dim(sampleMat)
# colnames(sampleMat)
# row.names(sampleMat) <- sampleMat$sample.name.match
#
# # replace sample data in phyloseq object
# sample_data(ps) <- sampleMat
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env.RData"))
```

2. Examine the occupancy and abundance distributions to determine ASVs to drop [commented out]

```
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env.RData"))
# ps
#
# # first, remove taxa that only show up in 1 sample
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# ps <- subset_taxa(ps, taxa_sums(ps.pa) > 1) # remove singletons
# ps # down to ~3K asvs
#
# abund <- taxa_sums(ps)
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# occ <- taxa_sums(ps.pa)
# df <- data.frame(asv = names(abund), abund, occ)
#
# hist(abund[abund<100], breaks = 20)
# hist(occ[occ<10], breaks = 10)
# occ.cutoff <- 4 # must be present in at least X samples (e.g. not an entire site)
# #abund.cutoff <- 40
#
# p.all <- ggplot(df, aes(x = log(occ), y = log(abund))) +
#   geom_point() +
#   geom_vline(xintercept = log(occ.cutoff), linetype = 2) +
#   xlab("Log occupancy") +
#   ylab("Log abundance") +
#   theme_bw() +
```

```
#   ggtitle("All tissues")
# p.all
# #
# # trim with these parameters
# ps <- subset_taxa(ps, taxa_sums(ps.pa) > occ.cutoff) # remove based on sample occupancy
# #ps <- subset_taxa(ps, taxa_sums(ps) > abund.cutoff) # remove based on reads
# ps
# #
# # ### Write the cleaned phyloseq object and attributes
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimASVs.RData"))
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs.RData"))
# ps
```

3. Create a phylogenetic tree from all ASV representative sequences.

1 - Prep taxonomy file for the perl script with all ASVs

```
#library(phyloseq)
#library(speedyseq)
ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs.RData"))
tax <- data.frame(tax_table(ps), stringsAsFactors = F)
tax %>%
  mutate(tip.label = ASV) %>%
  select(-ASV) -> asv.tax
asv.tax %>%
  group_by(phylum) %>%
  summarize(n = length(tip.label))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 11 x 2
##    phylum                   n
##    <chr>                <int>
##  1 p__Ascomycota          273
##  2 p__Basidiomycota       248
##  3 p__Blastocladiomycota    2
##  4 p__Chytridiomycota      67
##  5 p__Entorrhizomycota      1
##  6 p__Glomeromycota       253
##  7 p__Kickxellomycota       1
##  8 p__Mortierellomycota    26
##  9 p__Mucoromycota          9
## 10 p__Rozellomycota        29
## 11 <NA>                    23
```

```
# write.table(asv.tax, file = "data_intermediates/phylogeneticTree/asv_tax.txt",
#             row.names = FALSE,
#             col.names = FALSE, sep = "\t")
```

```
#perl taxonomy_to_tree.pl -h
#perl taxonomy_to_tree.pl -f asv_tax.txt > asv_tax.tre
```

The format of asv_tax.tre is odd, so open it in FigTree and rename to asv_tax_formated.tre

Load the tree and update big phyloseq objects, all asvs

```
ted.tree <- ape::read.nexus(file = "data_intermediates/phylogeneticTree/asv_tax_formated.tre")
ted.tree
```

```
##
## Phylogenetic tree with 1417 tips and 1430 internal nodes.
##
## Tip labels:
##  ASV_6764, ASV_3509, ASV_540, ASV_3014, ASV_1733, ASV_551, ...
##
## Rooted; includes branch lengths.
```

```
# ted.tree # this one is rooted as expected
# #ape::is.binary.phylo(ted.tree)

# cop.mat <- cophenetic(ted.tree)
# # example of ASVs that are all unknown o__Pleosporales
# cop.mat["ASV_593","ASV_1129"] # 120
# cop.mat["ASV_593","ASV_4035"] # 120
# # example of unknown o__Pleosporales to s__cladoniicola (within o__Pleosporales)
# cop.mat["ASV_593","ASV_935"] # 480
# # example of unknown o__Pleosporales to s__dioscoreae (within o__Pleosporales)
# cop.mat["ASV_593","ASV_667"] # 480
# # example of unknown o__Pleosporales to unknown at c_Doth ASV_2132
# cop.mat["ASV_593","ASV_2132"] # 600
# # example of s__cladoniicola (within o__Pleosporales) to unknown at c_Doth ASV_2132
# cop.mat["ASV_935","ASV_2132"] # 600

# branch lengths between all taxonomic levels is 60
# unknowns in a given phylum have the same total branch length as knowns


# # library(ggtree)
# # library(ape)
# #
# # all
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs.RData"))
# asvs <- taxa_names(ps)
# tree <- keep.tip(ted.tree, asvs)
# phy_tree(ps) <- tree
# p <- ggtree(ps, ladderize = T, aes(color = phylum)) +
#   geom_tiplab(size = 1) +
#   theme(legend.position = "right")
# pdf(file.path(out_path, "taxonomy_tree_allasvs.pdf"), width = 10, height = 20)
# p
# dev.off()
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
# #
```

B. Best way to capture variation in community composition in 1 dimension? [commented out, but see output/illumina/Q0/approach_considerat

1. Compare how different ASV transformations and ASV occupancy cut-offs influence the % of variation explained by ordination axis 1

Set up the tissue subset data

```
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs.RData"))
#
# ps.l <- prune_samples(grepl("L",sample_names(ps)), ps)
# ps.l <- prune_taxa(colSums(otu_table(ps.l)) != 0, ps.l)
# #saveRDS(ps.l, file = file.path(merged_path, "phyloseq_samps_env_trimASVs_leaf.RData"))
# #
#
# ps.r <- prune_samples(grepl("R",sample_names(ps)), ps)
# ps.r <- prune_taxa(colSums(otu_table(ps.r)) != 0, ps.r)
# #saveRDS(ps.r, file = file.path(merged_path, "phyloseq_samps_env_trimASVs_root.RData"))
# #
# ps.s <- prune_samples(grepl("S",sample_names(ps)), ps)
# ps.s <- prune_taxa(colSums(otu_table(ps.s)) != 0, ps.s)
# #saveRDS(ps.s, file = file.path(merged_path, "phyloseq_samps_env_trimASVs_soil.RData"))
```

CLR

```
# require(compositions)
# vec.occ <- c(4, 6, 8, 10)
# TISSUE <- c("L","R","S")
# prop.expln.list <- list()
# df.expln.list <- list()
# for(k in 1:length(TISSUE)){
#   for(i in 1:length(vec.occ)){
#
#   if(TISSUE[k] == "L"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_leaf.RData"))
#   }
#   if(TISSUE[k] == "R"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_root.RData"))
#   }
#   if(TISSUE[k] == "S"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_soil.RData"))
#   }
#
#   # trim OTU table
#   ps # 301 ASVs
#   ps.pa <- ps
#   otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
#   ps <- subset_taxa(ps, taxa_sums(ps.pa) > vec.occ[i]) # remove based on sample occupancy
#   ps
#
#   # calc clr
#   asv <- otu_table(ps)
#   asv_clr <- data.frame(clr(asv))
#
#   # do ordination
```

