Microbial community analysis

Workflow has been built with help of "Orchestrating microbiome analysis with R and Bioconductor" (Leo Lahti et. al), various analysis tool guides and own modifications.

```
#load required libraries
library(mia); packageVersion("mia")
```

```
[1] '1.12.0'
```

```
library(ape); packageVersion("ape")
```

```
[1] '5.8'
```

```
library(miaViz);packageVersion("miaViz")
```

```
[1] '1.12.0'
```

```
library(scater);packageVersion("scater")
```

```
[1] '1.32.1'
```

```
library(vegan);packageVersion("vegan")
```

```
[1] '2.6.6.1'
```

```
library(tidyverse); packageVersion("tidyverse")
```

```
[1] '2.0.0'
```

```
library(kableExtra);packageVersion("kableExtra")
```

```
[1] '1.4.0'
```

```
library(dplyr);packageVersion("dplyr")
```

```
[1] '1.1.4'
```

```
library(tibble);packageVersion("tibble")
```

```
[1] '3.2.1'
```

```
library(knitr);packageVersion("knitr")
```

```
[1] '1.48'
```

```
library(reshape2);packageVersion("reshape2")
```

```
[1] '1.4.4'
```

```
library(scales);packageVersion("scales")
```

```
[1] '1.3.0'
```

```
library(ggplot2);packageVersion("ggplot2")
```

```
[1] '3.5.1'
```

```
library(ggthemes);packageVersion("ggthemes")
```

```
[1] '5.1.0'
```

```
library(ggsci);packageVersion("ggsci")
```

```
[1] '3.2.0'
```

```
library(patchwork)
library(ALDEx2);packageVersion("ALDEx2")
```

```
[1] '1.36.0'
```

```
library(ANCOMBC);packageVersion("ANCOMBC")
```

```
[1] '2.6.0'
```

```
library(DT);packageVersion("DT")
```

```
[1] '0.33'
```

```
library(Maaslin2);packageVersion("Maaslin2")
```

```
[1] '1.18.0'
```

Set file locations

Set necessary file paths before running code.

```
# Path variables
asvfile <- "result_tables/asvs.tsv"
metafile <- "result_tables/metadata.tsv"
taxafile <- "result_tables/taxonomy.tsv"
treefile <- "result_tables/tree.nwk"
```

Import data

Data is imported and a TreeSummarizedExperiment object is created.

```
#Abundance data is imported from tabular txt file, rownames stored and emptied
counts <- read_tsv(asvfile, show_col_types = FALSE)
ASV_names <- counts$ASV_names
counts$ASV_names <- NULL
#Metadata is imported from tabular txt file, rownames stored and emptied
samples <- read_tsv(metafile, show_col_types = FALSE)
sampleid <- samples$sampleid
samples$sampleid <- NULL
#Taxonomy table is imported tabular txt file, rownames stored and emptied
taxonomy <-read_tsv(taxafile, show_col_types = FALSE)
taxanames <- taxonomy$ASV_names
taxonomy$ASV_names <- NULL
#Check if there are discrepancies between data tables
if( any( colnames(counts) != sampleid ) ){
    counts <- counts[ , sampleid ]}
if( any( ASV_names != taxanames ) ){
    counts <- counts[ taxanames, ]}
#Abundance values should be in numeric matrix format
counts <- as.matrix(counts)
#And should be added to a SimpleList
assays <-  SimpleList(counts = counts)
#colData and rowData should be in DataFrame format
colData <- DataFrame(colData)
rowData <- DataFrame(rowData)
#Create a TreeSummarized Experiment object
tse <- TreeSummarizedExperiment(assays = assays,
                                    colData = samples,
                                    rowData = taxonomy)
#Add amplicon variant names as rownames
rownames(tse) <- ASV_names
```

Add phylogenetic tree

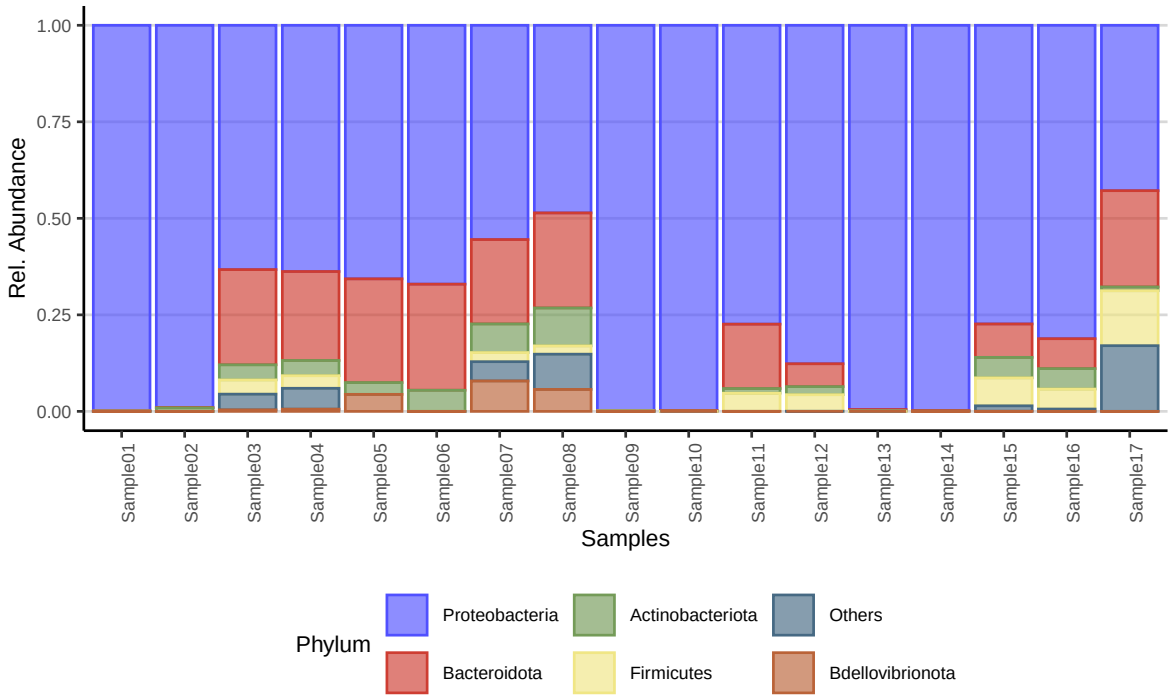
```
#tree in newick format was created with mafft & fasttree
phytree <- read.tree(treefile)
rowTree(tse) <- phytree
#view tse
#save as rds object
saveRDS(tse,"rds/tse.rds")
```

Community composition

Community composition can be visualised at different taxonomic ranks by agglomerating information and using getTopFeatures function. Barplots can be created either by arranging assay data to a long data table or straight by using plotAbundance function from miaViz package.

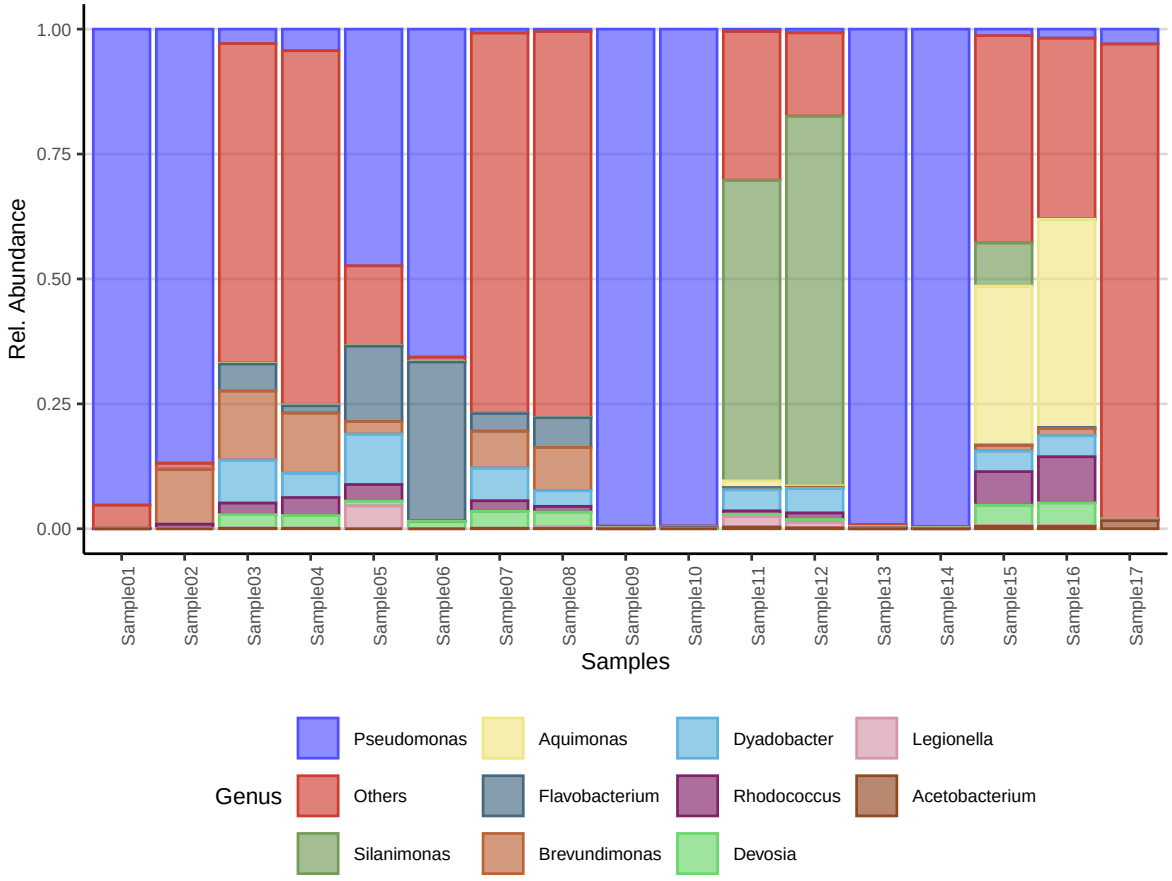
Here we plot top 5 phyla from samples. Rest have been relabeled to Others group.

```
n <- 5
p_level <- agglomerateByRank(tse, "Phylum", onRankOnly = TRUE)
p_level <- relAbundanceCounts(p_level, name="relabundance")
#Get top orders
top_phyla <- getTopFeatures(p_level,
                            top = n,
                            method="median",
                            assay_name = "relabundance")
#Leave only names for top phyla and label the rest to "Others"
phyla_renamed <-  lapply(rowData(p_level)$Phylum,
                  function(x){if (x %in% top_phyla) {x} else {"Others"}})
rowData(p_level)$Phylum <- as.character(phyla_renamed)
#Barplot object
abund1 <- plotAbundance(p_level,
              assay_name = "relabundance",
              rank = "Phylum",
              order_rank_by = "abund",
              add_x_text = TRUE,
              one_facet = TRUE) + labs(color = "Phylum", fill = "Phylum")
abund1$scales$scales <- NULL
abund1 + theme_hc(base_size = 9) +
  scale_fill_igv("default") + scale_color_igv("default") +
  theme(axis.text.x = element_text(angle = 90))
```



