Chart, waterfall chart

Description automatically generated

**Figure S1. Actinobacteria is the bacterial class that changes the most following varying OM removal treatments.** Changes in bacterial class are denoted by a log2 fold change plot.

[Correlation\_Analysis\_R](https://drive.google.com/file/d/1uwihzSzsW6eiP4OshfkfydkHpuZ-2WYI/view?usp=sharing)

### Correlation Analysis

# Team 4 - Seohee An, Priya Gill, Brandon Park, Chelsea Williams

# Loading packages

library(tidyverse)

library(vegan)

library(phyloseq)

library(DESeq2)

library(ggplot2)

# importing all data files

biom\_file <- import\_biom("table-with-taxonomy.biom")

metadata <- import\_qiime\_sample\_data("soil\_metadata.txt")

tree <- read\_tree\_greengenes("tree-soil.nwk")

### Alpha diversity functions

# Shannon's diversity

shannons = function(x){

present = x[x>0]

p = present/sum(present)

-sum(p\*log(p))

}

# Pielou's evennenss function

evenness = function(x){

present = x[x>0]

p = present/sum(present)

-sum(p\*log(p))/log(sum(x>0))

}

# Richness function

richness = function(x){

return(sum(x>0))

}

### Load data

biom = import\_biom("table-with-taxonomy.biom")

taxa\_table = otu\_table(biom\_file)

taxonomy = tax\_table(biom\_file)

metadata = read.table("soil\_metadata.txt",sep="\t",header=T,row.names = 1)

# Select only samples with metadata

microbial\_samples = colnames(taxa\_table)

metadata\_samples = rownames(metadata)

which\_metadata = c()

for (i in 1:dim(taxa\_table)[2]){

which\_metadata = c(which\_metadata,which(metadata\_samples == microbial\_samples[i]))

}

metadata = metadata[which\_metadata,]

### Calculate alpha diversity

metadata$richness = apply(taxa\_table,2,richness)

metadata$shannons = apply(taxa\_table,2,shannons)

metadata$evenness = apply(taxa\_table,2,evenness)

### Correlation plots of Shannon's vs metadata variables

# pH

ggplot(metadata,aes(x=pH,y=shannons)) +

geom\_point() +

labs(x = "pH", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

# Correlation

cor.test(metadata$pH,metadata$shannons)

# Regression

ph\_shannons\_lm = lm(shannons ~ pH, data = metadata)

summary(ph\_shannons\_lm)

# Total Nitrogen

ggplot(metadata,aes(x=Total.Nitrogen,y=shannons)) +

geom\_point() +

labs(x = "Total Nitrogen", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

# Correlation

cor.test(metadata$Total.Nitrogen,metadata$shannons)

# Regression

nitrogen\_shannons\_lm = lm(shannons ~ Total.Nitrogen, data = metadata)

summary(nitrogen\_shannons\_lm)

# Total Carbon

ggplot(metadata,aes(x=Total.Carbon,y=shannons)) +

geom\_point() +

labs(x = "Total Carbon", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

# Correlation

cor.test(metadata$Total.Carbon,metadata$shannons)

# Regression

carbon\_shannons\_lm = lm(shannons ~ Total.Carbon, data = metadata)

summary(carbon\_shannons\_lm)

# Moisture Content

ggplot(metadata,aes(x=Moisture.Content,y=shannons)) +

geom\_point() +

labs(x = "Moisture Content", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

# Correlation

cor.test(metadata$Moisture.Content,metadata$shannons)

# Regression

moisture\_shannons\_lm = lm(shannons ~ Moisture.Content, data = metadata)

summary(moisture\_shannons\_lm)

### Correlation plots of Shannon's Diversity vs metadata variables by LTSP

# pH

ggplot(metadata,aes(x=pH,y=shannons)) +

facet\_grid(~LTSP.Treatment) +

geom\_point() +

labs(x = "pH", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

ggplot(metadata,aes(x=pH,y=shannons,group=LTSP.Treatment,color=LTSP.Treatment)) +

geom\_point() +

labs(x = "pH", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

for (g in unique(metadata$LTSP.Treatment)){

print(paste("LTSP Treatment =",g))

temp\_metadata = filter(metadata,LTSP.Treatment==g)

