Tutorial Commands and Overview

Overall Assignment

From Dr. Rosen: “This is what I want you to review: the Microbiome R package, specifically these functions:

Alpha Diversity: https://microbiome.github.io/microbiome/Diversity.html

(please review how the various metrics are calculated like Chao diversity, simpson/shannon, etc.)

Beta Diversity:  http://microbiome.github.io/microbiome/Betadiversity.html

Univariate comparison of a taxon from microbial communities using linear mixed model:

http://microbiome.github.io/microbiome/Mixedmodels.html

Multivariate community-level differences:

http://microbiome.github.io/microbiome/PERMANOVA.html

Present on some of the datasets they use, so you can understand the results: http://microbiome.github.io/microbiome/Data.html”

Installing R

1) Download and install latest R version go to:

<https://cran.r-project.org/>

2) Install R studio from:

<https://www.rstudio.com/products/rstudio/download/#download>

3) Install Other packages

- Other packages are needed such as biocondyctor and phyloseq

-To install packages needed use:

>install.packages("phyloseq")

install.packages("kable")

-To check current version of R

>version

To update R version:

Some parts of the tutorial might require certain packages, without which you will get an error, and to intstall the package named in error use install.package command. Examples of additional packages to install, if missing:

>install.packages("BiocManager")

> install.packages("ggplot")

> install.packages("knitr")

To load a package use library() command

Example:

>library(microbiome)

Data imprt based on : <http://joey711.github.io/phyloseq/import-data>

1) Define file paths:

First, define the 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")

2) store the result of your import as some variable name:

myStoredData = import\_biom(rich\_dense\_biom, treefilename, refseqfilename, parseFunction=parse\_taxonomy\_greengenes)

3) To check what is stored:

>myStoredData

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

- To access specific parts of the file, for example if you want to look inside the taxonomy table:

>myStoredData@otu\_table or click the file icon with the green arrow (to the right, as you scroll over variable name, example the ‘tax\_table’

**Creating 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 diversity/species richness according to different body sites:

data(atlas1006)

> g <- global (atlas1006,index = “all”)

>plot\_richness(myData, x="BODY\_SITE", color="Description")

To calculate alpha diversity of each sample:

>tab <- global(pseq, index = "all")

kable(head(tab))

observed richness with given detection threshold(s)

>tab <- richness(pseq)

kable(head(tab))

Dominance: The dominance index refers to the abundance of the most abundant species.

# Absolute abundances for the single most abundant taxa in each sample

>tab <- dominance(pseq, index = "all")

kable(head(tab))

Rarity and low abundance

>tab <- rarity(pseq, index = "all")

kable(head(tab))

Beta Diversity

1) load data and packages

library(microbiome)

library(dplyr)

data(peerj32)

pseq <- peerj32$phyloseq

2) Calculate group divergences within the LGG (probiotic) and Placebo groups>plot\_bar(myData, fill="Genus")

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

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

3) view differences in sample diversity in a boxplot:

boxplot(list(LGG = b.lgg, Placebo = b.pla))

Note\* : if you get following error: ‘Error in plot.new() : figure margins too large’, just need to expand graph window and or use dev.off() which make RStudio open up a new graphics device window with default settings ( will also delete whatever previous graphs you have in there)

* To calculate beta diversity changing across time, example code is:
* 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)

Example code of Betadiversity changing 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 Comparisons:

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

2) Abundance boxplot

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

scale\_y\_log10()

print(p)

## 3) Linear model Univariate comparison comparison with random effect subject term

Test individual taxonomic group

# 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)

## [1] 0.4556962

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)

2) Visualize the population density and highlight sample groups (probiotic treatment LGG vs Placebo):

p <- plot\_landscape(pseq.rel, method = "NMDS", distance = "bray", col = "group", size = 3)

print(p)

3) evaluate whether group (probiotics vs. placebo) has a significant effect on overall gut microbiota composition. Perform PERMANOVA:

# 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] 0.29

4)Check that variance homogeneity assumptions hold (to ensure the reliability of the results):

# Note the assumption of similar multivariate spread among the groups

# ie. analogous to variance homogeneity

# Here the groups have signif. different spreads and

# permanova result may be potentially explained by that.

dist <- vegdist(t(otu))

anova(betadisper(dist, meta$group))

## Analysis of Variance Table

##

## Response: Distances

## Df Sum Sq Mean Sq F value Pr(>F)

## Groups 1 0.000016 0.0000160 0.0043 0.9483

## Residuals 42 0.157881 0.0037591

5) 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")