```
#   mod <- capscale(asv_clr~1, distance = "euclidean")
#   mod.summ <- summary(mod)
#   prop.expln.list[[i]] <- mod.summ$cont$importance[2,1:5]
#   #screeplot(mod)
# }
# df.expln <- list_to_df(prop.expln.list)
# df.expln$occ <- vec.occ
# df.expln.list[[k]] <- df.expln
#
# }
# names(df.expln.list) <- TISSUE
# df.expln.list
# df.expln.clr <- list_to_df(df.expln.list)
```

VST

```
# library(DESeq2)
# # note that for estimateSizeFactors, need to use
# gm_mean = function(x, na.rm=TRUE){ exp(sum(log(x[x > 0]), na.rm=na.rm) / length(x))}
#
# vec.occ <- c(4, 6, 8, 10)
# TISSUE <- c("L","R","S")
# prop.expln.list <- list()
# df.expln.list <- list()
# for(k in 1:length(TISSUE)){
#   for(i in 1:length(vec.occ)){
#
#   if(TISSUE[k] == "L"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_leaf.RData"))
#   }
#   if(TISSUE[k] == "R"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_root.RData"))
#   }
#   if(TISSUE[k] == "S"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_soil.RData"))
#   }
#
#   # trim OTU table
#   ps # 301 ASVs
#   ps.pa <- ps
#   otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
#   ps <- subset_taxa(ps, taxa_sums(ps.pa) > vec.occ[i]) # remove based on sample occupancy
#   ps
#
#   # calc vst
#   ps_ds <- phyloseq_to_deseq2(ps, ~1) # convert phyloseq to DeSeq object
#   geoMeans = apply(counts(ps_ds), 1, gm_mean) # calc geometric mean of each ASV
#   ps_ds = estimateSizeFactors(ps_ds, type="ratio", geoMeans = geoMeans)
#   ps_ds = estimateDispersions(ps_ds, fitType = "parametric")
#   #plotDispEsts(ps_ds) # plot the dispersion estimates
#   vst <- getVarianceStabilizedData(ps_ds)
#   vst <- t(vst) # need to make the rows samples
#   #colnames(vst)
#
```

```
#   # do ordination
#   mod <- capscale(vst~1, distance = "euclidean")
#   mod.summ <- summary(mod)
#   prop.expln.list[[i]] <- mod.summ$cont$importance[2,1:5]
#   #screeplot(mod)
# }
# df.expln <- list_to_df(prop.expln.list)
# df.expln$occ <- vec.occ
# df.expln.list[[k]] <- df.expln
#
# }
# names(df.expln.list) <- TISSUE
# df.expln.list
# df.expln.vst <- list_to_df(df.expln.list)
```

PhIRL

```
# tree <- ape::read.nexus(file = "data_intermediates/phylogeneticTree/asv_tax_formated.tre")
# library(ape)
# library(philr)
# # tree$tip.label
# # is.binary.phylo(tree)
# # is.rooted.phylo(tree)
# #
# vec.occ <- c(4, 6, 8, 10)
# TISSUE <- c("L","R","S")
# prop.expln.list <- list()
# df.expln.list <- list()
# for(k in 1:length(TISSUE)){
#   for(i in 1:length(vec.occ)){
#
#   if(TISSUE[k] == "L"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_leaf.RData"))
#   }
#   if(TISSUE[k] == "R"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_root.RData"))
#   }
#   if(TISSUE[k] == "S"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_soil.RData"))
#   }
#
#   # trim OTU table
#   ps # 301 ASVs
#   ps.pa <- ps
#   otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
#   ps <- subset_taxa(ps, taxa_sums(ps.pa) > vec.occ[i]) # remove based on sample occupancy
#   ps
#
#   # calc philr
#   asv <- otu_table(ps)
#   asv <- data.frame(asv)
#   # add 1 to everything
#   asv.one <- asv + 1
#   asv.one.mat <- as.matrix(asv.one) # philr requires a matrix
```

```
#   # make sure the ASVs and the tree tips match
#   pruned.tree <- keep.tip(tree, tip = colnames(asv.one))
#   is.binary(pruned.tree)
#   asv_philr <- philr(df = asv.one.mat, tr = pruned.tree, part.weights='enorm.x.gm.counts') # do the t
#
#   # do ordination
#   mod <- capscale(asv_philr~1, distance = "euclidean")
#   mod.summ <- summary(mod)
#   prop.expln.list[[i]] <- mod.summ$cont$importance[2,1:5]
#   #screeplot(mod)
# }
# df.expln <- list_to_df(prop.expln.list)
# df.expln$occ <- vec.occ
# df.expln.list[[k]] <- df.expln
#
# }
# names(df.expln.list) <- TISSUE
# df.expln.list
# df.expln.philr <- list_to_df(df.expln.list)
```

DPCOA

```
# vec.occ <- c(4, 6, 8, 10)
# TISSUE <- c("L","R","S")
# prop.expln.list <- list()
# df.expln.list <- list()
# for(k in 1:length(TISSUE)){
#   for(i in 1:length(vec.occ)){
#
#   if(TISSUE[k] == "L"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_leaf.RData"))
#   }
#   if(TISSUE[k] == "R"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_root.RData"))
#   }
#   if(TISSUE[k] == "S"){
#     ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs_soil.RData"))
#   }
#
#   # trim OTU table
#   ps # 301 ASVs
#   ps.pa <- ps
#   otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
#   ps <- subset_taxa(ps, taxa_sums(ps.pa) > vec.occ[i]) # remove based on sample occupancy
#   ps
#
#   # calc DPCOA
#   pruned.tree <- keep.tip(tree, tip = taxa_names(ps))
#   phy_tree(ps) <- phy_tree(pruned.tree)
#   mod <- DPCoA(ps, correction = cailliez)
#   #plot_ordination(ps, mod, "biplot")
#   prop.expln.list[[i]] <- mod$eig[1:5] / sum(mod$eig)
#
# }
```

```
# df.expln <- list_to_df(prop.expln.list)
# df.expln$occ <- vec.occ
# df.expln.list[[k]] <- df.expln
#
# }
# names(df.expln.list) <- TISSUE
# df.expln.list
# df.expln.dpcoa <- list_to_df(df.expln.list)
```

Plot: varexpln_sensitivity.pdf

```
# df.expln.clr$transform <- "clr"
# df.expln.vst$transform <- "vst"
# df.expln.philr$transform <- "philr"
# df.expln.dpcoa$transform <- "dpcoa"
# colnames(df.expln.dpcoa)[1:5] <- colnames(df.expln.clr)[1:5]
# df.expln.clr %>%
#   rbind(df.expln.vst) %>%
#   rbind(df.expln.philr) %>%
#   rbind(df.expln.dpcoa) %>%
#   dplyr::rename('Tissue'='source') %>%
#   gather(key = "component", value = "value", -c(occ, Tissue, transform)) -> df.expln.l
#
# df.expln.l %>%
#   filter(component == "MDS1") -> df.tmp
# ggplot(df.tmp, aes(x = occ, y = (value*100), color = transform)) +
#   geom_point() +
#   geom_line() +
#   facet_grid(~Tissue) +
#   theme_bw() +
#   xlab("Minimum ASV occupancy") +
#   ylab("PC1 variance explained (%)")
# ggsave(filename = file.path(out_path, "varexpln_sensitivity.pdf"),
#        width = 6, height = 3.5)
```

Use DPCoA and go with a cutoff of 6 occurrances Reference for DPCoA: Pavoine 2004. From dissimilarities among species to dissimilarities among communities: a double principal coordinate analysis. https://doi.org/10.1016/j.jtbi.2004.02.014

2. Update occupancy cut off. Also trim the phylotree and put it into the phyloseq object [commented out]

All

