

Shifts in root-associated fungal communities under drought conditions in Ricinus communis L.

Stephanie Hereira-Pacheco, Centro Tlaxcala de Biología de la Conducta, UATx

06 - 04 - 2022

Qiime2 Scripts

Step 1: EXTRACT BARCODES

For this step, It will be used the ‘extract_barcode.py’ script used in qiime1.

```
#I'll use one library called "Ste1" with Ste1_1.fastq and Ste2_1.fastq

extract_barcode.py -f Ste1_1.fastq -r Ste1_2.fastq -c barcode_paired_end \
--bc1_len 8 --bc2_len 8 -o extract_barcode_st1
```

-f : forward reads

-r : reverse reads

-c: input type [default: barcode_single_end]

-bc1_len and -bc2_len : Specify the length, in base pairs, of barcodes

-o : output

Step 2: IMPORT TO QIIME AND DEMULTIPLEX SEQUENCES

For this step, we need to create a directory with the three files output from the previous step, containing:

1. forward.fastq.gz: file that contains the forward sequence reads
2. reverse.fastq.gz: file that contains the reverse sequence reads
3. barcodes.fastq.gz: file that contains the barcode sequence reads

```
qiime tools import \
--type EMPPairedEndSequences \
--input-path extract_barcode_st1 \
--output-path ste1.qza
```

-type : type of file , in this case paired end sequences. Check other import types¹.

-input-path: directory with the files to import

-output-path: artifact name output

And then, we perform the demultiplexing:

```
qiime demux emp-paired --i-seqs ste1.qza \
--m-barcodes-file ../18S/STE1.txt \
--m-barcodes-column BarcodeSequence \
--output-dir demux_STE1 \
--p-no-golay-error-correction
```

¹<https://docs.qiime2.org/2021.4/tutorials/importing/>

- i-seqs : artifact with the import paired end sequences
- m-barcodes-file : mapping file containing information of the sequences
- m-barcodes-column: column name of the Barcode sequences
- output-dir : output directory with the demultiplexed samples and error correction details
- p-no-golay-error-correction: by default perform a correction with a barcode of 12 nt if not use this option (in our case is 16 nt)

Step 3: REMOVE PRIMERS AND VISUALIZATION

```
qiime cutadapt trim-paired \
--i-demultiplexed-sequences demux_STE1/per_sample_sequences.qza \
--p-cores 4 --p-front-f TTAGCATGGAATAATRRAATAGGA \
--p-front-r TCTGGACCTGGTGAGTTCC \
--o-trimmed-sequences demux_STE1_trimmed.qza
```

- i-demultiplexed-sequences : demultiplexed sequences (.qza artifact)
- p-cores : number of threads
- p-front-f : forward primer sequences (front if is in the beginning of the sequences)
- p-front-r : reverse primer sequences (front if is in the beginning of the sequences)
- o-trimmed-sequences : output

```
qiime demux summarize \
--i-data demux_STE1_trimmed.qza
--o-visualization demux_STE1_trimmed.qzv
```

- i-data : demultiplexed and/or trimmed sequences
- o-visualization : output

In this case, due to de the low quality of reverse reads we will continue with the forward sequences and let's set the truncation length of 240 bp.

Step 4: RUN DADA2

In this step, we will perform as an example a loop that can be used in the previous steps and the next ones:

```
for i in demux_STE1_trimmed.qza demux_STE2_trimmed.qza demux_STE3.qza \
demux_STE4.qza demux_STE5_trimmed.qza;
do qiime dada2 denoise-single \
--i-demultiplexed-seqs $i \
--p-trunc-len 240 \
--output-dir dada_single_240_$i; done
```

- i-demultiplexed-seqs : demultiplexed and trimmed sequences
- p-trunc-len 240 : length to trunc sequences to obtain good quality (usually when sequencing drops)
- output-dir : output directory that will contain feature-table and representative sequences

Step 5: MERGING TABLES AND SEQUENCES

First, merge tables and seqs:

```
qiime feature-table merge \
--i-tables dada_single_240_demux_STE1_trimmed/table.qza \
--i-tables dada_single_240_demux_STE2_trimmed/table.qza \
--i-tables dada_single_240_demux_STE3_trimmed/table.qza \
```

```
--i-tables dada_single_240_demux_STE4_trimmed/table.qza \
--i-tables dada_single_240_demux_STE5_trimmed/table.qza \
--o-merged-table merge_table_240.qza
```

-i-tables : table to merge (put every time you want to add a different table)