Top 10 genera

```
n <- 10
g_level <- agglomerateByRank(tse, "Genus", onRankOnly = TRUE)
g_level <- relAbundanceCounts(g_level, name="relabundance")
#Get top orders
top_genera <- getTopFeatures(g_level,
                          top = n,
                          method = "median",
                          assay_name = "relabundance")
#Leave only names for top phyla and label the rest to "Others"
genera_renamed <-  lapply(rowData(g_level)$Genus,
                    function(x){if (x %in% top_genera) {x} else {"Others"}})
rowData(g_level)$Genus <- as.character(genera_renamed)
#Plot composition as a bar plot
abund2 <- plotAbundance(g_level,
              assay_name = "relabundance",
              rank = "Genus",
              order_rank_by = "abund",
              add_x_text = TRUE,
              one_facet = TRUE) + labs(fill="Genus", color="Genus")
abund2$scales$scales <- NULL
abund2 + theme_hc(base_size = 9) + scale_fill_igv("default") +
              scale_color_igv("default") + theme(axis.text.x = element_text(angle = 90))
```



Abundance information can be presented also in tables. Next, samples are merged to groups, taxonomy agglomerated and arranged by abundance.

Top taxa in filtered samples vs non-filtered.

```
#Merge filtered values, recount relative abundance and agglomeration to Genus
col4 <- mergeCols(tse,colData(tse)$Filtered)
col4 <- agglomerateByRank(col4, "Genus")
col4 <- relAbundanceCounts(col4)
#Create data frames for merged groups
opt1 <- data.frame(assay(col4,"relabundance")) %>%
                          rownames_to_column(var = "opt1_asv") %>%
  arrange(desc(no)) %>% dplyr::select('Non-filtered' = opt1_asv,
                                      'Rel Abundance'= no)
opt2 <- data.frame(assay(col4,"relabundance")) %>%
                        rownames_to_column(var = "opt2_asv") %>%
  arrange(desc(yes)) %>% dplyr::select(Filtered = opt2_asv,
                                       'Rel Abundance' = yes)
#How many to list in table
n <- 10
col4_table <- cbind(opt1[1:n,], opt2[1:n,])
kable(col4_table, digits = 2, caption = "Common taxa") %>%
  kable_styling(latex_options = c("HOLD_position","striped"), font_size = 12) %>%
  row_spec(0, background = "indigo", color = "ivory")
```

Table 1: Common taxa

Non-filtered	Rel Abundance	Filtered	Rel Abundance
Family:Rhodobacteraceae	0.15	Genus:Pseudomonas	0.63
Genus:Silanimonas	0.09	Genus:Flavobacterium	0.10
Family:Comamonadaceae	0.08	Family:Pseudomonadaceae	0.07
Genus:Brevundimonas	0.04	Family:Sphingomonadaceae	0.04
Genus:Pedobacter	0.03	Family:Comamonadaceae	0.04
Genus:Dyadobacter	0.03	Genus:Brevundimonas	0.02
Family:Rhizobiaceae	0.03	Genus:Dyadobacter	0.02
Genus:Algoriphagus	0.03	Family:Microbacteriaceae	0.02
Genus:Aquimonas	0.03	Genus:Chryseobacterium	0.02
Genus:Alishewanella	0.02	Genus:Peredibacter	0.01

Top taxa in microalgae categories.

```
#Merge filtered values, recount relative abundance and agglomeration to Genus
col5 <- mergeCols(tse, colData(tse)$Algae)
col5 <- agglomerateByRank(col5, "Genus")
col5 <- relAbundanceCounts(col5)
#Create data frames for merged groups
opt1 <- data.frame(assay(col5,"relabundance")) %>%
                          rownames_to_column(var = "opt1_asv") %>%
  arrange(desc(chlorella_s)) %>% dplyr::select(Chlorella = opt1_asv,
                                               'Rel Abundance'= chlorella_s)
opt2 <- data.frame(assay(col5,"relabundance")) %>%
                        rownames_to_column(var = "opt2_asv") %>%
  arrange(desc(selenastrum)) %>% dplyr::select(Selenatrum = opt2_asv,
                                               'Rel Abundance' = selenastrum)
#How many to list in table
n <- 10
col5_table <- cbind(opt1[1:n,], opt2[1:n,])
kable(col5_table, digits = 2, caption = "Common taxa") %>%
  kable_styling(latex_options = c("HOLD_position", "striped"), font_size = 12) %>%
  row_spec(0, background = "indigo", color = "ivory")
```

Table 2: Common taxa

Chlorella	Rel Abundance	Selenatrum	Rel Abundance
Genus:Pseudomonas	0.43	Genus:Pseudomonas	0.29
Genus:Silanimonas	0.09	Genus:Flavobacterium	0.10
Family:Rhodobacteraceae	0.08	Family:Comamonadaceae	0.07
Family:Pseudomonadaceae	0.06	Family:Rhodobacteraceae	0.06
Family:Sphingomonadaceae	0.05	Genus:Dyadobacter	0.03
Genus:Brevundimonas	0.04	Family:Microbacteriaceae	0.03
Family:Comamonadaceae	0.03	Family:Pseudomonadaceae	0.03
Genus:Algoriphagus	0.03	Genus:Chryseobacterium	0.02
Genus:Alishewanella	0.02	Genus:Brevundimonas	0.02
Genus:Dyadobacter	0.02	Family:Rhizobiaceae	0.02

Top taxa in culture age categories (Note that numeric values need to be converted to characters).

```
#Merge filtered values, recount relative abundance and agglomeration to Genus
col3 <- mergeCols(tse, as.character(colData(tse)$Age))
col3 <- agglomerateByRank(col3, "Genus")
col3 <- relAbundanceCounts(col3)
#Create data frames for merged groups
opt1 <- data.frame(assay(col3, "relabundance")) %>%
                          rownames_to_column(var = "opt1_asv") %>%
  arrange(desc(X5)) %>% dplyr::select(Day5 = opt1_asv, 'Rel Abundance'= X5)
opt2 <- data.frame(assay(col3, "relabundance")) %>%
                        rownames_to_column(var = "opt2_asv") %>%
  arrange(desc(X30)) %>% dplyr::select(Day30 = opt2_asv, 'Rel Abundance' = X30)
#How many to list in table
n <- 10
col3_table <- cbind(opt1[1:n,], opt2[1:n,])
kable(col3_table, digits = 2, caption = "Common taxa") %>%
  kable_styling(latex_options = c("HOLD_position","striped"), font_size = 12) %>%
  row_spec(0, background="indigo", color="ivory")
```

Table 3: Common taxa

Day5	Rel Abundance	Day30	Rel Abundance
Genus:Pseudomonas	0.34	Genus:Pseudomonas	0.38
Genus:Flavobacterium	0.08	Family:Rhodobacteraceae	0.21
Family:Comamonadaceae	0.07	Genus:Silanimonas	0.16
Family:Pseudomonadaceae	0.05	Genus:Aquimonas	0.05
Genus:Brevundimonas	0.04	Genus:Algoriphagus	0.03
Family:Sphingomonadaceae	0.03	Genus:Roseococcus	0.02
Genus:Dyadobacter	0.03	Family:Clostridiaceae	0.02
Family:Microbacteriaceae	0.02	Genus:Dyadobacter	0.02
Family:Rhodobacteraceae	0.02	Family:Comamonadaceae	0.01
Genus:Chryseobacterium	0.02	Genus:Rhodococcus	0.01