```
# occ.cutoff <- 6 # must be present in at least X samples (e.g. not an entire site)
# #abund.cutoff <- 10 # must have at least X reads
#
# # All
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env.RData"))
# ps

# otumat <- as.matrix(otu_table(ps))
# require(vegan)
# png(file = file.path(out_path, "rarecurve_all.png"), width=500, height=500)
# rarecurve(otumat,
```

```
#           step=100,
#           xlab="Number of reads per sample",
#           ylab="Cumulative number of ASVs", label=TRUE)
# dev.off()

#
# # remove taxa that only show up in 1 sample
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# ps <- subset_taxa(ps, taxa_sums(ps.pa) > 1) # remove singletons
# ps # down to 3810 taxa
#
# # plot
# abund <- taxa_sums(ps)
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# occ <- taxa_sums(ps.pa)
# df <- data.frame(asv = names(abund), abund, occ)
# p.all <- ggplot(df, aes(x = log(occ), y = log(abund))) +
#   geom_point() +
#   geom_vline(xintercept = log(occ.cutoff), linetype = 2) +
#   #geom_hline(yintercept = log(abund.cutoff), linetype = 2) +
#   xlab("Log occupancy") +
#   ylab("Log abundance") +
#   theme_bw() +
#   ggtitle("All")
# p.all
#
# # trim with these parameters
# ps <- subset_taxa(ps, taxa_sums(ps.pa) > occ.cutoff) # remove based on sample occupancy
# #ps <- subset_taxa(ps, taxa_sums(ps) > abund.cutoff) # remove based on reads
# ps # down to 932 asvs
# # check for empty samples
# sum(sample_sums(ps) == 0)
#
# # write the cleaned phyloseq object and attributes
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimASVs.RData"))
#
# # add the phylo tree again
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimASVs.RData"))
# asvs <- taxa_names(ps)
# tree <- ape::read.nexus(file = "data_intermediates/phylogeneticTree/asv_tax_formated.tre")
# tree <- keep.tip(tree, asvs)
# phy_tree(ps) <- tree
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
```

Leaf

```
# occ.cutoff <- 6 # must be present in at least X samples (e.g. not an entire site)
# #abund.cutoff <- 40 # must have at least X reads
#
# #leaf
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData")) #ASV table
# ps <- subset_samples(ps, Tissue == "L")
```

```
#
# # remove taxa that only show up in 1 sample
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# ps <- subset_taxa(ps, taxa_sums(ps.pa) > 1) # remove singletons
# ps # down to 203 taxa
# # trim with these parameters
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# ps.trim <- subset_taxa(ps, taxa_sums(ps.pa) > occ.cutoff &
#                      taxa_sums(ps) > abund.cutoff) # remove based on sample occupancy
# ps.trim # down to 185 asvs
# ps <- ps.trim
#
# # check for empty samples
# sum(sample_sums(ps) == 0)
#
# ps
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_leaf.RData"))
```

Root

```
# occ.cutoff <- 6 # must be present in at least X samples (e.g. not an entire site)
# #abund.cutoff <- 40 # must have at least X reads
#
# #root
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData")) #untrimmed ASV
# ps <- subset_samples(ps, Tissue == "R")
# ps
#
# # remove taxa that only show up in 1 sample
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# ps <- subset_taxa(ps, taxa_sums(ps.pa) > 1) # remove singletons
# ps # down to 607 taxa
# # trim with these parameters
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# ps.trim <- subset_taxa(ps, taxa_sums(ps.pa) > occ.cutoff &
#                      taxa_sums(ps) > abund.cutoff) # remove based on sample occupancy
# ps.trim # down to 187 asvs
# ps <- ps.trim
#
# # check for empty samples
# sum(sample_sums(ps) == 0)
#
# ps
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_root.RData"))
```

Soil

```
# occ.cutoff <- 6 # must be present in at least X samples (e.g. not an entire site)
# #abund.cutoff <- 40 # must have at least X reads
#
# #soil
```

```
# ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData")) #untrimmed ASV
# ps <- subset_samples(ps, Tissue == "S")
# ps
#
# # remove taxa that only show up in 1 sample
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# ps <- subset_taxa(ps, taxa_sums(ps.pa) > 1) # remove singletons
# ps # down to 752 taxa
# # trim with these parameters
# ps.pa <- ps # make a presence/absence obj
# otu_table(ps.pa) <- (otu_table(ps.pa) > 0)*1
# ps.trim <- subset_taxa(ps, taxa_sums(ps.pa) > occ.cutoff &
#                        taxa_sums(ps) > abund.cutoff) # remove based on sample occupancy
# ps.trim # down to 441 asvs
# ps <- ps.trim
#
# # check for empty samples
# sum(sample_sums(ps) == 0)
#
# # check for singletons
# ps.trim.pa <- ps.trim
# otu_table(ps.trim.pa) <- (otu_table(ps.trim.pa) > 0)*1
# sum(taxa_sums(otu_table(ps.trim.pa)) < 4)
#
# ps
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_soil.RData"))
```

C. Summarize diversity

1. How many unknowns at each taxonomic level?

```
ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
ps
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 932 taxa and 332 samples ]
## sample_data() Sample Data:       [ 332 samples by 75 sample variables ]
## tax_table()   Taxonomy Table:    [ 932 taxa by 9 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 932 tips and 254 internal nodes ]
## refseq()      DNAStringSet:      [ 932 reference sequences ]
```

```
# how many reads/sample? by tissue
# asv <- otu_table(ps)
# readssamp <- data.frame(sample.name.match = row.names(asv),
#                         reads = rowSums(asv))
# sam <- data.frame(sample_data(ps), stringsAsFactors = F)
# sam %>%
#   left_join(readssamp) -> sam
# sam %>%
#   group_by(Tissue) %>%
#   summarize(mean = mean(reads),
```

```
#             sd = sd(reads),
#             se = sd/sqrt(length(reads))) -> tmp
# tmp
# ggplot(tmp, aes(x = Tissue, y = mean)) +
#   geom_point() +
#   geom_errorbar(aes(ymin = mean - se, ymax = mean + se)) +
#   theme_classic() +
#   xlab("Within-plant habitat") +
#   ylab("Reads per sample") +
#   ylim(c(0, 18000))
# ggsave(filename = file.path(out_path, "readsPersample.png"), width = 3, height = 3)


tax.df <- data.frame(tax_table(ps), stringsAsFactors = F)
ps
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 932 taxa and 332 samples ]
## sample_data() Sample Data:       [ 332 samples by 75 sample variables ]
## tax_table()   Taxonomy Table:    [ 932 taxa by 9 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 932 tips and 254 internal nodes ]
## refseq()      DNAStringSet:      [ 932 reference sequences ]
```

```
tax.df %>%
  summarize(k.p = sum(!is.na(phylum)),
            k.c = sum(!is.na(class)),
            k.o = sum(!is.na(order)),
            k.f = sum(!is.na(family)),
            k.g = sum(!is.na(genus))) -> k.vec
k.vec
```