-o-merged-table : output/merge table

```
qiime feature-table merge-seqs \
--i-data dada_single_240_demux_STE1_trimmed/representative_sequences.qza \
--i-data dada_single_240_demux_STE2_trimmed/representative_sequences.qza \
--i-data dada_single_240_demux_STE3_trimmed/representative_sequences.qza \
--i-data dada_single_240_demux_STE4_trimmed/representative_sequences.qza \
--i-data dada_single_240_demux_STE5_trimmed/representative_sequences.qza \
--o-merged-data merge_seqs_dada_240.qza
```

-i-data : sequences to merge (put every time you want to add a different sequence)

-o-merged-data : output/merge sequences

Then, let's visualize them:

```
qiime feature-table summarize \
--i-table merge_table_240.qza \
--m-sample-metadata-file MAPPINGS/FINALMAP18S \
--o-visualization merge_table_240.qzv \
```

-i-table : merged table

-m-sample-metadata-file : mapping file containing all libraries

-o-visualization : output/ visualization artifact

```
qiime metadata tabulate \
--m-input-file merge_seqs_dada_240.qza \
--o-visualization merge_seqs_dada_240.qzv \
```

-m-input-file : merged sequences

-o-visualization : output/ visualization artifact

Step 6: ASSIGN TAXONOMY

```
qiime feature-classifier classify-consensus-blast \
--i-query merge_seqs_dada_240.qza \
--i-reference-taxonomy /home/steph/Descargas/silva-138-99-tax.qza \
--i-reference-reads /home/steph/Descargas/silva-138-99-seqs.qza \
--o-classification taxonomy_blast_240_0.97.qza --p-perc-identity 0.97
```

classify-consensus-blast : using blast (other options are vsearch and sklearn)

-i-query : seqs merged

-i-reference-taxonomy : artifact imported of taxonomy silva reference database

-i-reference-reads : artifact imported of reads silva reference database

-o-classification output artifact with taxonomy

-p-perc-identity : percent of identity

Step 7: FILTERING AND GROUPING TABLE

- **Removing taxa of Plants**

I checked the feature table and the division Phragmoplastophyta is all assigned to plants

```
qiime taxa filter-table \
--i-table merge_table_240.qza \
--i-taxonomy taxonomy_blast_240_0.97.qza \
--p-exclude Phragmoplastophyta \
--o-filtered-table merge_table_240_noplant.qza
```

-i-table : merge table

-i-taxonomy : taxonomy (from assign taxonomy)

-p-exclude : taxa to exclude

-o-filtered-table : output/artifact

- **Filtering initial treatments and min frequency**

```
qiime feature-table filter-samples \
--i-table merge_table_240_noplant.qza \
--m-metadata-file ../../MAPPINGS/FINALMAP_GROUPED.txt \
--p-where "[Treatments]='T0'" --p-exclude-ids --p-min-frequency 520 \
--o-filtered-table merge_table_240_noplant_filtered.qza
```

-i-table : input table

-m-metadata-file : mapping file

-p-where ("[Treatments]='T0'") : sql code to indicate what column and condition to filter

-p-exclude-ids : to indicate that we will exclude base on the conditions (if not it will retain the data from the condition)

-p-min-frequency : min frequency to retain (sampling depth)

-o-filtered-table : output/table filtered

- **Filtering Uncultivated samples**

```
qiime feature-table filter-samples \
--i-table merge_table_240_noplant_filtered.qza \
--m-metadata-file ../../MAPPINGS/FINALMAP_GROUPED.txt \
--p-where "[Type_of_soil]='Uncultivated'" --p-exclude-ids \
--o-filtered-table merge_table_240_noplant_filtered_nous.qza
```

-i-table : input table

-m-metadata-file : mapping file

-p-where ("[Type_of_soil]='Uncultivated'") : sql code to indicate what column and condition to filter

-p-exclude-ids : to indicate that we will exclude base on the conditions (if not it will retain the data from the condition)

-o-filtered-table : output/table filtered

- **Grouping table (joining replicates and filtering)**

grouped_table_240.qza

```
qiime feature-table group \
--i-table merge_table_240.qza \
--m-metadata-file ../../MAPPINGS/FINALMAP18s.tsv \
```

```
--m-metadata-column group \
--p-mode sum --p-axis sample \
--o-grouped-table grouped_table_240.qza
```

-i-table : input table

-m-metadata-file: metadata file

-m-metadata-column : column name from the metadata

-p-mode : mode of joining samples (in this case, sum the counts, other choices median, mean)

-p-axis : Along which axis to group (it can be features or sample)