Alpha diversity

Diversity can be studied using diversity indexes. Values can be added to **colData** under defined names. We create table with Shannon, Faith and observed features diversity indexes.

```
#Calculate Shannon index
tse <- mia::estimateDiversity(tse,
                              assay_name = "counts",
                              index = "shannon",
                              name = "Shannon")
#Calculate phylogenetic Faith index
tse <- mia::estimateFaith(tse,
                          abund_values = "counts",
                          index = "faith",
                          name = "Faith",
                          tree_name = "phylo")
#Calculate richness with Chao1 index
tse <- mia::estimateRichness(tse,
                              abund_values = "counts",
                              index = "observed",
                              name = "Observed")
#Create table
kable(data.frame(Shannon = colData(tse)$Shannon, Faith = colData(tse)$Faith,
                 Observed_features = colData(tse)$Observed), digits = 2) %>%
                 kable_styling(latex_options = c("HOLD_position", "striped"),
                               font_size = 12) %>%
                 row_spec(0, background = "indigo", color = "ivory")
```

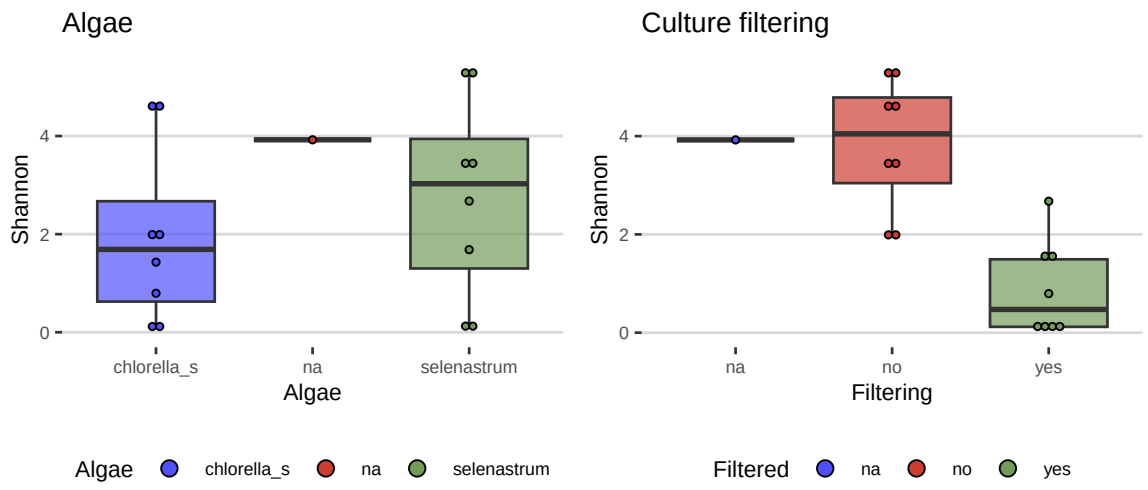
	Shannon	Faith	Observed_features
Sample01	0.79	7.71	40
Sample02	1.43	7.71	51
Sample03	4.64	18.86	353
Sample04	4.58	19.61	362
Sample05	2.67	11.33	65
Sample06	1.68	12.12	52
Sample07	5.34	34.92	718
Sample08	5.24	34.36	690
Sample09	0.12	7.18	33
Sample10	0.12	6.34	29
Sample11	2.03	9.13	92
Sample12	1.95	10.44	102
Sample13	0.15	6.66	29
Sample14	0.10	6.63	31
Sample15	3.51	18.13	194
Sample16	3.38	17.47	164
Sample17	3.92	26.79	411

Boxplots can be used to compare sample categories

```
#Shannon boxplot I
plot1 <- ggplot(as.data.frame(colData(tse)), aes(x = Algae, y = Shannon,
                                                 fill = Algae)) +
  geom_boxplot(alpha = 0.7, show.legend = FALSE) +
  geom_dotplot(binaxis = "y", stackdir = "center", binwidth = 0.3, dotsize = 0.5) +
  labs (title = "Algae", y = "Shannon", x = "Algae")
#Shannon boxplot II
plot2 <- ggplot(as.data.frame(colData(tse)), aes(x = Filtered, y = Shannon,
                                                 fill = Filtered)) +
  geom_boxplot(alpha = 0.7, show.legend = FALSE) +
  geom_dotplot(binaxis = "y", stackdir = "center", binwidth = 0.3, dotsize = 0.5) +
  labs (title = "Culture filtering", y = "Shannon", x = "Filtering")
#Shannon boxplot III
plot3 <- ggplot(as.data.frame(colData(tse)), aes(x = Name, y = Shannon,
                                                 fill = Name)) +
  geom_boxplot(alpha = 0.7, show.legend = FALSE) +
  geom_dotplot(binaxis = "y", stackdir = "center", binwidth = 0.3, dotsize = 0.5) +
  labs (title = "Samples", y = "Shannon", x = "Sample")
#Shannon boxplot IV
plot4 <- ggplot(as.data.frame(colData(tse)), aes(x = as.character(Age), y = Shannon,
                                                 fill = Algae)) +
               geom_boxplot(alpha = 0.7, show.legend = FALSE) +
  geom_dotplot(binaxis = "y", stackdir = "center", binwidth = 0.3, dotsize = 0.5) +
  labs (title = "Algae and age", y = "Shannon", x = "Age of culture") +
  scale_x_discrete(limits = rev)
#Shannon boxplot V
plot5 <- ggplot(as.data.frame(colData(tse)), aes(x = as.character(Age), y = Shannon,
                                                 fill= Filtered)) +
  geom_boxplot(alpha = 0.7, show.legend = FALSE) +
  geom_dotplot(binaxis = "y", stackdir = "center", binwidth = 0.3, dotsize = 0.5) +
  labs (title = "Filtering and age", y = "Shannon", x = "Age of culture") +
  scale_x_discrete(limits = rev)
```

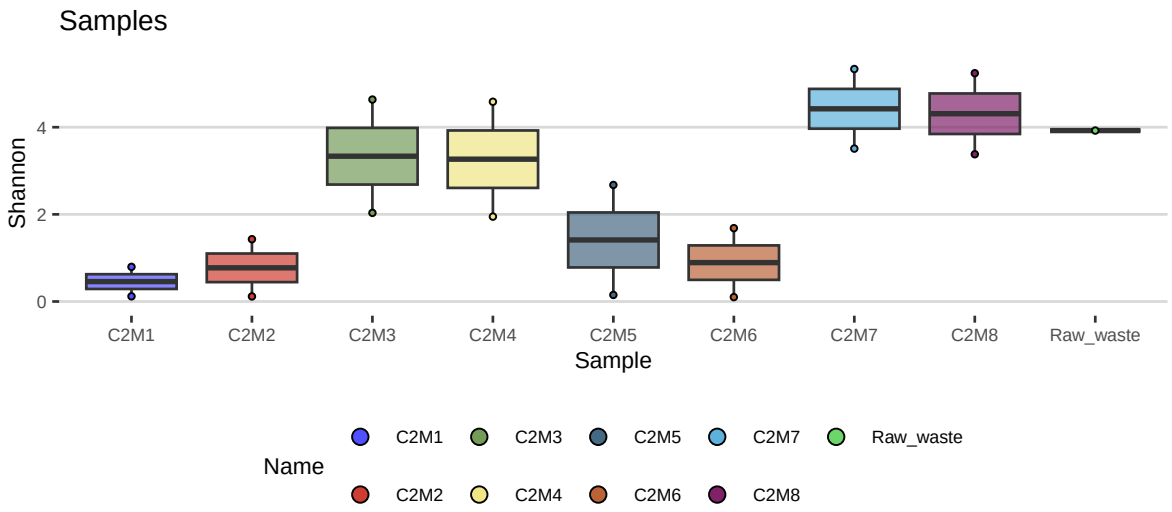
Algae and culture filtering boxplots (n=8).

```
plot1 +  theme_hc(base_size=9) + scale_fill_igv() + plot2 + theme_hc(base_size=9) + scale_fill_igv()
```



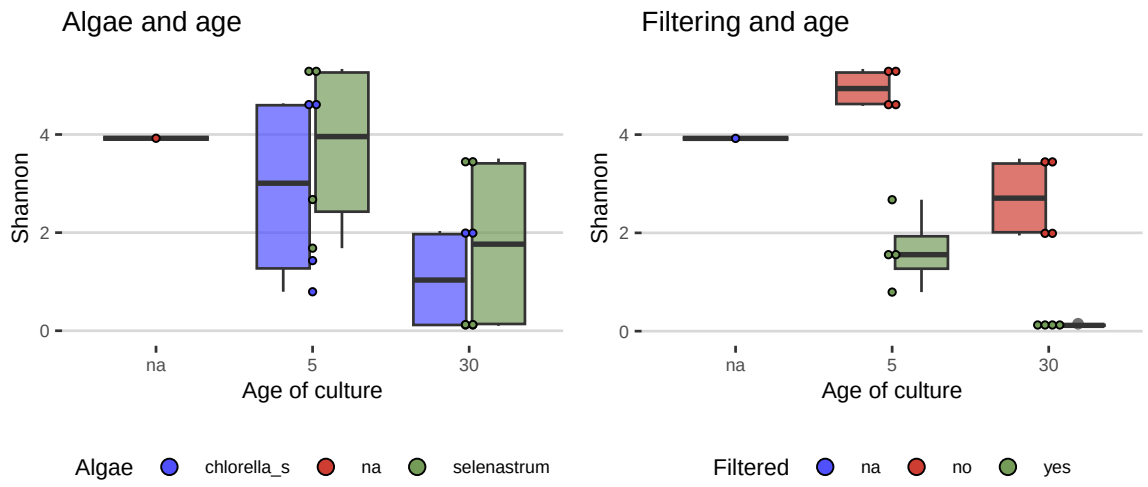
Sample comparison boxplot (n=2)

```
plot3 + theme_hc(base_size = 9) + scale_fill_igv()
```



Dual boxplots (n=4)

```
plot4 + theme_hc(base_size = 9) + scale_fill_igv() + plot5 + theme_hc(base_size = 9) + scale_fill_igv()
```



Filtering has an effect on diversity. Both microalgae seem also to decrease diversity compared to untreated. However, there is only one control sample and decrease is not statistically significant. Culture age also decreases diversity. This is most evident in final boxplots.