```
##   k.p k.c k.o k.f k.g
## 1 932 786 740 586 386
```

```
n.asvs <- dim(tax.df)[1]
(k.vec / n.asvs) *100
```

```
##   k.p      k.c      k.o      k.f      k.g
## 1 100 84.33476 79.39914 62.87554 41.41631
```

```
# unknown phylum
tax.df %>%
  filter(is.na(phylum)) -> unk.p
unk.p
```

```
## [1] kingdom          phylum           class            order
## [5] family           genus            species          ASV
## [9] phylum.fromBlast
## <0 rows> (or 0-length row.names)
```

```
# unknown class
tax.df %>%
  filter(is.na(class)) -> unk.c
# unknown class
tax.df %>%
  filter(is.na(order)) -> unk.o
```

```
seqs <- refseq(ps)
unk.p.seqs <- seqs[names(seqs) %in% unk.p$ASV]
#writeXStringSet(unk.p.seqs, file.path(merged_path, "unk_phylum_after6Trim.fasta"))
unk.c.seqs <- seqs[names(seqs) %in% unk.c$ASV]
unk.o.seqs <- seqs[names(seqs) %in% unk.o$ASV]
unk.o.seqs
```

```
## DNAStringSet object of length 192:
##       width seq                                               names
##   [1]   352 TGGCTCTTGCAACGATGAAGAA...ATCAGGCAAGGCTACCCGCTGA ASV_2299
##   [2]   298 TGGCTCTTGCAACGATGAAGAA...ATCAAGCAAGACTACCCGCTGA ASV_5152
##   [3]   350 AGGCTCTTGCAGCGATGAAGAA...ATCAGGCAAGATTACCCGCTGA ASV_2411
##   [4]   384 TGGCTCTTGCAACGATGAAGAA...ATCAGGTAAGGCTACCCGCTGA ASV_3039
##   [5]   296 TGGCTCTTGCAACGATGAAGAA...ATCAGGCAAGACTACCCGCTGA ASV_4353
##   ...   ... ...
## [188]   386 TGGCTCTCGCATCGATGAAGAA...ATCAGGTAGGGCTACCCGCTGA ASV_3443
## [189]   376 TGGCTCTCGCATCGATGAAGAA...ATCAGGTAGGAATACCCGCTGA ASV_3320
## [190]   373 TGGCTCTCGCATCGATGAAGAA...ATCAGGTAGGAATACCCGCTGA ASV_2380
## [191]   417 TGGCTCTCGCATCGATGAAGAA...ATCAAGCAAGAACACCCGCTGA ASV_2084
## [192]   404 TGGCTCTCGCATCGATGAAGAA...ATCAAGTAAGACTACCCGCTGA ASV_1074
```

```
#look unidentified phylum ASVs on MycoBank
#ASV_4060 = maybe Rozellomycota, *
#ASV_5277 = maybe Rozellomycota, *
#ASV_686 = Chytridiomycota, **
#ASV_2273 = no sequences found -- genbank Galactomyces sp/Ascomycota
#ASV_2030 = no sequences found -- genbank Acaulopage sp/Zoopagomycota
#ASV_1640 = maybe Chytridiomycota, *
#ASV_3099 = no sequences found -- genbank Saccharomycetales sp/Ascomycota
#ASV_1596 = maybe Rozellomycota, *
#ASV_1536 = maybe Chytridiomycota, *
#ASV_4425 = no sequences found -- genbank Vermispora sp/Ascomycota
#ASV_2128 = no sequences found -- genbank Bulleribasidium/Basidiomycota
#ASV_4536 = maybe Rozellomycota, *
#ASV_2973 = probably Rozellomycota, **
#ASV_3323 = no sequences found -- genbank Ascomycota
#ASV_336 = probably Chytridiomycota, **
#ASV_699 = maybe Chytridiomycota, *
#ASV_6526 = maybe Basidiobolomycota, *
#ASV_1898 = 1 result as "unidentified fungi", * -- genbank Polyphlyctis sp./Chytridiomycota
#ASV_2573 = 1 result as "unidentified fungi", * -- genbank Polyphlyctis sp./Chytridiomycota
#ASV_3971 = maybe Rozellomycota, no star
#ASV_4397 = maybe Rozellomycota, *
#ASV_5935 = probably Chytridiomycota, ***
#ASV_4145 = probably Chytridiomycota, **

# # update taxonomy with these phylum assignments
# up.asvs <- c("ASV_4060","ASV_5277","ASV_686","ASV_2273","ASV_2030","ASV_1640","ASV_3099",
#   "ASV_1596","ASV_1536","ASV_4425","ASV_2128","ASV_4536","ASV_2973",
#   "ASV_3323","ASV_336","ASV_699","ASV_6526","ASV_1898","ASV_2573","ASV_3971",
#   "ASV_4397","ASV_5935","ASV_4145")
# up.phylum <- c("Rozellomycota","Rozellomycota","Chytridiomycota",
#                "Ascomycota","Zoopagomycota",
#                "Chytridiomycota","Ascomycota","Rozellomycota","Chytridiomycota",
```

```
#                 "Ascomycota","Basidiomycota","Rozellomycota",
#                 "Rozellomycota","Ascomycota","Chytridiomycota","Chytridiomycota",
#                 "Basidiobolomycota","Chytridiomycota","Chytridiomycota",
#                 "Rozellomycota","Rozellomycota","Chytridiomycota","Chytridiomycota")
# indx <- data.frame(up.asvs, up.phylum, stringsAsFactors = F)
# indx$phylum <- paste0("p__",up.phylum)
#
# x <- tax.df$ASV %in% indx$up.asvs
# tax.df[x,"phylum"] <- indx$phylum
# tax.mat <- as.matrix(tax.df)
# tax_table(ps) <- tax.mat
# saveRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
#ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
```

2. Update the taxonomic tree because I added phylum assignemnt to the remaining unknowns

```
ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
tax <- data.frame(tax_table(ps), stringsAsFactors = F)
colnames(tax)
```

```
## [1] "kingdom"          "phylum"           "class"            "order"
## [5] "family"           "genus"            "species"          "ASV"
## [9] "phylum.fromBlast"
```

```
tax %>%
  mutate(tip.label = ASV) %>%
  select(-ASV) -> asv.tax
asv.tax %>%
  group_by(phylum) %>%
  summarize(n = length(tip.label))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 12 x 2
##    phylum                    n
##    <chr>                 <int>
##  1 p__Ascomycota           277
##  2 p__Basidiobolomycota      1
##  3 p__Basidiomycota        249
##  4 p__Blastocladiomycota     2
##  5 p__Chytridiomycota       76
##  6 p__Entorrhizomycota       1
##  7 p__Glomeromycota        253
##  8 p__Kickxellomycota        1
##  9 p__Mortierellomycota     26
## 10 p__Mucoromycota           9
## 11 p__Rozellomycota         36
## 12 p__Zoopagomycota          1
```

```
# write.table(asv.tax, file = "data_intermediates/phylogeneticTree/asv_tax_nounkp.txt",
#             row.names = FALSE,
#             col.names = FALSE, sep = "\t")
```

```
#perl taxonomy_to_tree.pl -f asv_tax_nounkp.txt > asv_tax_nounkp.tre
```

The format of asv_tax_nounkp.tre is odd, so open it in FigTree and rename to asv_tax_nounkp_formated.tre

3. Update phylogenetic tree in phyloseq objects