# Correlation

temp\_cor = cor.test(temp\_metadata$pH,temp\_metadata$shannons)

print(paste("Correlation =",temp\_cor$estimate,

"p value =",temp\_cor$p.value))

# Regression

om\_ph\_shannons\_lm = lm(shannons ~ pH, data = temp\_metadata)

print(paste("Regression slope =",summary(om\_ph\_shannons\_lm)$coefficients[2,1],

"p value =",summary(om\_ph\_shannons\_lm)$coefficients[2,4]))

}

# Total Nitrogen

ggplot(metadata,aes(x=Total.Nitrogen,y=shannons)) +

facet\_grid(~LTSP.Treatment) +

geom\_point() +

labs(x = "Total Nitrogen", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

ggplot(metadata,aes(x=Total.Nitrogen,y=shannons,group=LTSP.Treatment,color=LTSP.Treatment)) +

geom\_point() +

labs(x = "Total Nitrogen", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

for (g in unique(metadata$LTSP.Treatment)){

print(paste("LTSP Treatment =",g))

temp\_metadata = filter(metadata,LTSP.Treatment==g)

# Correlation

temp\_cor = cor.test(temp\_metadata$Total.Nitrogen,temp\_metadata$shannons)

print(paste("Correlation =",temp\_cor$estimate,

"p value =",temp\_cor$p.value))

# Regression

om\_nitrogen\_shannons\_lm = lm(shannons ~ Total.Nitrogen, data = temp\_metadata)

print(paste("Regression slope =",summary(om\_nitrogen\_shannons\_lm)$coefficients[2,1],

"p value =",summary(om\_nitrogen\_shannons\_lm)$coefficients[2,4]))

}

# Total Carbon

ggplot(metadata,aes(x=Total.Carbon,y=shannons)) +

facet\_grid(~LTSP.Treatment) +

geom\_point() +

labs(x = "Total Carbon", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

ggplot(metadata,aes(x=Total.Carbon,y=shannons,group=LTSP.Treatment,color=LTSP.Treatment)) +

geom\_point() +

labs(x = "Total Carbon", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

for (g in unique(metadata$LTSP.Treatment)){

print(paste("LTSP Treatment =",g))

temp\_metadata = filter(metadata,LTSP.Treatment==g)

# Correlation

temp\_cor = cor.test(temp\_metadata$Total.Carbon,temp\_metadata$shannons)

print(paste("Correlation =",temp\_cor$estimate,

"p value =",temp\_cor$p.value))

# Regression

om\_carbon\_shannons\_lm = lm(shannons ~ Total.Carbon, data = temp\_metadata)

print(paste("Regression slope =",summary(om\_carbon\_shannons\_lm)$coefficients[2,1],

"p value =",summary(om\_carbon\_shannons\_lm)$coefficients[2,4]))

}

# Moisture Content

ggplot(metadata,aes(x=Moisture.Content,y=shannons)) +

facet\_grid(~LTSP.Treatment) +

geom\_point() +

labs(x = "Moisture Content", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

ggplot(metadata,aes(x=Moisture.Content,y=shannons,group=LTSP.Treatment,color=LTSP.Treatment)) +

geom\_point() +

labs(x = "Moisture Content", y = "Shannon's Diversity") +

geom\_smooth(method="lm",formula=y~x) +

theme\_bw(base\_size = 15)

for (g in unique(metadata$LTSP.Treatment)){

print(paste("LTSP Treatment =",g))

temp\_metadata = filter(metadata,LTSP.Treatment==g)

# Correlation

temp\_cor = cor.test(temp\_metadata$Moisture.Content,temp\_metadata$shannons)

print(paste("Correlation =",temp\_cor$estimate,

"p value =",temp\_cor$p.value))