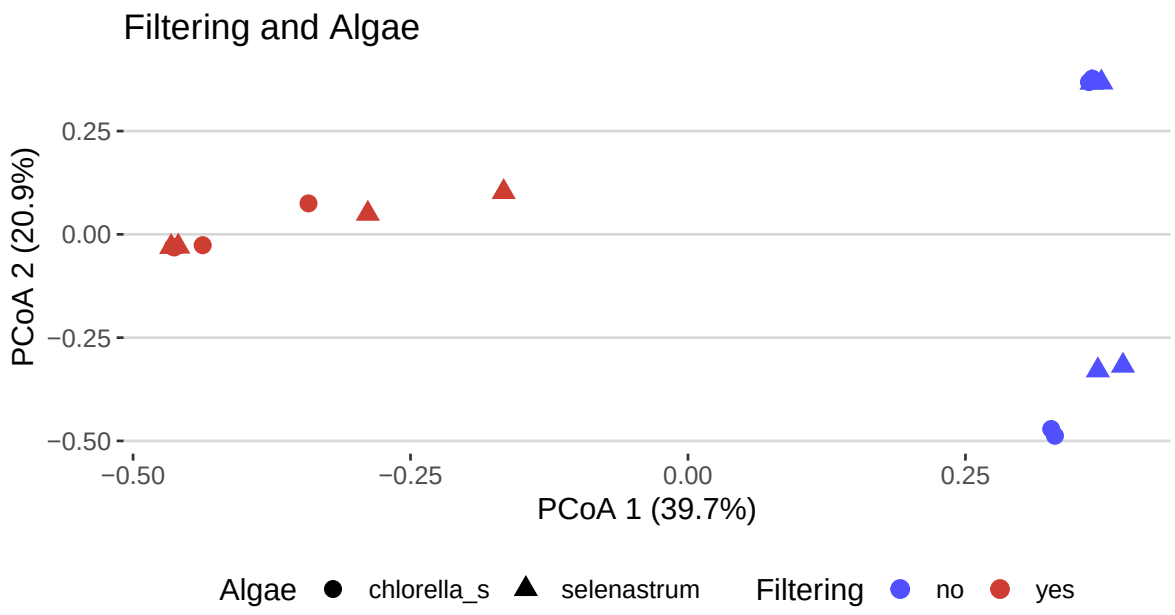
Beta diversity

Bray-Curtis dissimilarity analysis

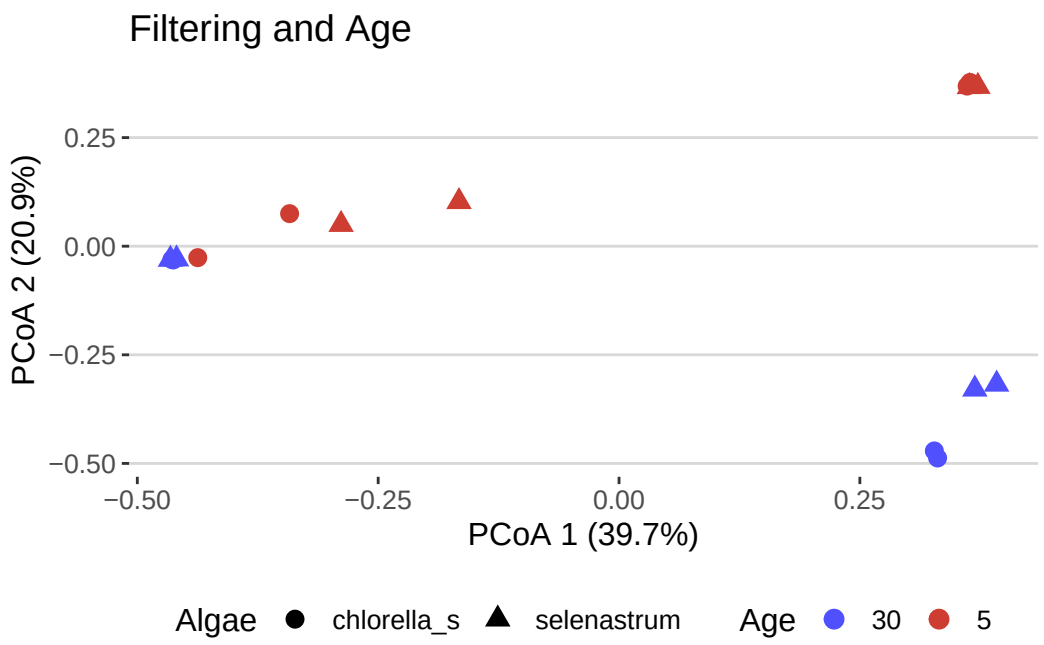
```
# Convert counts to relabundance
tse <- transformAssay(tse, method = "relabundance", assay.type = "counts")
# Perform Bray-Curtis distance calculation
tse <- runMDS(tse, FUN = vegan::vegdist, method = "bray",
              name = "Bray", exprs_values = "relabundance")
#Create 2D ggplot object
pcoa_bray <- plotReducedDim(tse, "Bray")
# Calculate explained variance
e <- attr(reducedDim(tse, "Bray"), "eig");
rel_eig <- e/sum(e[e>0])
#Create dataframe for each axis
bray_curtis_df <- data.frame(pcoa1 = pcoa_bray$data[,1],
                             pcoa2 = pcoa_bray$data[,2])
#Binding sample attributes to same data frame
#At same time culture age in Days is converted from numeric to character
bray_attributes <-cbind(bray_curtis_df,
                        Filtering = colData(tse)$Filtered,
                        Age = as.character(colData(tse)$Age),
                        Names = colData(tse)$Name,
                        Algae = colData(tse)$Algae)
bray_attributes <- bray_attributes[1:16,]
#Create series of plots using combined data frame
filtering <- ggplot(data = bray_attributes, aes(x = pcoa1, y = pcoa2,
                    color = Filtering, shape = Algae)) + geom_point(size = 3) +
                    labs(x = paste("PCoA 1 (", round(100 * rel_eig[[1]],1),
                    "%", ")", sep = ""), y = paste("PCoA 2 (",
                    round(100 * rel_eig[[2]],1), "%", ")", sep = ""),)
age <- ggplot(data = bray_attributes, aes(x = pcoa1, y = pcoa2, color = Age,
                     shape = Algae)) + geom_point(size = 3) +
                     labs(x = paste("PCoA 1 (", round(100 * rel_eig[[1]],1),
                     "%", ")", sep = ""), y = paste("PCoA 2 (",
                    round(100 * rel_eig[[2]],1), "%", ")", sep = ""),)
```

Results.

```
filtering + theme_hc() + ggtitle("Filtering and Algae") +
           theme(axis.title = element_text()) + scale_color_igv()
```



```
age + theme_hc() + ggtitle("Filtering and Age") +
            theme(axis.title = element_text()) + scale_color_igv()
```



Filtering change community composition.

Beta diversity using unifrac

```
tse <- runMDS(tse, FUN = mia::calculateUnifrac, name = "unweighted_uni",
              tree = rowTree(tse),
              ntop = nrow(tse),
              exprs_values = "relabundance",
              weighted = FALSE)
tse <- runMDS(tse, FUN = mia::calculateUnifrac, name = "weighted_uni",
              tree = rowTree(tse),
              ntop = nrow(tse),
              exprs_values = "relabundance",
              weighted = TRUE)
#Create ggplot objects
unweighted <- plotReducedDim(tse, "unweighted_uni")
weighted <- plotReducedDim(tse, "weighted_uni")
#Create data frames
unweighted_df <- data.frame(pcoa1 = unweighted$data[,1],
                            pcoa2 = unweighted$data[,2])
weighted_df <- data.frame(pcoa1 = weighted$data[,1],
                          pcoa2 = weighted$data[,2])
#We  want to include sample metadata to the same data frame
#At same time culture age in Days is converted from numeric data to character
unweighted_attributes <-cbind(unweighted_df,
                        Filtering = colData(tse)$Filtered,
                        Age = as.character(colData(tse)$Age),
                        Names = colData(tse)$Name,
                        Algae = colData(tse)$Algae,
                        Group = colData(tse)$Group)
weighted_attributes <-cbind(weighted_df,
                        Filtering = colData(tse)$Filtered,
                        Age = as.character(colData(tse)$Age),
                        Names = colData(tse)$Name,
                        Algae = colData(tse)$Algae,
                        Group = colData(tse)$Group)
# Calculate explained variances
```