-o-grouped-table : output/table

Now, let's filter this grouped table (as we did before, see parameters in the previous steps):

```
qiime taxa filter-table \
--i-table mgrouped_table_240.qza \
--i-taxonomy taxonomy_blast_240_0.97.qza \
--p-exclude Phragmoplastophyta \
--o-filtered-table grouped_table_240_noplant.qza
```

```
qiime feature-table filter-samples \
--i-table grouped_table_240_noplant.qza \
--m-metadata-file ../../MAPPINGS/FINALMAP_GROUPED.txt \
--p-where "[Treatments]='TO'" --p-exclude-ids --p-min-frequency 1500 \
--o-filtered-table grouped_table_240_filt_noplant.qza
```

```
qiime feature-table filter-samples \
--i-table grouped_240_fil_noplant.qza \
--p-where "[Type_of_soil]='Uncultivated'" \
--m-metadata-file ../../MAPPINGS/FINALMAP_GROUPED.txt \
--p-exclude-ids \
--o-filtered-table grouped_240_fil_noplant_nous.qza
```

Step 8: FILTERING SEQUENCES

For this step we will filter the representative sequences base on the table filtered.

```
qiime feature-table filter-seqs \
--i-data seqs_and_taxonomy/merge_seqs_dada_240.qza \
--i-table merge_table_240_noplant_filtered_nous.qza \
--o-filtered-data seqs_and_taxonomy/merge_seqs_dada_240_noplant_filtered_nous.qza

qiime feature-table filter-seqs \
--i-data seqs_and_taxonomy/merge_seqs_dada_240.qza \
--i-table grouped_240_fil_noplant_nous.qza \
--o-filtered-data seqs_and_taxonomy/grouped_seqs_dada_240_noplant_filtered_nous.qza
```

-i-data : input sequences

-i-table : input table use to filter

-o-filtered-data : output/filtered sequences

Step 9: BUILDING THE TREE

For this step we will build the phylogenetic tree *denovo*.

```

qiime phylogeny align-to-tree-mafft-fasttree \
--i-sequences merge_seqs_dada_240_noplant_filtered_nous.qza \
--output-dir tree_merge

qiime phylogeny align-to-tree-mafft-fasttree \
--i-sequences grouped_seqs_dada_240_noplant_filtered_nous.qza \
--output-dir tree_grouped

```

-i-sequences : sequences filtered

-output-dir : output director that will contain the alignment, masked alignment, the tree and the rooted treed.

Plot of the experimental design

```

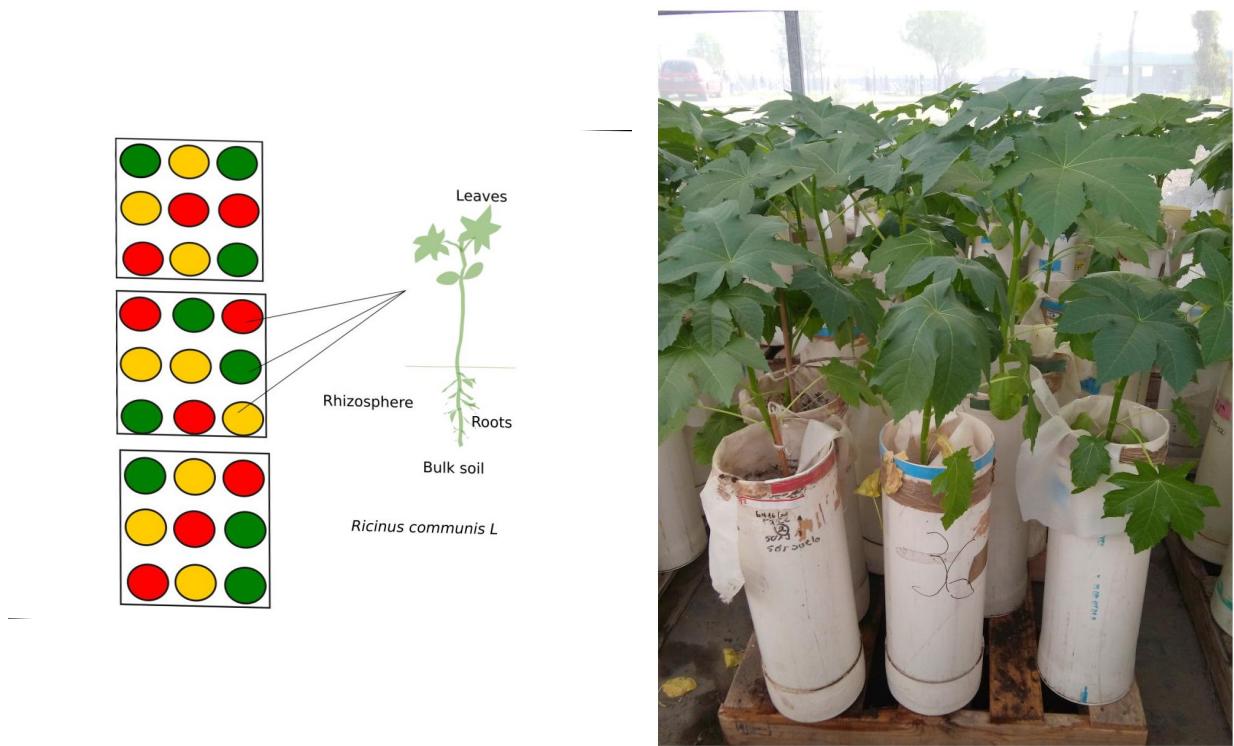
#load libraries and files
library(imager)
library(cowplot)
library(ggpubr)
library(tidyverse)
library(magrittr)
library(ggpubr)
library(readxl)
library(viridis)

ricinus<-load.image(file = "../Figures/ricinus2.jpg")
exper<-load.image(file = "../Figures/exper.jpg")

photo_panel <- ggdraw() + draw_image(ricinus, scale = 1)
photo_panel2 <- ggdraw() + draw_image(exper)

plot_grid(photo_panel, photo_panel2)