```
ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
tree <- ape::read.nexus(file = "data_intermediates/phylogeneticTree/asv_tax_nounkp_formated.tre")
asvs <- taxa_names(ps)
pruned.tree <- ape::keep.tip(tree, asvs)
phy_tree(ps) <- pruned.tree

# by tissue
ps
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 932 taxa and 332 samples ]
## sample_data() Sample Data:       [ 332 samples by 75 sample variables ]
## tax_table()   Taxonomy Table:    [ 932 taxa by 9 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 932 tips and 1162 internal nodes ]
## refseq()      DNAStringSet:      [ 932 reference sequences ]
```

```
occ.cutoff <- 6 # must be present in at least X samples (e.g. not an entire site)

ps.l <- subset_samples(ps, Tissue == "L")
ps.l.pa <- ps.l
otu_table(ps.l.pa) <- (otu_table(ps.l.pa) >= occ.cutoff) *1
ps.l <- prune_taxa(colSums(otu_table(ps.l.pa)) != 0, ps.l)
ps.l # 207 ASVs, 109 samples
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 207 taxa and 109 samples ]
## sample_data() Sample Data:       [ 109 samples by 75 sample variables ]
## tax_table()   Taxonomy Table:    [ 207 taxa by 9 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 207 tips and 85 internal nodes ]
## refseq()      DNAStringSet:      [ 207 reference sequences ]
```

```
#saveRDS(ps.l, file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_leaf.RData"))

ps.r <- subset_samples(ps, Tissue == "R")
ps.r.pa <- ps.r
otu_table(ps.r.pa) <- (otu_table(ps.r.pa) >= occ.cutoff) *1
ps.r <- prune_taxa(colSums(otu_table(ps.r.pa)) != 0, ps.r)
ps.r # 675 ASVs, 111 samples
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 675 taxa and 111 samples ]
## sample_data() Sample Data:       [ 111 samples by 75 sample variables ]
## tax_table()   Taxonomy Table:    [ 675 taxa by 9 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 675 tips and 171 internal nodes ]
## refseq()      DNAStringSet:      [ 675 reference sequences ]
```

```
#saveRDS(ps.r, file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_root.RData"))

ps.s <- subset_samples(ps, Tissue == "S")
ps.s.pa <- ps.s
```

```
otu_table(ps.s.pa) <- (otu_table(ps.s.pa) >= occ.cutoff) *1
ps.s <- prune_taxa(colSums(otu_table(ps.s.pa)) != 0, ps.s)
ps.s # 789 ASVs, 112 samples
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 789 taxa and 112 samples ]
## sample_data() Sample Data:       [ 112 samples by 75 sample variables ]
## tax_table()   Taxonomy Table:    [ 789 taxa by 9 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 789 tips and 202 internal nodes ]
## refseq()      DNAStringSet:      [ 789 reference sequences ]
```

```
#saveRDS(ps.s, file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_soil.RData"))
```

4. Plot phylogentic tree with ASVs

```
ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
tree <- ape::read.nexus(file = "data_intermediates/phylogeneticTree/asv_tax_nounkp_formated.tre")
asvs <- taxa_names(ps)
pruned.tree <- ape::keep.tip(tree, asvs)
phy_tree(ps) <- pruned.tree

# library(ggtree)
# p <- ggtree(ps, ladderize = T, aes(color = phylum)) +
#   geom_tiplab(size = 1) +
#   theme(legend.position = "right")
#p
# pdf(file.path(out_path, "taxonomy_tree_allasvs.pdf"), width = 10, height = 20)
# p
# dev.off()
```

5. Alpha - Summarize the number of ASVs per sample

```
ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env.RData"))
ps
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 3811 taxa and 332 samples ]
## sample_data() Sample Data:       [ 332 samples by 74 sample variables ]
## tax_table()   Taxonomy Table:    [ 3811 taxa by 9 taxonomic ranks ]
## refseq()      DNAStringSet:      [ 3811 reference sequences ]
```

```
rich.all <- estimate_richness(ps)
#rich.all

#ps.l <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_leaf.RData"))
ps.l <- subset_samples(ps, Tissue == "L")
rich.l <- estimate_richness(ps.l)

#ps.r <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_root.RData"))
ps.r <- subset_samples(ps, Tissue == "R")
ps.r
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 3811 taxa and 111 samples ]
```

```
## sample_data() Sample Data:       [ 111 samples by 74 sample variables ]
## tax_table()   Taxonomy Table:    [ 3811 taxa by 9 taxonomic ranks ]
## refseq()      DNAStringSet:      [ 3811 reference sequences ]
```

```
rich.r <- estimate_richness(ps.r) # this is the only one I get a warning about singletons for??

#ps.s <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_soil.RData"))
ps.s <- subset_samples(ps, Tissue == "S")
ps.s
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 3811 taxa and 112 samples ]
## sample_data() Sample Data:       [ 112 samples by 74 sample variables ]
## tax_table()   Taxonomy Table:    [ 3811 taxa by 9 taxonomic ranks ]
## refseq()      DNAStringSet:      [ 3811 reference sequences ]
```

```
rich.s <- estimate_richness(ps.s)


rich.all$dataset <- "All"
rich.l$dataset <- "L"
rich.r$dataset <- "R"
rich.s$dataset <- "S"
rich.df <- rbind(rich.all, rich.l, rich.r, rich.s)
rich.df %>%
  group_by(dataset) %>%
  summarize(n = length(Observed),
            mean = mean(Observed),
            se = sd(Observed)/sqrt(n)) -> summ.asvs
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
summ.asvs
```

```
## # A tibble: 4 x 4
##   dataset     n  mean    se
##   <chr>   <int> <dbl> <dbl>
## 1 All       332  74.2  2.17
## 2 L         109  47.0  1.45
## 3 R         111  59.4  2.10
## 4 S         112 115.   3.42
```

```
#write.csv(summ.asvs, file = file.path(out_path, "asvs_per_samp.csv"))
```

6. Alpha – Plot and test differences in richness and phylogenetic diversity

```
# phylogenetic distance
library(DESeq2)
```

```
## Loading required package: S4Vectors
```

```
## Loading required package: stats4
```

```
## Loading required package: BiocGenerics
```

```
## Loading required package: parallel
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB

## The following objects are masked from 'package:dplyr':
##
##     combine, intersect, setdiff, union

## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, basename, cbind, colnames,
##     dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
##     grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
##     order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##     rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
##     union, unique, unsplit, which, which.max, which.min

##
## Attaching package: 'S4Vectors'

## The following objects are masked from 'package:dplyr':
##
##     first, rename

## The following object is masked from 'package:tidyr':
##
##     expand

## The following object is masked from 'package:base':
##
##     expand.grid

## Loading required package: IRanges

##
## Attaching package: 'IRanges'

## The following object is masked from 'package:phyloseq':
##
##     distance

## The following objects are masked from 'package:dplyr':
##
##     collapse, desc, slice

## The following object is masked from 'package:purrr':
##
##     reduce

## Loading required package: GenomicRanges

## Loading required package: GenomeInfoDb

## Loading required package: SummarizedExperiment

## Loading required package: Biobase
```

```
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.