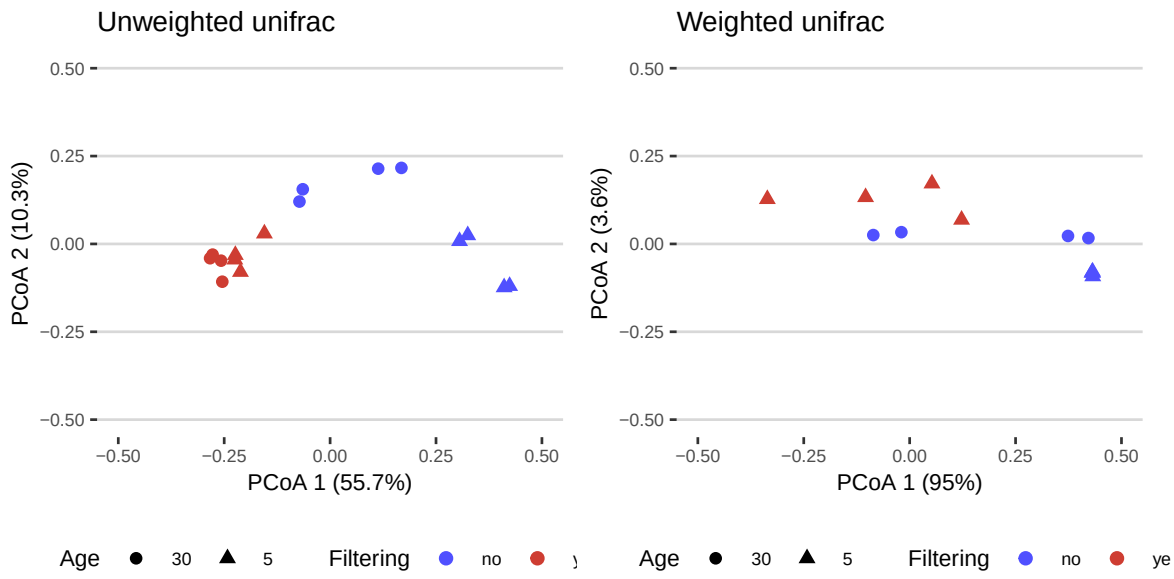
```
eu <- attr(reducedDim(tse, "unweighted_uni"), "eig");
urel_eig <- eu/sum(eu[eu>0])
ew <- attr(reducedDim(tse, "weighted_uni"), "eig");
wrel_eig <- ew/sum(ew[ew>0])
# Removing wastewater from plots
unweighted_attributes <- unweighted_attributes[1:16,]
weighted_attributes <- weighted_attributes[1:16,]
```

Next create ggplot objects

```
#Create series of plots using combined data frame
uni1 <- ggplot(data = unweighted_attributes,
        aes(x = pcoa1, y = pcoa2, color = Filtering, shape = Age)) +
        geom_point(size = 2) +
        labs(x = paste("PCoA 1 (", round(100 * urel_eig[[1]],1), "%", ")", sep = ""),
        y = paste("PCoA 2 (", round(100 * urel_eig[[2]],1), "%", ")", sep = ""),
        title = "Unweighted unifrac") +
        scale_y_continuous(limits = c(-0.5, 0.5)) +
        scale_x_continuous(limits = c(-0.5, 0.5)) +
        theme_hc(base_size = 9) + scale_color_igv()
uni2 <- ggplot(data = weighted_attributes,
        aes(x = pcoa1, y = pcoa2, color = Filtering, shape = Age)) +
        geom_point(size = 2) +
        labs(x = paste("PCoA 1 (", round(100 * wrel_eig[[1]],1), "%", ")", sep = ""),
        y = paste("PCoA 2 (", round(100 * wrel_eig[[2]],1), "%", ")", sep = ""),
        title = "Weighted unifrac") +
        scale_y_continuous(limits = c(-0.5, 0.5)) +
        scale_x_continuous(limits = c(-0.5, 0.5)) +
        theme_hc(base_size = 9) + scale_color_igv()
uni3 <- ggplot(data = unweighted_attributes,
        aes(x = pcoa1, y = pcoa2, color = Algae, shape = Age)) +
        geom_point(size = 2) +
        labs(x = paste("PCoA 1 (", round(100 * urel_eig[[1]],1), "%", ")", sep = ""),
        y = paste("PCoA 2 (", round(100 * urel_eig[[2]],1), "%", ")", sep = ""),
        title = "Unweighted unifrac") +
        scale_y_continuous(limits = c(-0.5, 0.5)) +
        scale_x_continuous(limits = c(-0.5, 0.5)) +
        theme_hc(base_size = 9) + scale_color_igv() +
        theme(legend.position = "bottom", legend.box = "vertical",
              legend.margin = margin())
uni4 <- ggplot(data = weighted_attributes,
        aes(x=pcoa1, y=pcoa2, color = Algae, shape = Age)) +
        geom_point(size=2) +
        labs(x = paste("PCoA 1 (", round(100 * wrel_eig[[1]],1), "%", ")", sep = ""),
        y = paste("PCoA 2 (", round(100 * wrel_eig[[2]],1), "%", ")", sep = ""),
        title = "Weighted unifrac") +
        scale_y_continuous(limits = c(-0.5, 0.5)) +
        scale_x_continuous(limits = c(-0.5, 0.5)) +
        theme_hc(base_size = 9) + scale_color_igv() +
        theme(legend.position = "bottom", legend.box = "vertical",
              legend.margin = margin())
```

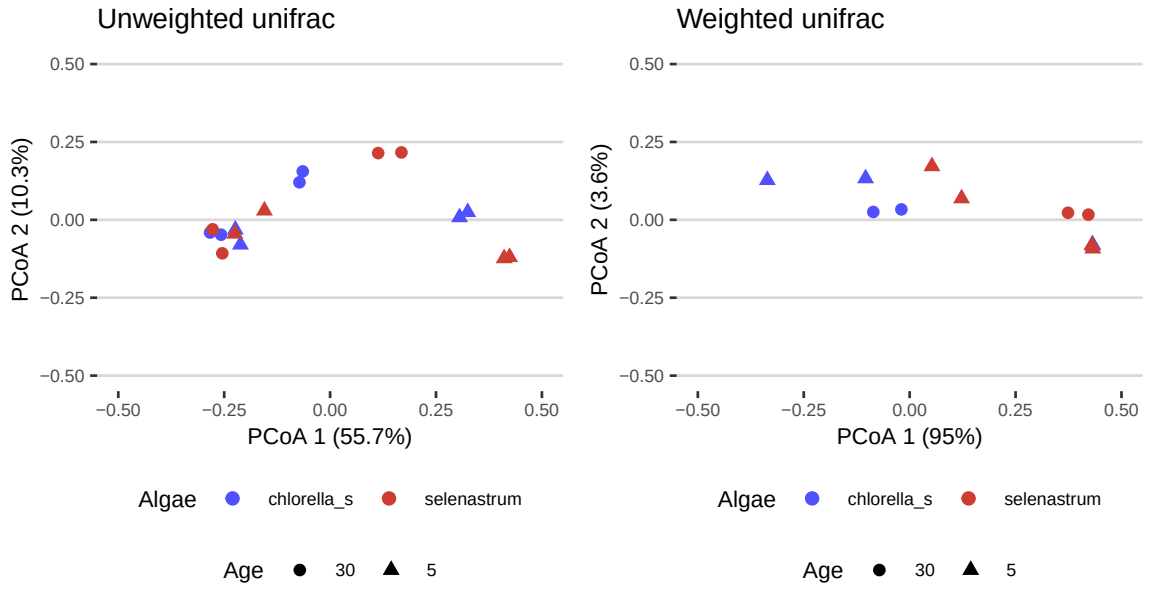
First, comparison of culture filtering and age

```
uni1 + uni2
```



Next algae species along with culture age

```
uni3 + uni4
```



Results are similar to Bray-Curtis. Weighted plots show smaller differences.

Permanova analysis

Permanova measures importance of each variable to total variance.

```
#dbRDA assay
#tse <- transformAssay(tse, method = "relabundance")
tse <- runRDA(tse, assay.type = "relabundance",
                   formula = assay ~ Filtered + Age + Algae,
                   distance = "bray",
                   na.action = na.exclude)
rda_info <- attr(reducedDim(tse, "RDA"), "significance")
kable(rda_info$permanova, digits = 2) %>%
                kable_styling(latex_options = c("HOLD_position", "striped"),
                              font_size = 12) %>%
                row_spec(0, background="indigo", color="ivory")
```

	Df	SumOfSqs	F	Pr(>F)	Total variance	Explained variance
Model	4	3.99	6.10	0.00	5.95	0.67
Filtered	1	2.25	13.76	0.00	5.95	0.38
Age	1	0.89	5.45	0.00	5.95	0.15
Algae	1	0.28	1.72	0.13	5.95	0.05
Residual	12	1.96	NA	NA	5.95	0.33

Filtering is most important factor. Culture age is also statistically significant, while Algae is not.

From same df, we can extract information, if homogeneity assumption is fulfilled.