# Regression

om\_moisture\_shannons\_lm = lm(shannons ~ Moisture.Content, data = temp\_metadata)

print(paste("Regression slope =",summary(om\_moisture\_shannons\_lm)$coefficients[2,1],

"p value =",summary(om\_moisture\_shannons\_lm)$coefficients[2,4]))

}

[Differential\_Abundance\_Analysis\_R](https://drive.google.com/file/d/1jJgzXk8fw18qlPYbz2iBjur_vtG-xPPH/view?usp=sharing)

### Differential Abundance Analysis

# Team 4 - Seohee An, Priya Gill, Brandon Park, Chelsea Williams

# Load CRAN packages

library(tidyverse)

library(vegan)

# Load Bioconductor packages

library(phyloseq)

library(DESeq2)

# Calculate relative abundance

calculate\_RA <- function(x) x/sum(x)

# Helper functions

gm\_mean <- function(x, na.rm = TRUE) {

exp(sum(log(x[x > 0]), na.rm = na.rm) / length(x))

}

# Export biom file and tree from QIIME2 and provide original metadata file

biom\_file <- import\_biom("table-with-taxonomy.biom")

metadata <- import\_qiime\_sample\_data("soil\_metadata.txt")

tree <- read\_tree\_greengenes("tree-soil.nwk")

# Combine all information into a single phyloseq object

physeq <- merge\_phyloseq(biom\_file, metadata, tree)

# Convert taxonomic rank from numbers to proper names

colnames(tax\_table(physeq)) <- c("Kingdom", "Phylum", "Class", "Order",

"Family", "Genus", "Species")

# Keep only abundant ASVs

# First determine counts of ASV across all samples

total\_counts <- taxa\_sums(physeq)

# Calculate relative abundance for each ASV

relative\_abundance <- calculate\_RA(total\_counts)

# Determine which ASVs are more abundant than 0.1%

abundant <- relative\_abundance > 0.001

abundant\_taxa <- prune\_taxa(abundant, physeq)

# Check the resulting new phyloseq object with much fewer taxa

abundant\_taxa

# Phyloseq object is called "Class"

class <- tax\_glom(abundant\_taxa, taxrank = "Class", NArm = FALSE)

class

# Only keep OM1 samples

OM1class <- subset\_samples(class, LTSP.Treatment == "OM1")

OM1class

# Limit to OM1 samples that were filtered

deseq\_feature <- phyloseq\_to\_deseq2(OM1class, ~ LTSP.Treatment)

geo\_means <- apply(counts(deseq\_feature), 1, gm\_mean)

deseq\_feature <- estimateSizeFactors(deseq\_feature, geoMeans = geo\_means)

deseq\_feature <- DESeq(deseq\_feature, fitType="local")

feature\_diff\_abund <- results(deseq\_feature)

# Define cutoff for the adjusted p-value

alpha <- 0.05

# Reformat information as data frame including feature as variable

significant\_feature <- as\_tibble(feature\_diff\_abund, rownames = "feature")

# Keep only significant results and sort by adjusted p-value

significant\_feature <- filter(significant\_feature, padj < alpha)

significant\_feature <- arrange(significant\_feature, padj)

# Get the taxonomic information as a data frame

taxa\_df <- as\_tibble(as.data.frame(tax\_table(physeq)), rownames = "feature")

# Combine the significant features with taxonomic classification

significant\_feature <- inner\_join(significant\_feature, taxa\_df)

# There 14 features that are different at FDR-corrected p < 0.05!

dim(significant\_feature)

# Plot differential abundance

significant\_feature %>%

ggplot(aes(x = log2FoldChange, y = Class)) +

geom\_col() +

xlab("Log2 Fold Change") +

# Label for y-axis

ylab("Class") +

theme\_bw(base\_size = 16)

# Convert to relative abundance and define taxrank as Class

physeq\_RA <- transform\_sample\_counts(physeq, calculate\_RA)

species\_RA <- tax\_glom(physeq\_RA, taxrank = "Class", NArm = FALSE)

# Graph actinobacteria and filter for that class with subset\_taxa

actinobacteria <- subset\_taxa(species\_RA, Class == "c\_\_Actinobacteria")