```



```

table_exper<- read_excel("../Data/experiment.xls")

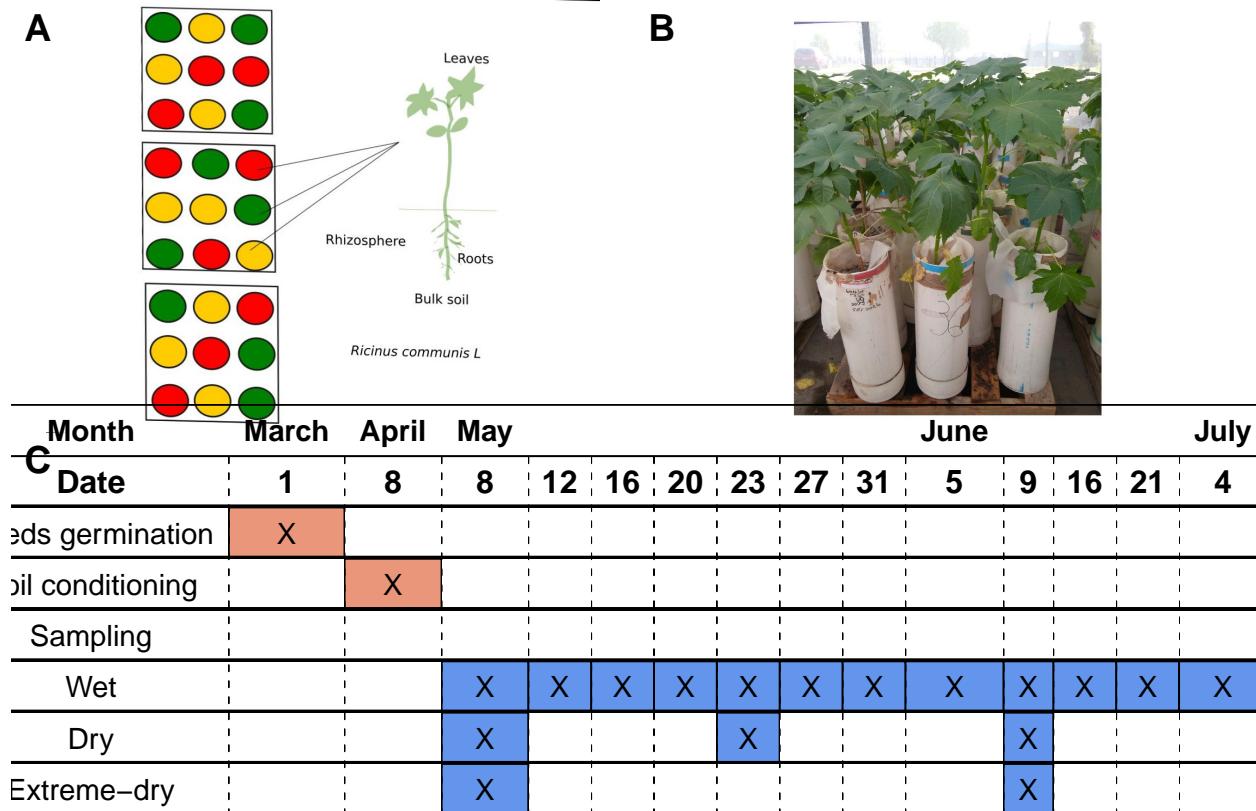
table_exper[is.na(table_exper)] <- ""
names_cols<-c("Month", "March" , "April" , "May" , "", "", "", "", "", "", ,
             "" , "June" , "", "", "", "July" , "")
colnames(table_exper)<-names_cols

table_experiment<-table_exper %>% ggtexttable(rows = NULL, theme = ttheme("blank"))%>%
  tab_add_vline(at.column = 1:16, column.side = "left", from.row = 2, linetype = 2)%>%
  tab_add_hline(at.row = 2:8, row.side = "bottom", linewidth = 3, linetyp
e = 1) %>%
  tbody_add_border() %>%
  thead_add_border()%>%
  table_cell_bg(row =6 , column = 4:15, fill="cornflowerblue")%>%
  table_cell_bg(row =7 , column = 4, fill="cornflowerblue")%>%
  table_cell_bg(row =8 , column = 4, fill="cornflowerblue")%>%
  table_cell_bg(row =7 , column = 8, fill="cornflowerblue")%>%
  table_cell_bg(row =7 , column = 12, fill="cornflowerblue")%>%
  table_cell_bg(row =8 , column = 12, fill="cornflowerblue")%>%
  table_cell_bg(row =3 , column = 2, fill="darksalmon")%>%
  table_cell_bg(row =4 , column = 3, fill="darksalmon")%>%
  table_cell_bg(row =5 , column = 16, fill="darksalmon")%>%
  table_cell_font(row = 2, column = 1:tab_ncol(.), face = "bold")%>%
  tab_add_footnote(
    text = "*cells in blue = Watering, cells in red = Experiment key points",
    size = 10, face = "italic")

p1<- plot_grid(photo_panel, photo_panel2, labels = c("A", "B"))