##
## Attaching package: 'Biobase'

## The following object is masked from 'package:phyloseq':
##
##     sampleNames

## Loading required package: DelayedArray

## Loading required package: matrixStats

##
## Attaching package: 'matrixStats'

## The following objects are masked from 'package:Biobase':
##
##     anyMissing, rowMedians

## The following object is masked from 'package:dplyr':
##
##     count

##
## Attaching package: 'DelayedArray'

## The following objects are masked from 'package:matrixStats':
##
##     colMaxs, colMins, colRanges, rowMaxs, rowMins, rowRanges

## The following object is masked from 'package:purrr':
##
##     simplify

## The following objects are masked from 'package:base':
##
##     aperm, apply, rowsum
library(picante)

## Loading required package: ape

## Loading required package: vegan

## Loading required package: permute

## Loading required package: lattice

## This is vegan 2.5-6

## Loading required package: nlme

##
## Attaching package: 'nlme'

## The following object is masked from 'package:IRanges':
##
##     collapse
```

```
## The following object is masked from 'package:dplyr':
##
##     collapse
```

```
gm_mean = function(x, na.rm=TRUE){ exp(sum(log(x[x > 0]), na.rm=na.rm) / length(x))}
#
# calc vst
ps <- readRDS(ps, file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))
ps_ds <- phyloseq_to_deseq2(ps, ~1) # convert phyloseq to DeSeq object
```

```
## converting counts to integer mode
```

```
geoMeans = apply(counts(ps_ds), 1, gm_mean) # calc geometric mean of each ASV
ps_ds = estimateSizeFactors(ps_ds, type="ratio", geoMeans = geoMeans)
ps_ds = estimateDispersions(ps_ds, fitType = "parametric")
```

```
## gene-wise dispersion estimates
```

```
## mean-dispersion relationship
```

```
## final dispersion estimates
```

```
#plotDispEsts(ps_ds) # plot the dispersion estimates
vst <- getVarianceStabilizedData(ps_ds)
vst <- t(vst) # need to make the rows samples
# calculate Faith's PD
df.pd <- pd(vst, phy_tree(ps), include.root = F)
df.pd$sample.name.match <- row.names(df.pd)
sam <- data.frame(sample_data(ps))
df.pd %>%
  left_join(sam) -> alpha
```

```
## Joining, by = "sample.name.match"
```

```
alpha %>%
  group_by(Tissue) %>%
  summarize(n = length(SR),
            SR.mean = mean(SR),
            SR.se = sd(SR)/sqrt(n),
            PD.mean = mean(PD),
            PD.se = sd(PD)/sqrt(n)) -> alpha.tab
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
mod.sr <- lm(SR ~ Tissue, data = alpha)
TukeyHSD(aov(mod.sr))
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = mod.sr)
##
## $Tissue
##          diff       lwr      upr     p adj
## R-L -3.706257 -9.020603  1.60809 0.2295516
## S-L 28.749181 23.446602 34.05176 0.0000000
## S-R 32.455438 27.177124 37.73375 0.0000000
```

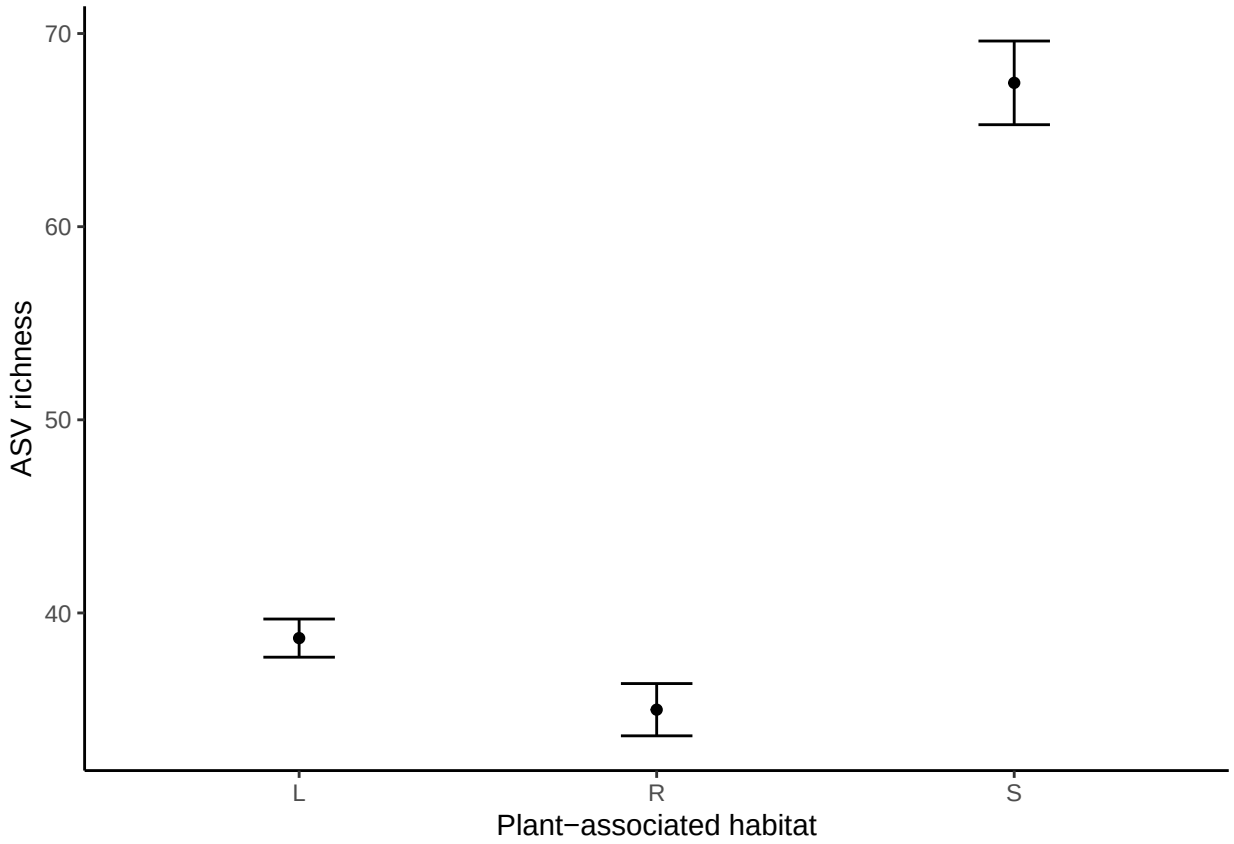
```r
mod.pd <- lm(PD ~ Tissue, data = alpha)
anova(mod.pd)
```

```
## Analysis of Variance Table
##
## Response: PD
##            Df     Sum Sq    Mean Sq F value    Pr(>F)
## Tissue      2 4865236395 2432618198  188.24 < 2.2e-16 ***
## Residuals 329 4251729893   12923191
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