# Plot data by changing the table from wide to long format

actinobacteria\_melt <- psmelt(actinobacteria)

# Plot with ggplot function

# Define what dataset to plot and what variable on x-axis and y-axis

ggplot(actinobacteria\_melt, aes(x = as.factor(LTSP.Treatment), y = Abundance)) +

# Visualize data as Boxplot

geom\_boxplot(aes(fill = as.factor(LTSP.Treatment))) +

# Label for x-axis

xlab("LTSP Treatment") +

# Label for y-axis

ylab("Relative Abundance") +

# Plot title

ggtitle("The Relative Abundance of Actinobacteria") +

# colors for the different groups

scale\_fill\_manual(values=c("gray42", "gray42", "gray42", "gray42")) +

guides(fill = FALSE) +

# Set a standard theme for plot

theme\_bw(base\_size = 16)

[Beta\_Diversity\_Analysis\_R](https://drive.google.com/file/d/1dMWXcVoZIoGfN3lIZiMcXX_363Fue96i/view?usp=sharing)

### Beta Diversity Analysis

# Team 4 - Seohee An, Priya Gill, Brandon Park, Chelsea Williams

# Loading packages

library(tidyverse)

library(vegan)

library(phyloseq)

library(DESeq2)

# Helper functions

gm\_mean <- function(x, na.rm = TRUE) {

exp(sum(log(x[x > 0]), na.rm = na.rm) / length(x))

}

# Importing all data files

biom\_file <- import\_biom("table-with-taxonomy.biom")

metadata <- import\_qiime\_sample\_data("soil\_metadata.txt")

tree <- read\_tree\_greengenes("tree-soil.nwk")

# Combine all objects into phyloseq object

physeq <- merge\_phyloseq(biom\_file, metadata, tree)

# Overview of phyloseq object

physeq

# Use sample\_data() to look at metadata of phyloseq object

# Look only at first 6 lines of table with head()

head(sample\_data(physeq))

# Set set of random numbers

set.seed(600)

# taxonomic rank names

rank\_names(physeq)

# Rename column names for taxonomic ranks

colnames(tax\_table(physeq)) <- c("Kingdom", "Phylum", "Class",

"Order", "Family", "Genus",

"Species")

rank\_names(physeq)

# Filter data based on metadata category with ==

OM1 <- subset\_samples(physeq, LTSP.Treatment == "OM1")

OM1

# Filter out data based on metadata category with !=

no\_OM1 <- subset\_samples(physeq, LTSP.Treatment != "OM1")

no\_OM1

# format data as table with as.data.frame()

as.data.frame(sample\_names(OM1))

# Show read count for each sample using sample\_sums()

as.data.frame(sample\_sums(OM1))

# Exclude samples with less than 7000 reads

OM1\_7000 <- prune\_samples(sample\_sums(OM1) >= 7000, OM1)

OM1\_7000

# Beta diversity PCoA plot

# Diversity requires rarefied taxa tables

physeq\_rar <- rarefy\_even\_depth(physeq, sample.size = 4800)

# Convert to RA (relative abundance)

physeq\_rar\_RA <- transform\_sample\_counts(physeq\_rar, function(x) x/sum(x))

# Define ordinate() and set method = PCoA and use weighted unifrac metric

ord <- ordinate(physeq\_rar\_RA, method = "PCoA",

distance = "wunifrac")

# Plot data

plot\_ordination(physeq\_rar\_RA,

ord,

type = "sample",

color = "LTSP.Treatment",

title = "PCoA (Weighted Unifrac)") +

# Adding text to plot

annotate(geom = "text",

label = ".",

x = - 0.025,

y = 0.025,

size = 4) +

# Manually adjust colours for points

scale\_colour\_manual(values = c("red", "orange",

"forestgreen", "blue"),

labels = c("OM1", "OM2", "OM3", "REF")) +

stat\_ellipse(type = "norm", size = 1) +

guides(colour = guide\_legend("LTSP Treatment")) +

theme\_bw(base\_size = 14)