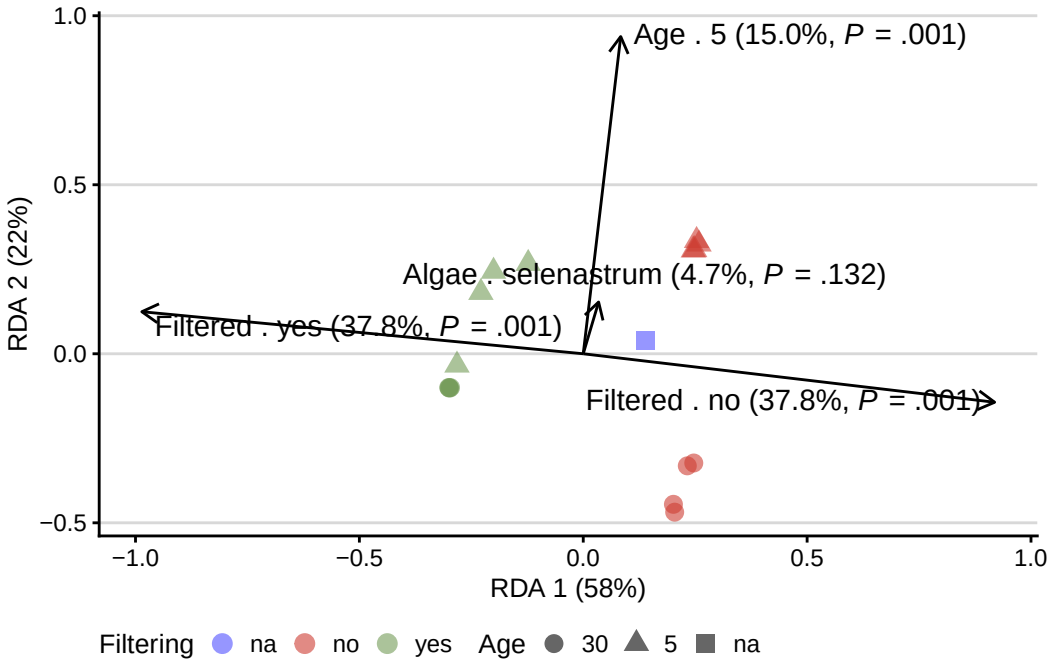
```
kable(rda_info$homogeneity, digits = 2) %>%
                kable_styling(latex_options = c("HOLD_position", "striped"),
                              font_size = 12) %>%
                row_spec(0, background="indigo", color="ivory")
```

	Df	Sum Sq	Mean Sq	F	N.Perm	Pr(>F)	Total variance	Explained variance
Filtered	2	0.38	0.19	3.80	999	0.05	1.09	0.35
Age	2	0.25	0.13	0.96	999	0.59	2.09	0.12
Algae	2	0.29	0.15	7.19	999	0.01	0.58	0.51

Filtering fulfiill homogeneity assumption, Age doesn't

We can also plot results using plotRDA function from miaViz package.

```
# Generate RDA plot
permanova <- plotRDA(tse, "RDA", colour = "Filtered", shape = "Age", add.ellipse = FALSE,
        parse.labels=TRUE)
permanova$scales$scales <- NULL
#change dot size
permanova$layers[[1]]$aes_params$size <-3
#add theme and color palette
permanova  + theme_hc(base_size = 10) +
        scale_colour_igv() + labs(colour="Filtering")
```



Differential abundance

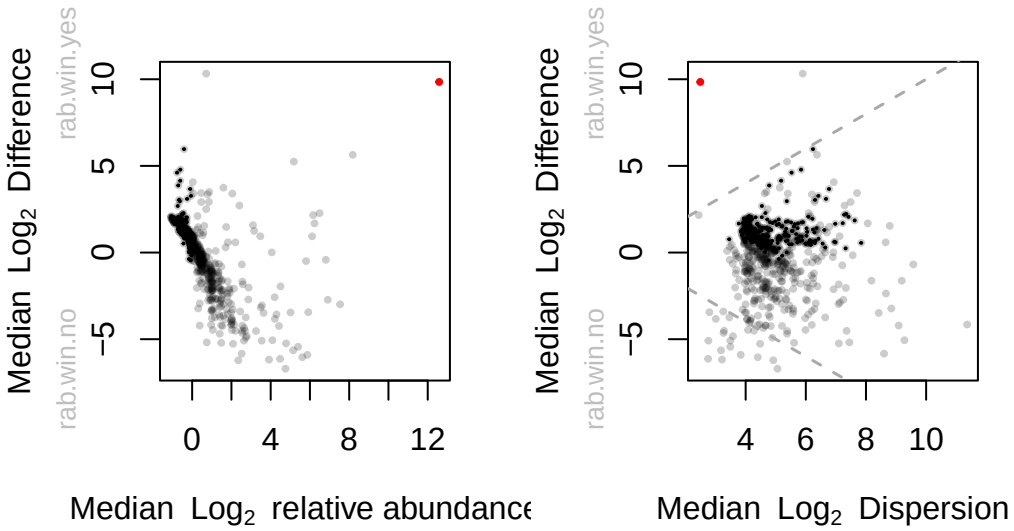
Differential abundances of microbial features can be studied with several R packages. ALDEx2 is one of the them.

Data is preprocessed by removing taxa based on low prevalenc. Also, raw wastewater sample is dropped.

```
#Filtering data based on prevalence.We also drop uncategorized raw wastewater
tse_daa <- subsetByPrevalentTaxa(tse[,1:16], detection = 0, prevalence = 0.1)
#We prepare also list of all taxa labels for later use
featureids <- as.data.frame(getTaxonomyLabels(tse_daa, make_unique=FALSE),
                            rownames(tse_daa))
featureids <- rownames_to_column(featureids, var="asv")
colnames(featureids) <- c("ASV","taxon")
```

Aldex2 analysis on *filtered* category.

```
#ALDEx2 analysis can be performed in modular fashion
#aldex.clr - generates random instances of the centred log-ratio transformed values
filter_aldex <- aldex.clr(assay(tse_daa), tse_daa$Filtered, useMC = TRUE, mc.samples=256, verbose = FALSE)
#aldex.ttest - perform Welch's t and Wilcoxon  test when there are only two conditions
filter_tt <- aldex.ttest(filter_aldex, paired.test = FALSE, verbose = FALSE)
#aldex.effect - estimate effect size and the within and between condition values
filter_effect <- aldex.effect(filter_aldex, CI = TRUE, verbose = FALSE)
#Merge two outputs
filter_aldex_out <- data.frame(filter_tt, filter_effect)
#Create plots
par(mfrow = c(1, 2))
  aldex.plot(filter_aldex_out, type = "MA", test = "welch")
  aldex.plot(filter_aldex_out, type = "MW", test = "welch")
```



In figure, red dots represent significantly changed taxa, grey dots are abundant taxa and black dots are rare taxa.

We have five variants in which wilcoxon probability test result is p <= 0.05.

```
#Filter significantly different taxa and create table
aldex_res <- rownames_to_column(filter_aldex_out, "genus")
aldex_res <- aldex_res %>% dplyr::filter(wi.eBH <= 0.05) %>% dplyr::select(genus, we.eBH, wi.eBH, effect, overlap)
#Merge genus id and taxa names into single table
identity <- merge(aldex_res, featureids, by.x=c("genus"), by.y=c("ASV"))
identity <- identity %>% relocate("genus","taxon","we.eBH","wi.eBH","effect","overlap")
kable(identity, digits=2) %>% kable_styling(latex_options = c("HOLD_position",
                                                              "striped"),
                                            font_size = 12) %>%
  row_spec(0, background="indigo", color="ivory")
```

genus	taxon	we.eBH	wi.eBH	effect	overlap
ASV29	Family:Carnobacteriaceae	0.07	0.01	-1.83	0.00
ASV64	Genus:Brevundimonas	0.05	0.00	-2.08	0.00
ASV70	Genus:Dyadobacter	0.06	0.00	-1.90	0.00
ASV78	Family:Comamonadaceae	0.17	0.03	-1.40	0.03
ASV86	Family:Devosiaceae	0.10	0.01	-1.81	0.00

Testing algae types or culture age did not provide significantly different features

Ancom-BC2 (Analysis of Compositions of Microbiomes with Bias Correction)

```
#Perform the analysis
ancom_out = ancombc2(data = tse_daa, assay_name = "relabundance",
              fix_formula = "Filtered + Algae + Age",
              p_adj_method = "holm", prv_cut = 0, lib_cut = 0,
              group = "Filtered", struc_zero = TRUE, global = TRUE)
saveRDS(ancom_out, "rds/ancom_out.rds")
```

```
ancom_out <- readRDS("rds/ancom_out.rds")
```

Results are collected into out$res data frame. We can filter statistically significant variants.