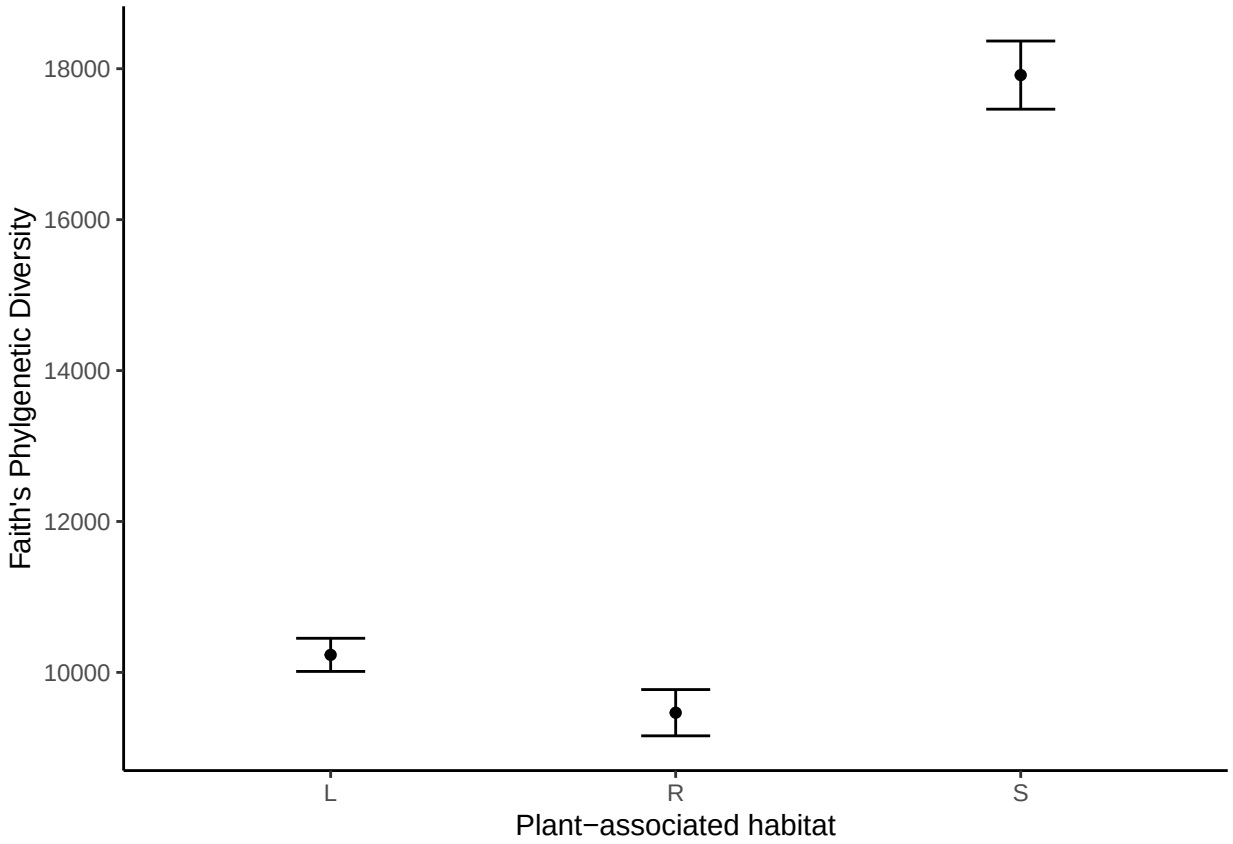
```r
TukeyHSD(aov(mod.pd))
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = mod.pd)
##
## $Tissue
##          diff       lwr       upr    p adj
## R-L -766.5311 -1907.822  374.7595 0.255166
## S-L 7682.8801  6544.117 8821.6436 0.000000
## S-R 8449.4112  7315.859 9582.9636 0.000000
```

```r
#write.csv(alpha.tab, file = file.path(out_path, "alphaDiv.csv"))

p1 <- ggplot(alpha.tab, aes(x = Tissue, y = SR.mean)) +
  geom_point() +
  geom_errorbar(aes(ymin = SR.mean - SR.se,
                    ymax = SR.mean + SR.se), width = .2) +
  ylab("ASV richness") +
  xlab("Plant-associated habitat") +
  theme_classic()

p2 <- ggplot(alpha.tab, aes(x = Tissue, y = PD.mean)) +
  geom_point() +
  geom_errorbar(aes(ymin = PD.mean - PD.se,
                    ymax = PD.mean + PD.se), width = .2) +
  ylab("Faith's Phylgenetic Diversity") +
  xlab("Plant-associated habitat") +
  theme_classic()

p1
```




```
p2
```




```
# library(gridExtra)
# pdf(file = file.path(out_path, "alphaDiv.pdf"), width = 4, height = 6)
# grid.arrange(p1 + ggtitle("a"),
#              p2 + ggtitle("b"), ncol = 1)
# dev.off()
```

7. Beta – Plot DPCOAs

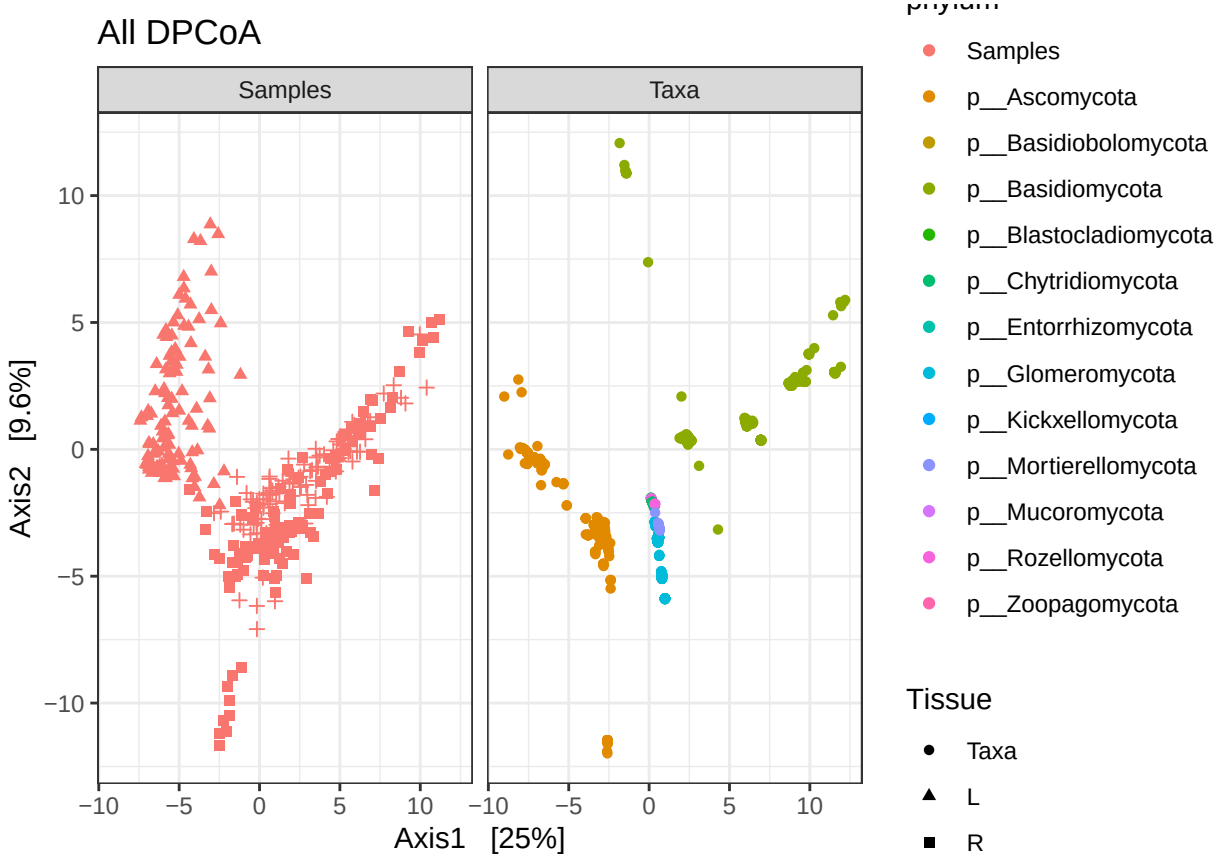
All

```
ps <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs.RData"))

# ps
tree <- phy_tree(ps)
tree
```

```
##
## Phylogenetic tree with 932 tips and 254 internal nodes.
##
## Tip labels:
##  ASV_3509, ASV_540, ASV_3014, ASV_1733, ASV_551, ASV_2654, ...
##
## Unrooted; includes branch lengths.
```

```
asv <- data.frame(otu_table(ps), stringsAsFactors = F)
# # # square root of the cophenetic/patristic (cophenetic.phylo)
# # # cophenetic.phylo = pairwise distances between the pairs of tips from a phylogenetic tree using it
# # #detach("package:compositions", unload = TRUE)
```

```
# library(ade4); packageVersion("ade4")
# phylo.dist <- cophenetic.phylo(tree)
# phylo.dist <- as.dist(phylo.dist)
# sqrt.phylo.dist <- sqrt(phylo.dist)
# mod.all <- dpcoa(df = asv, dis = sqrt.phylo.dist, scannf = FALSE, nf = 2, RaoDecomp = TRUE)
#saveRDS(mod.all, file = file.path(out_path, "dpcoa_all.RData"))
mod.all <- readRDS(file = file.path(out_path, "dpcoa_all.RData"))

plot_ordination(ps, mod.all, type="split",
                color = "phylum", shape = "Tissue") +
  ggplot2::scale_colour_discrete() +
  ggplot2::theme_bw() +
  ggtitle("All DPCoA")
```




```
# ggsave(filename = file.path(out_path, "dpcoa_all.pdf"),
#        width = 6, height = 4)
#
df <- data.frame(ASV = row.names(mod.all$dls),
                 DPCoA1 = mod.all$dls$CS1,
                 DPCoA2 = mod.all$dls$CS2)
tax <- data.frame(tax_table(ps), stringsAsFactors = F)
tax %>%
  left_join(df) -> df.tax
```

```
## Joining, by = "ASV"
```

```
###
```