```

```
p2<- plot_grid(p1, table_experiment, labels = c("", "C"), nrow = 2)
p2
```



*cells in blue = Watering, cells in red = Experiment key points

```
#ggsave('Figures_final/FigS1.design.pdf', width = 10, height = 5, dpi = 300, plot = p2)
```

```
#loading libraries
library(hillR)
library(tidyverse)
library(qiime2R)
library(dplyr)
library(cowplot)
library(RColorBrewer)
library(gttable)
library(ggpubr)
```

```
#setwd("/home/steph/Documents/Documentos/fastas nuevos/18S/R_project")
```

```
MD_q<- read.delim("../Data/diversity.txt")
MD_q0<- read.delim("../Data/MD_q0.txt")
MD_q1<- read.delim("../Data/MD_q1.txt")
MD_q2<- read.delim("../Data/MD_q2.txt")
```

```
MD_q<- read.delim("../Data/diversity.txt")%>%mutate(
  qs= case_when(
    str_detect(q, "q0") ~ "q = 0",
    str_detect(q, "q1") ~ "q = 1",
```

```

str_detect(q, "q2") ~ "q = 2"))

MD_q$Treatment <- factor(MD_q$Treatment, levels = c("Wet", "Dry", "Extreme_dry"))
MD_q0$Treatment <- factor(MD_q0$Treatment, levels = c("Wet", "Dry", "Extreme_dry"))
MD_q1$Treatment <- factor(MD_q1$Treatment, levels = c("Wet", "Dry", "Extreme_dry"))
MD_q2$Treatment <- factor(MD_q2$Treatment, levels = c("Wet", "Dry", "Extreme_dry"))

a0<-MD_q %>% filter(q=="q0" )%>% ggbarplot(., x = "Type", y = "value", facet.by = c(
  "qs", "Treatment"), fill = "Type_of_soil", add = "mean_se") + scale_fill_manual(
  values = c("#800000", "#808000", "#008000", "#D35400", "#2E4053"))+theme(
  legend.position = "none", axis.title.x = element_blank())+ylab("Effective number of ASVs")+
  theme( panel.border = element_blank(),
  panel.spacing.x = unit(0,"line"), axis.line= element_line(colour = "black"),
  strip.text.y = element_text(size=14, face="italic"),
  strip.text.x = element_text(size=14),
  axis.title = element_text(size = 14))
a1<-MD_q %>% filter(q=="q1" )%>% ggbarplot(., x = "Type", y = "value", facet.by = c(
  "qs", "Treatment"), fill = "Type_of_soil", add = "mean_se") + scale_fill_manual(
  values = c("#800000", "#808000", "#008000", "#D35400", "#2E4053"))+theme(
  legend.position = "none", axis.title.x = element_blank())+ylab("Effective number of ASVs")+
  theme( panel.border = element_blank(),
  panel.spacing.x = unit(0,"line"), axis.line= element_line(colour = "black"),
  strip.text.y = element_text(size=14, face="italic"),
  strip.text.x = element_text(size=14),
  axis.title = element_text(size = 14))
a2<-MD_q %>% filter(q=="q2" )%>% ggbarplot(., x = "Type", y = "value", facet.by = c(
  "qs", "Treatment"), fill = "Type_of_soil", add = "mean_se") + scale_fill_manual(
  values = c("#800000", "#808000", "#008000", "#D35400", "#2E4053"))+theme(
  legend.position = "none", axis.title.x = element_blank())+ylab("Effective number of ASVs")+
  theme( panel.border = element_blank(),
  panel.spacing.x = unit(0,"line"), axis.line= element_line(colour = "black"),
  strip.text.y = element_text(size=14, face="italic"),
  strip.text.x = element_text(size=14),
  axis.title = element_text(size = 14))

library(cowplot)
tres<-plot_grid(a0, a1, a2, nrow = 3, labels = c("A", "B", "C"))

MD_q$Treatment <- factor(MD_q$Treatment, levels = c("Wet", "Dry", "Extreme_dry"))

b0<-MD_q0 %>% ggbarplot(., x = "Type", y = "MD_q", facet.by = c(
  "qs", "Treatment"), fill = "Type_of_soil", add = "mean_se") + scale_fill_manual(
  values = c("#800000", "#808000", "#008000", "#D35400", "#2E4053"))+theme(
  legend.position = "none", axis.title.x = element_blank())+ylab("Mean functional diversity")+
  theme( panel.border = element_blank(),
  panel.spacing.x = unit(0,"line"), axis.line= element_line(colour = "black"),
  strip.text.y = element_text(size=14, face="italic"),
  strip.text.x = element_text(size=14),
  axis.title = element_text(size = 14))

b1<-MD_q1 %>% ggbarplot(., x = "Type", y = "MD_q", facet.by = c(
  "qs", "Treatment"), fill = "Type_of_soil", add = "mean_se") + scale_fill_manual(
  values = c("#800000", "#808000", "#008000", "#D35400", "#2E4053"))+theme(