First variable is Filtering (yes/no)

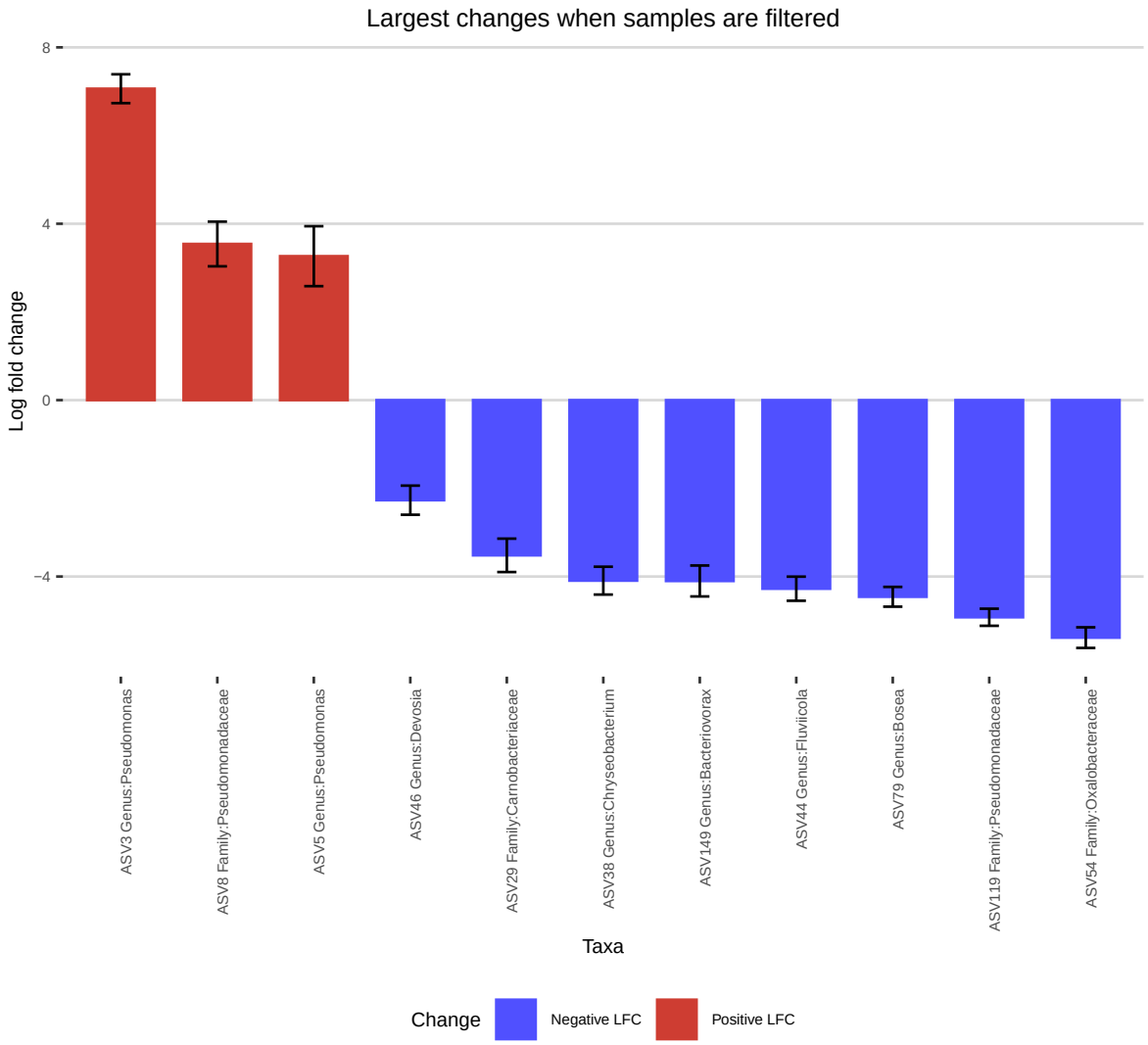
```
#Create data frame, filter Diff = TRUE and arrange by Lfc
#We also combine taxaid and taxonomic name into first column for our figure
df_filtered <- data.frame(ASV = ancom_out$res$taxon, Lfc =ancom_out$res$lfc_Filteredyes, SE =
                          ancom_out$res$se_Filteredyes, Q = ancom_out$res$q_Filteredyes,
                          Diff = ancom_out$res$diff_Filteredyes) %>%
  filter(Diff == "TRUE") %>% arrange(desc(Lfc)) %>% left_join(featureids, by = "ASV")
df_filtered$ASV <- paste(df_filtered$ASV,df_filtered$taxon)
df_filtered <- df_filtered %>% dplyr::select(,-6) %>%  mutate(Change = ifelse(Lfc > 0, "Positive LFC", "Negative LFC"))

kable (df_filtered, caption="Taxa that are changed by filtering",
       digits=2,) %>% kable_styling(latex_options = c("HOLD_position",
                                                      "striped"),
                                    font_size = 12) %>%
  row_spec(0, background="indigo", color="ivory")
```

Table 4: Taxa that are changed by filtering

ASV	Lfc	SE	Q	Diff	Change
ASV3 Genus:Pseudomonas	7.06	0.33	0.00	TRUE	Positive LFC
ASV8 Family:Pseudomonadaceae	3.54	0.51	0.01	TRUE	Positive LFC
ASV5 Genus:Pseudomonas	3.26	0.68	0.04	TRUE	Positive LFC
ASV46 Genus:Devosia	-2.27	0.33	0.04	TRUE	Negative LFC
ASV29 Family:Carnobacteriaceae	-3.52	0.38	0.00	TRUE	Negative LFC
ASV38 Genus:Chryseobacterium	-4.09	0.32	0.02	TRUE	Negative LFC
ASV149 Genus:Bacteriovorax	-4.10	0.35	0.03	TRUE	Negative LFC
ASV44 Genus:Fluviicola	-4.28	0.27	0.01	TRUE	Negative LFC
ASV79 Genus:Bosea	-4.46	0.22	0.03	TRUE	Negative LFC
ASV119 Family:Pseudomonadaceae	-4.92	0.19	0.01	TRUE	Negative LFC
ASV54 Family:Oxalobacteraceae	-5.39	0.23	0.02	TRUE	Negative LFC

Bar plot of log fold changes including standard error

```
#Create ordered taxa list
p_filter <- ggplot(data = df_filtered,
           aes(x = factor(ASV, level=df_filtered$ASV), y = Lfc, fill = Change,
               color = Change)) +
  geom_bar(stat = "identity", width = 0.7,
           position = position_dodge(width = 0.4)) +
  geom_errorbar(aes(ymin = Lfc - SE, ymax = Lfc + SE), width = 0.2,
                position = position_dodge(0.05), color = "black") +
  labs(x = "Taxa", y = "Log fold change",
       title = "Largest changes when samples are filtered") +
  theme_hc(base_size = 8) + scale_fill_igv() + scale_color_igv() +
  theme(plot.title = element_text(hjust = 0.5),
        panel.grid.minor.y = element_blank(),
        axis.text.x = element_text(angle = 90, hjust=1))
p_filter
```



Examination of Algae did not produce significant changes

Culture age variable.

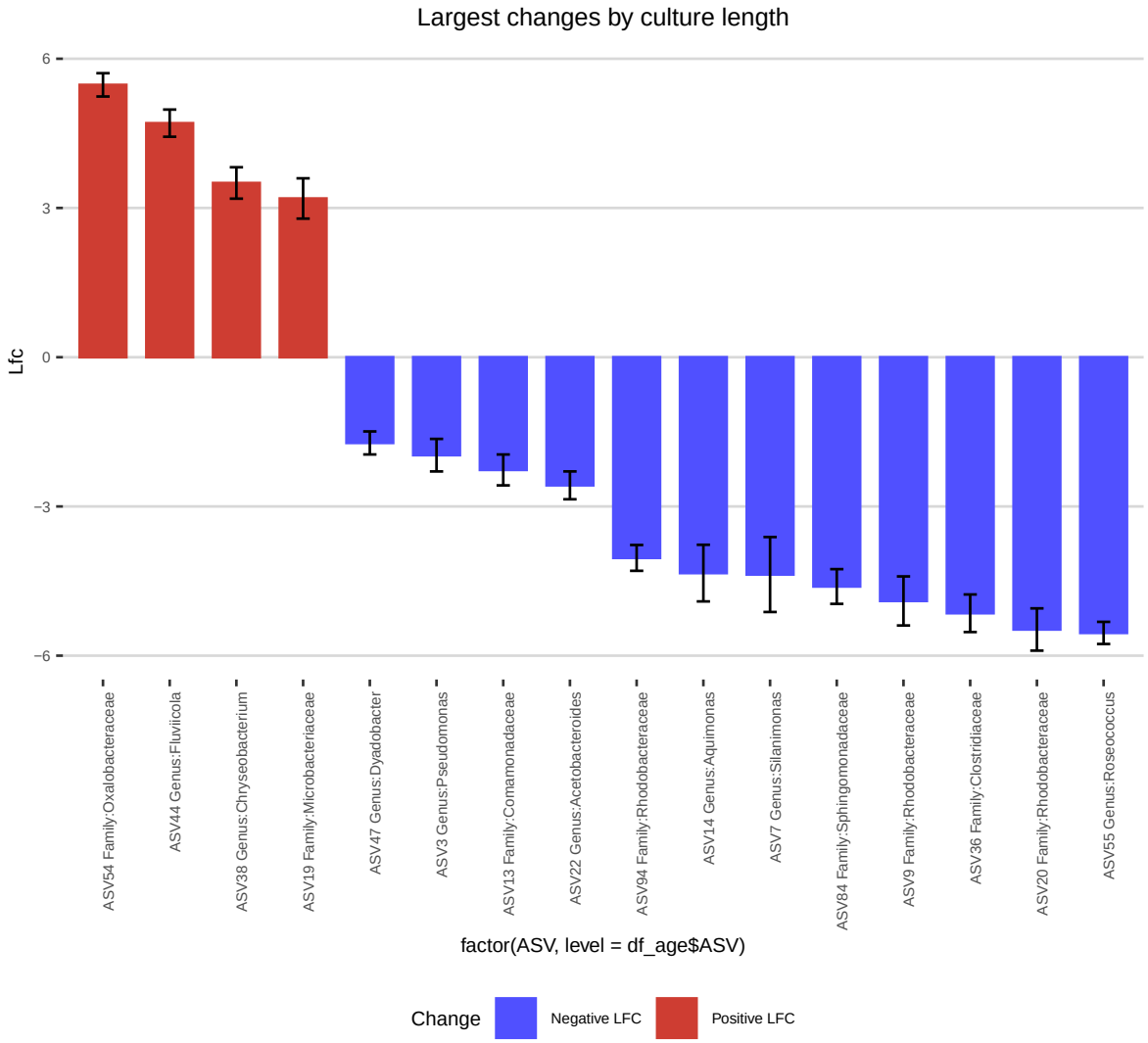
```
#Create new data frame with Lfc, SE, Diff values
df_age <- data.frame(ASV = ancom_out$res$taxon, Lfc =ancom_out$res$lfc_Age5, SE =
                       ancom_out$res$se_Age5, Q = ancom_out$res$q_Age5,
                       Diff = ancom_out$res$diff_Age5) %>%
  filter(Diff == "TRUE") %>% arrange(desc(Lfc)) %>% left_join(featureids, by = "ASV")
df_age$ASV <- paste(df_age$ASV,df_age$taxon)
df_age <- df_age %>% dplyr::select(,-6) %>%  mutate(Change = ifelse(Lfc > 0, "Positive LFC", "Negative LFC"))
kable (df_age, caption="Taxa that are changed by culture length",
       digits=2,) %>% kable_styling(latex_options = c("HOLD_position",
                                                      "striped"),
                                    font_size = 12) %>%
  row_spec(0, background="indigo", color="ivory")
```