Leaf [commented out]

```
# ps.l <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_leaf.RData"))
# ps.l
# # mod.l <- DPCoA(ps.l, correction = cailliez)
# # saveRDS(mod.l, file = file.path(out_path, "dpcoa_leaf.RData"))
# mod.l <- readRDS(file = file.path(out_path, "dpcoa_leaf.RData"))
# plot_ordination(ps.l, mod.l, type="split",
#                 color = "phylum") +
#   ggplot2::scale_colour_discrete() +
#   ggplot2::theme_bw() +
#   ggtitle("Leaf DPCoA")
# ggsave(filename = file.path(out_path, "dpcoa_leaf.pdf"),
#        width = 6, height = 4)
```

Who are the Basidiomycete ASVs driving variation along Axis1?

```
# dls <- data.frame(ASV = row.names(mod.l$dls), mod.l$dls,
#            row.names = NULL, stringsAsFactors = F)
# tax <- data.frame(tax_table(ps.l))
# tax %>%
#   left_join(dls) ->dls.df
# dls.df %>%
#   filter(CS1 < -1)
```

All of these are Puccinia andropogonis (a common rust)

Who are the Ascomycete ASVs that drive low Axis2 values?

```
# dls.df %>%
#   filter(CS2 < -0.4) %>%
#   arrange(CS2) %>%
#   select(CS2, ASV, phylum, class, order, family, genus, species)
```

All in the order Pleosporales

Who are the Ascomycete ASVs that drive high Axis2 values?

```
# dls.df %>%
#   filter(CS2 > 0.4) %>%
#   arrange(CS2) %>%
#   select(CS2, ASV, phylum, class, order, family, genus, species)
```

Half of these ASVs classify to the family Mycosphaerellaceae

Root [commented out]

```
# ps.r <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_root.RData"))
# ps.r
# # mod.r <- DPCoA(ps.r, correction = cailliez)
# # saveRDS(mod.r, file = file.path(out_path, "dpcoa_root.RData"))
#
# mod.r <- readRDS(file = file.path(out_path, "dpcoa_root.RData"))
# plot_ordination(ps.r, mod.r, type="split",
#                 color = "phylum") +
#   ggplot2::scale_colour_discrete() +
#   ggplot2::theme_bw() +
```

```
#   ggtitle("Root DPCoA")
# ggsave(filename = file.path(out_path, "dpcoa_root.pdf"),
#        width = 6, height = 4)
```

Who are the Ascomycete ASVs driving low values on Axis1?

```
# dls <- data.frame(ASV = row.names(mod.r$dls), mod.r$dls,
#            row.names = NULL, stringsAsFactors = F)
# tax <- data.frame(tax_table(ps.r))
# tax %>%
#   left_join(dls) -> dls.df
# dls.df %>%
#   filter(CS1 < -0.6) %>%
#   arrange(CS1)
```

All of these are unclassified below phylum.

Who are the Basidiomycete ASVs driving high values on Axis1?

```
# dls.df %>%
#   filter(CS1 > 0.7) %>%
#   arrange(CS1) %>%
#   select(CS1, ASV, phylum, class, order, family, genus, species)
```

Many match to Mycena pura

Who are the Basidiomycete ASVs driving high values on Axis2?

```
# dls.df %>%
#   filter(CS2 > 0.5) %>%
#   arrange(CS2) %>%
#   select(CS2, ASV, phylum, class, order, family, genus, species)
```

ASV_5 is unclassified below phylum. The glomeromycota are also underpin high values on Axis2.

Soil [commented out]

```
# ps.s <- readRDS(file = file.path(merged_path, "phyloseq_samps_env_trimTreeASVs_soil.RData"))
# ps.s
# #mod.s <- DPCoA(ps.s, correction = cailliez)
# #saveRDS(mod.s, file = file.path(out_path, "dpcoa_soil.RData"))
# mod.s <- readRDS(file = file.path(out_path, "dpcoa_soil.RData"))
# plot_ordination(ps.s, mod.s, type="split",
#                 color = "phylum") +
#   ggplot2::scale_colour_discrete() +
#   ggplot2::theme_bw() +
#   ggtitle("Soil DPCoA")
# ggsave(filename = file.path(out_path, "dpcoa_soil.pdf"),
#        width = 6, height = 4)
```

Who is the Basidio ASV driving high values on Axis1?

```
# dls <- data.frame(ASV = row.names(mod.s$dls), mod.s$dls,
#            row.names = NULL, stringsAsFactors = F)
# tax <- data.frame(tax_table(ps.s))
# tax %>%
#   left_join(dls) ->dls.df
# dls.df %>%
```

```
#   filter(CS1 > 0.75) %>%
#   arrange(CS1)
```

Genus Camarophyllopsis

Who are the Basidiomycete ASVs driving high values on Axis2?

```
# dls.df %>%
#   filter(CS2 > 0.25) %>%
#   arrange(CS2) %>%
#   select(CS2, ASV, phylum, class, order, family, genus, species)
```

Unclassified Agaricales

Who are the Ascomycete ASVs driving low values on Axis2?

```
# dls.df %>%
#   filter(CS2 < -0.5) %>%
#   arrange(CS2) %>%
#   select(CS2, ASV, phylum, class, order, family, genus, species)
```

Unclassified Agaricales