```

```

    legend.position = "none", axis.title.x = element_blank())+ylab("Mean functional diversity")+
  theme( panel.border = element_blank(),
  panel.spacing.x = unit(0,"line"), axis.line= element_line(colour = "black"),
  strip.text.y = element_text(size=14, face="italic"),
  strip.text.x = element_text(size=14),
  axis.title = element_text(size = 14))

b2<-MD_q2 %>% ggbarplot(., x = "Type", y = "MD_q", facet.by = c(
  "qs", "Treatment"),fill = "Type_of_soil", add = "mean_se") + scale_fill_manual(
  values = c("#800000", "#808000", "#008000", "#D35400", "#2E4053"))+theme(
  legend.position = "none", axis.title.x = element_blank())+ylab("Mean functional diversity")+
  theme( panel.border = element_blank(),
  panel.spacing.x = unit(0,"line"), axis.line= element_line(colour = "black"),
  strip.text.y = element_text(size=14, face="italic"),
  strip.text.x = element_text(size=14),
  axis.title = element_text(size = 14))

tres_b<- plot_grid(b0, b1, b2, nrow = 3, labels = c("D", "E", "F"))
all<- plot_grid(tres, tres_b, ncol = 2)

all
```

A

Condition	Roots	Rhizosphere	Bulk soil
Wet	~12	~18	~10
Dry	~8	~15	~3
Extreme_dry	~18	~25	~5

D

Condition	Roots	Rhizosphere	Bulk soil
Wet	~350	~550	~400
Dry	~250	~450	~100
Extreme_dry	~600	~1000	~200

B

Condition	Roots	Rhizosphere	Bulk soil
Wet	~4.0	~2.5	~5.0
Dry	~3.0	~2.0	~1.5
Extreme_dry	~8.0	~6.0	~2.0

E

Condition	Roots	Rhizosphere	Bulk soil
Wet	~120	~150	~200
Dry	~80	~130	~50
Extreme_dry	~380	~350	~100

C

Condition	Roots	Rhizosphere	Bulk soil
Wet	~3.0	~1.5	~3.5
Dry	~2.0	~1.0	~1.5
Extreme_dry	~5.5	~4.0	~1.0

F

Condition	Roots	Rhizosphere	Bulk soil
Wet	~80	~120	~180
Dry	~70	~110	~50
Extreme_dry	~250	~220	~80

```

#ggsave('/home/steph/Documents/Documentos/fastas nuevos/18S/R_project/Figures_final//Figure1.diversity_%
#width = 18, height = 10, dpi = 300, plot = all)
```