Table 5: Taxa that are changed by culture length

ASV	Lfc	SE	Q	Diff	Change
ASV54 Family:Oxalobacteraceae	5.48	0.23	0.02	TRUE	Positive LFC
ASV44 Genus:Fluviicola	4.71	0.27	0.01	TRUE	Positive LFC
ASV38 Genus:Chryseobacterium	3.50	0.32	0.03	TRUE	Positive LFC
ASV19 Family:Microbacteriaceae	3.19	0.41	0.05	TRUE	Positive LFC
ASV47 Genus:Dyadobacter	-1.72	0.23	0.03	TRUE	Negative LFC
ASV3 Genus:Pseudomonas	-1.97	0.33	0.01	TRUE	Negative LFC
ASV13 Family:Comamonadaceae	-2.27	0.31	0.00	TRUE	Negative LFC
ASV22 Genus:Acetobacteroides	-2.58	0.28	0.02	TRUE	Negative LFC
ASV94 Family:Rhodobacteraceae	-4.03	0.26	0.00	TRUE	Negative LFC
ASV14 Genus:Aquimonas	-4.34	0.57	0.00	TRUE	Negative LFC
ASV7 Genus:Silanimonas	-4.37	0.75	0.01	TRUE	Negative LFC
ASV84 Family:Sphingomonadaceae	-4.61	0.35	0.00	TRUE	Negative LFC
ASV9 Family:Rhodobacteraceae	-4.90	0.49	0.00	TRUE	Negative LFC
ASV36 Family:Clostridiaceae	-5.15	0.38	0.00	TRUE	Negative LFC
ASV20 Family:Rhodobacteraceae	-5.47	0.42	0.00	TRUE	Negative LFC
ASV55 Genus:Roseococcus	-5.54	0.22	0.00	TRUE	Negative LFC

Bar plot of LFC including standard error.

```
p_day <- ggplot(data = df_age,
           aes(x = factor(ASV, level=df_age$ASV), y = Lfc,
               fill = Change, color = Change)) +
  geom_bar(stat = "identity", width = 0.7,
           position = position_dodge(width = 0.4)) +
  geom_errorbar(aes(ymin = Lfc - SE, ymax = Lfc + SE), width = 0.2,
                position = position_dodge(0.05), color = "black") +
  labs(title = "Largest changes by culture length") +
  theme_hc(base_size=8) +  scale_fill_igv() + scale_color_igv() +
  theme(plot.title = element_text(hjust = 0.5),
        panel.grid.minor.y = element_blank(),
        axis.text.x = element_text(angle = 90, hjust = 1))
p_day
```



MaAsLin2 package is another DAA analysis package.

```
#Maaslin requires data frame as metadata input
meta_data <- data.frame(colData(tse_daa))
#Counts table needs to be transposed
variant_table <- t(assay(tse_daa))
#Maaslin settings
maaslin_filtering <- Maaslin2(
  variant_table,
  meta_data,
  output = "Maaslin2-filtering",
  transform = "AST",
  fixed_effects = c("Filtered"),
  reference = c("Filtered", "no"),
  normalization = "TSS",
  standardize = FALSE,
  min_prevalence = 0
)
saveRDS(maaslin_filtering, "rds/maaslin_filtering.rds")
```

Note: Maaslin2 will also write results to output folder defined. If you use several fixed effects, it will create additional heatmap plot.

```
maaslin_filtering <- readRDS("rds/maaslin_filtering.rds")
```

Filtering significant results to table by qval value (<= 0.05).

```
maaslin_table <- maaslin_filtering$results %>% dplyr::select(ASV = feature,
                                                             Coef = coef, SE = stderr,qval,
                                                             N, Nonzero = N.not.zero) %>%
  filter(qval <= 0.05) %>% arrange(desc(Coef))
kable(maaslin_table, digits=2) %>% kable_styling(latex_options = c("HOLD_position",
                                                                   "striped"),
                                                 font_size = 12) %>%
  row_spec(0, background="indigo", color="ivory")
```

ASV	Coef	SE	qval	N	Nonzero
ASV3	1.03	0.14	0.00	16	16
ASV382	-0.01	0.00	0.03	16	6
ASV464	-0.01	0.00	0.03	16	6
ASV378	-0.02	0.00	0.03	16	6
ASV98	-0.02	0.00	0.02	16	7
ASV194	-0.02	0.00	0.01	16	7
ASV245	-0.02	0.00	0.00	16	9
ASV249	-0.02	0.01	0.03	16	6
ASV307	-0.02	0.00	0.01	16	7
ASV242	-0.03	0.01	0.03	16	6
ASV239	-0.03	0.01	0.03	16	6
ASV262	-0.03	0.00	0.00	16	8
ASV184	-0.03	0.01	0.03	16	6
ASV193	-0.03	0.01	0.03	16	6
ASV52	-0.03	0.01	0.03	16	10
ASV183	-0.03	0.01	0.03	16	9
ASV85	-0.04	0.01	0.01	16	7
ASV101	-0.04	0.01	0.03	16	7
ASV87	-0.04	0.01	0.01	16	11
ASV119	-0.04	0.01	0.03	16	7
ASV106	-0.05	0.01	0.03	16	8
ASV79	-0.06	0.01	0.03	16	7
ASV78	-0.06	0.01	0.01	16	10
ASV70	-0.06	0.01	0.03	16	8
ASV46	-0.06	0.01	0.03	16	10
ASV64	-0.06	0.01	0.02	16	8
ASV29	-0.09	0.01	0.00	16	12
ASV47	-0.09	0.01	0.00	16	10

Testing algae and age did not provide significantly different variants

For filtering, we can summarize results from different DAA functions and look for common features.

```
#Create daa summaries
aldex_summary <- aldex_res %>% dplyr::select(ASV = genus, Aldex2 = wi.eBH)
ancom_summary <- ancom_out$res %>% dplyr::select(ASV=taxon, Ancombc2 = q_Filteredyes) %>%
  filter(Ancombc2 <= 0.05)
maaslin_summary <- maaslin_filtering$results %>% dplyr::select(ASV=feature,Maaslin2=qval) %>%
  dplyr::filter(Maaslin2 <= 0.05)
#Join three summaries together
daa_summary <- full_join(aldex_summary,ancom_summary, by="ASV")
daa_summary <- full_join(daa_summary, maaslin_summary, by="ASV")
#Create TRUE-FALSE data frame and calculate rowsum score
daa_summary <- daa_summary %>% dplyr::mutate(
   dplyr::across(c(Aldex2:Maaslin2), ~ .x <= 0.05),
    across(-ASV, function(x) ifelse(is.na(x), FALSE, x)),
    Score = rowSums(across(c(Aldex2:Maaslin2)))
) %>% filter(Score > 1)
daa_summary <- daa_summary %>% left_join(featureids, by = "ASV") %>% arrange(ASV)
daa_summary <- daa_summary[c("ASV","taxon", "Aldex2", "Ancombc2","Maaslin2","Score")]
kable(daa_summary, caption="Differential taxa with score of 2") %>%
  kable_styling(latex_options = c("HOLD_position", "striped"),
                font_size = 12) %>%
  row_spec(0, background="indigo", color="ivory")
```

Table 6: Differential taxa with score of 2

ASV	taxon	Aldex2	Ancombc2	Maaslin2	Score
ASV119	Family:Pseudomonadaceae	FALSE	TRUE	TRUE	2
ASV29	Family:Carnobacteriaceae	TRUE	TRUE	TRUE	3
ASV3	Genus:Pseudomonas	FALSE	TRUE	TRUE	2
ASV46	Genus:Devosia	FALSE	TRUE	TRUE	2
ASV64	Genus:Brevundimonas	TRUE	FALSE	TRUE	2
ASV70	Genus:Dyadobacter	TRUE	FALSE	TRUE	2
ASV78	Family:Comamonadaceae	TRUE	FALSE	TRUE	2
ASV79	Genus:Bosea	FALSE	TRUE	TRUE	2