**Basic setup and some useful functions**

1. **Install R**  
   <https://cran.r-project.org/>
2. **Install R Studio**  
   <https://www.rstudio.com/products/rstudio/download/#download>
3. **Commands to install all the packages required**  
   install.packages("phyloseq")  
   install.packages("kable")  
   install.packages("BiocManager")

install.packages("ggplot")

install.packages("knitr")

1. **Load packages**  
   library(microbiome)
2. **Define file paths**  
   rich\_dense\_biom = system.file("extdata", "rich\_dense\_otu\_table.biom", package="phyloseq")

rich\_sparse\_biom = system.file("extdata", "rich\_sparse\_otu\_table.biom", package="phyloseq")

min\_dense\_biom = system.file("extdata", "min\_dense\_otu\_table.biom", package="phyloseq")

min\_sparse\_biom = system.file("extdata", "min\_sparse\_otu\_table.biom", package="phyloseq")

treefilename = system.file("extdata", "biom-tree.phy", package="phyloseq")

refseqfilename = system.file("extdata", "biom-refseq.fasta", package="phyloseq")

1. **Store the result of your imported data**  
   myStoredData = import\_biom(rich\_dense\_biom, treefilename, refseqfilename, parseFunction=parse\_taxonomy\_greengenes)
2. **Check what is stored**  
   myStoredData # Print all the data

view(myStoredData) # Access file and open up to see different variables

1. **Access specific part of the file**  
   myStoredData@otu\_table
2. **Create phylogenetic tree**  
   plot\_tree(myData, color="Genus", shape="BODY\_SITE", size="abundance")

plot\_tree(myData, color="Bar", shape="BODY\_SITE", size="abundance")

plot\_tree(myData, color="Species", shape="BarcodeSequence", size="abundance")

**Diversity Calculations**

**Alpha**

1. **Load data**  
   data(atlas1006)
2. **Calculate Alpha diversity for each sample**  
   tab <- global(pseq, index = “all”)  
   kable(head(tab))
3. **Dominance (refers to the abundance of most abundant specie)**  
   tab <- dominance(pseq, index = “all”)  
   kable(head(tab))
4. **Rarity (refers to abundance of less abundant specie)**  
   tab <- rarity(pseq, index = “all”)  
   kable(head(tab))

**Beta**

1. **Load data and packages**  
   library(microbiome)

library(dplyr)

data(peerj32)

pseq <- peerj32$phyloseq

1. **Calculate group divergence**  
   b.pla <- divergence(subset\_samples(pseq, group == "Placebo"))

b.lgg <- divergence(subset\_samples(pseq, group == "LGG"))

1. **View differences in sample using a boxplot**  
     
   boxplot(list(LGG = b.lgg, Placebo = b.pla))
2. **Calculate Beta diversity change over time**  
   betas <- list()

groups <- as.character(unique(meta(pseq)$group))

for (g in groups) {

#df <- meta(subset\_samples(pseq, group == g))

df <- subset(meta(pseq), group == g)

beta <- c()

for (subj in df$subject) {

# Pick the samples for this subject

dfs <- subset(df, subject == subj)

# Check that the subject has two time points

if (nrow(dfs) == 2) {

s <- as.character(dfs$sample)

# Here with just two samples we can calculate the

# beta diversity directly

beta[[subj]] <- 1-cor(abundances(pseq)[, s[[1]]],

abundances(pseq)[, s[[2]]],

method = "spearman")

}

}

betas[[g]] <- beta

}

boxplot(betas)

1. **Example Code of Betadiversity change over time**  
     
   data(atlas1006)

pseq <- atlas1006

# Identify subject with the longest time series (most time points)

s <- names(which.max(sapply(split(meta(pseq)$time, meta(pseq)$subject), function (x) {length(unique(x))})))

f

# Pick the metadata for this subject and sort the samples by time

library(dplyr)

df <- meta(pseq) %>% filter(subject == s) %>% arrange(time)

# Calculate the beta diversity between each time point and the baseline (first) time point

beta <- c(0, 0) # Baseline similarity

s0 <- subset(df, time == 0)$sample

for (tp in df$time[-1]) {

# Pick the samples for this subject

#If the same time point has more than one sample, pick one at random

st <- sample(subset(df, time == tp)$sample, 1)

a <- abundances(pseq)

b <- 1 - cor(a[, s0], a[, st], method = "spearman")

beta <- rbind(beta, c(tp, b))

}

colnames(beta) <- c("time", "beta")

beta <- as.data.frame(beta)

library(ggplot2)

p <- ggplot(beta, aes(x = time, y = beta)) +

geom\_point() + geom\_line()

print(p)

**Univariate Comparison**

1. **Load example data**  
   library(microbiome)

library(ggplot2)

library(dplyr)

data(peerj32)

pseq <- peerj32$phyloseq

1. **Abundance Plot**

p <- boxplot\_abundance(pseq, x = "time", y = "Akkermansia", line = "subject") +

scale\_y\_log10()

print(p)

1. **Linear Model Univariate Comparison with random effect subject term**

# Get sample metadata

dfs <- meta(pseq)

# Add abundance as the signal to model

dfs$signal <- abundances(pseq)["Akkermansia", rownames(dfs)]

# Paired comparison

# with fixed group effect and random subject effect

library(lme4)

out <- lmer(signal ~ group + (1|subject), data = dfs)

out0 <- lmer(signal ~ (1|subject), data = dfs)

comp <- anova(out0, out)

pv <- comp[["Pr(>Chisq)"]][[2]]

print(pv)

**Multivariate Analysis**

1. **Load example data**  
     
   # Load libraries

library(microbiome)

library(ggplot2)

library(dplyr)

# Probiotics intervention example data

data(peerj32) # Source: https://peerj.com/articles/32/

pseq <- peerj32$phyloseq # Rename the example data

# Pick relative abundances (compositional) and sample metadata

pseq.rel <- microbiome::transform(pseq, "compositional")

otu <- abundances(pseq.rel)

meta <- meta(pseq.rel)

1. **Visualize the population density and highlight sample groups**  
     
   p <- plot\_landscape(pseq.rel, method = "NMDS", distance = "bray", col = "group", size = 3)

print(p)

1. **Evaluate whether group has a significant effect on overall gut microbiota composition**  
     
   # samples x species as input

library(vegan)

permanova <- adonis(t(otu) ~ group,

data = meta, permutations=99, method = "bray")

# P-value

print(as.data.frame(permanova$aov.tab)["group", "Pr(>F)"])

1. **Check that variance homogeneity assumptions**  
     
   dist <- vegdist(t(otu))

anova(betadisper(dist, meta$group))

1. **Coefficients for the top taxa separating the groups**  
     
   coef <- coefficients(permanova)["group1",]

top.coef <- coef[rev(order(abs(coef)))[1:20]]

par(mar = c(3, 14, 2, 1))

barplot(sort(top.coef), horiz = T, las = 1, main = "Top taxa")