Lmer models with interaction effects -Taxonomic alpha diversity

```
library(lme4)
```

```

## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyverse':
##
```

```

## expand, pack, unpack
library(nlme)

## Attaching package: 'nlme'
## The following object is masked from 'package:lme4':
## lmList

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

library(cowplot)
library(pgirmess)

## Registered S3 method overwritten by 'spdep':
## method from
## plot.mst ape
library(emmeans)

alpha<-read.delim("../Data/alpha_all_filt.tsv") %>% dplyr::select(sampleid, q, value)
alpha<- alpha %>% spread(q, value)
metadata<- read.delim("../Data/FINALMAP18S_plant.csv", check.names = F)
alpha<- alpha %>% inner_join(metadata,by = c("sampleid"="#SampleID"))

alpha<- alpha %>% mutate(Treatments =case_when(
Treatment == "1" ~ "Wet",
Treatment == "2" ~ "Dry",
Treatment == "3" ~ "Extreme-dry"))

#q0
q0_lme<-lme(q0~ Type_of_soil+Treatments+Type_of_soil*Treatments, random=~1 |Plant, data = alpha)
q0_lme_perm<-PermTest(q0_lme)
q0_lme_perm

## Monte-Carlo test
## Call:
## PermTest.lme(obj = q0_lme)
##
## Based on 1000 replicates
## Simulated p-value:
##                               p.value
## (Intercept)                  0.999
## Type_of_soil                  0.020
## Treatments                     0.178
## Type_of_soil:Treatments      0.084

#q1
q1_lme<-lme(q1~ Type_of_soil+Treatments+Type_of_soil*Treatments, random=~1 |Plant, data = alpha)
q1_lme_perm<-PermTest(q1_lme)

```

```

q1_lme_perm

##
## Monte-Carlo test
##
## Call:
## PermTest.lme(obj = q1_lme)
##
## Based on 1000 replicates
## Simulated p-value:
##                               p.value
## (Intercept)                  0.979
## Type_of_soil                  0.640
## Treatments                     0.163
## Type_of_soil:Treatments      0.013

#q2
q2_lme<-lme(q2~ Type_of_soil+Treatments+Type_of_soil*Treatments, random=~1 |Plant, data = alpha)
q2_lme_perm<-PermTest(q2_lme)
q2_lme_perm

##
## Monte-Carlo test
##
## Call:
## PermTest.lme(obj = q2_lme)
##
## Based on 1000 replicates
## Simulated p-value:
##                               p.value
## (Intercept)                  0.947
## Type_of_soil                  0.667
## Treatments                     0.188
## Type_of_soil:Treatments      0.014

Lmer models with interaction effects -Functional alpha diversity

#Functional
MD_q0<- read.delim("../Data/MD_q0.txt") %>% mutate(cate="q0")
MD_q1<- read.delim("../Data/MD_q1.txt")%>% mutate(cate="q1")
MD_q2<- read.delim("../Data/MD_q2.txt")%>% mutate(cate="q2")

alpha_f<-rbind(MD_q0, MD_q1, MD_q2) %>% dplyr::select(sampleid, cate, MD_q)
alpha_f<- alpha_f %>% spread(cate, MD_q)
alpha_f<- alpha_f %>% inner_join(metadata, by = c("sampleid"="#SampleID")) %>% mutate(
  Treatments =case_when(
    Treatment == "1" ~ "Wet",
    Treatment == "2" ~ "Dry",
    Treatment == "3" ~ "Extreme-dry"))

#q0
q0_lme_f<-lme(q0~ Type_of_soil+Treatment+Type_of_soil*Treatment, random=~1 |Plant, data = alpha)
q0_lme_perm_f<-PermTest(q0_lme_f)
q0_lme_perm_f

```

```

##  

## Monte-Carlo test  

##  

## Call:  

## PermTest.lme(obj = q0_lme_f)  

##  

## Based on 1000 replicates  

## Simulated p-value:  

##  

##          p.value  

## (Intercept)      0.336  

## Type_of_soil     0.529  

## Treatment        0.185  

## Type_of_soil:Treatment 0.014  

#q1  

q1_lme_f<-lme(q1~ Type_of_soil+Treatment+Type_of_soil*Treatment, random=~1 |Plant, data = alpha)  

q1_lme_perm_f<-PermTest(q1_lme_f)  

q1_lme_perm_f  

##  

## Monte-Carlo test  

##  

## Call:  

## PermTest.lme(obj = q1_lme_f)  

##  

## Based on 1000 replicates  

## Simulated p-value:  

##  

##          p.value  

## (Intercept)      0.150  

## Type_of_soil     0.070  

## Treatment        0.159  

## Type_of_soil:Treatment 0.003  

#q2  

q2_lme_f<-lme(q2~ Type_of_soil+Treatment+Type_of_soil*Treatment, random=~1 |Plant, data = alpha)  

q2_lme_perm_f<-PermTest(q2_lme_f)  

q2_lme_perm_f  

##  

## Monte-Carlo test  

##  

## Call:  

## PermTest.lme(obj = q2_lme_f)  

##  

## Based on 1000 replicates  

## Simulated p-value:  

##  

##          p.value  

## (Intercept)      0.157  

## Type_of_soil     0.080  

## Treatment        0.134  

## Type_of_soil:Treatment 0.005

```

Sorting by letters

```

#Letters were added manually resulting from the next code

alpha<- alpha %>% unite("interact", c("Type_of_soil", "Treatments"), remove = F)

q0_lme<-lme(q0~ interact, random=~1 |Plant, data = alpha)
q0_lme_means<-emmeans(q0_lme, pairwise ~ interact)
multcomp::cld(object = q0_lme_means$emmeans,
              Letters = letters)

##   interact          emmean    SE df lower.CL upper.CL .group
## Non-rizospheric_Dry     2.89 2.10 26   -1.436    7.21    a
## Non-rizospheric_Extreme-dry 4.21 2.23 26   -0.373    8.78    a
## Roots_Dry                8.64 2.22 26    4.067   13.21   ab
## Non-rizospheric_Wet      8.76 2.37 26    3.876   13.64   abc
## Roots_Wet                 9.87 2.79 26    4.124   15.61   abc
## Rizospheric_Dry         11.78 2.55 26    6.532   17.03   abc
## Rizospheric_Wet         13.03 2.37 26    8.147   17.91   abc
## Roots_Extreme-dry       16.47 2.37 26   11.594   21.35   bc
## Rizospheric_Extreme-dry 19.46 2.22 26   14.885   24.03    c
##
## Degrees-of-freedom method: containment
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 9 estimates
## significance level used: alpha = 0.05
## NOTE: Compact letter displays can be misleading
##        because they show NON-findings rather than findings.
##        Consider using 'pairs()', 'pwpp()', or 'pwpm()' instead.

q1_lme<-lme(q1~ interact, random=~1 |Plant, data = alpha)
q1_lme_means<-emmeans(q1_lme, pairwise ~ interact)
multcomp::cld(object = q1_lme_means$emmeans,
              Letters = letters)

##   interact          emmean    SE df lower.CL upper.CL .group
## Non-rizospheric_Dry     1.44 0.932 26   -0.473    3.36    a
## Non-rizospheric_Extreme-dry 1.64 0.988 26   -0.394    3.67    a
## Rizospheric_Dry         1.83 1.138 26   -0.514    4.16    a
## Rizospheric_Wet         2.11 1.055 26   -0.061    4.28    a
## Roots_Dry                 2.67 0.987 26    0.644    4.70    a
## Roots_Wet                  3.55 1.246 26    0.987    6.11   ab
## Non-rizospheric_Wet      3.94 1.055 26    1.771    6.11   ab
## Rizospheric_Extreme-dry  4.67 0.987 26    2.641    6.70   ab
## Roots_Extreme-dry        8.26 1.055 26    6.093   10.43    b
##
## Degrees-of-freedom method: containment
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 9 estimates
## significance level used: alpha = 0.05
## NOTE: Compact letter displays can be misleading
##        because they show NON-findings rather than findings.
##        Consider using 'pairs()', 'pwpp()', or 'pwpm()' instead.

q2_lme<-lme(q2~ interact, random=~1 |Plant, data = alpha)
q2_lme_means<-emmeans(q2_lme, pairwise ~ interact)
multcomp::cld(object = q2_lme_means$emmeans,

```

```

Letters = letters)

##   interact           emmean    SE df lower.CL upper.CL .group
## Non-rizospheric_Dry      1.25 0.650 26  -0.0889     2.58  a
## Non-rizospheric_Extreme-dry 1.27 0.689 26  -0.1438     2.69  a
## Rizospheric_Dry          1.33 0.794 26  -0.3004     2.96  a
## Rizospheric_Wet          1.67 0.736 26   0.1600     3.19  a
## Roots_Dry                 2.02 0.689 26   0.6022     3.43  a
## Roots_Wet                 2.71 0.869 26   0.9231     4.50  ab
## Non-rizospheric_Wet       2.86 0.736 26   1.3468     4.37  ab
## Rizospheric_Extreme-dry   3.12 0.689 26   1.6993     4.53  ab
## Roots_Extreme-dry         5.77 0.736 26   4.2603     7.29  b
##
## Degrees-of-freedom method: containment
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 9 estimates
## significance level used: alpha = 0.05
## NOTE: Compact letter displays can be misleading
##        because they show NON-findings rather than findings.
##        Consider using 'pairs()', 'pwpp()', or 'pwpm()' instead.

alpha_f<- alpha_f %>% unite("interact", c("Type_of_soil", "Treatments"), remove = F)

q0_lme_f<-lme(q0~ interact, random=~1 |Plant, data = alpha_f)
q0_lme_means_f<-emmeans(q0_lme_f, pairwise ~ interact)
multcomp::cld(object = q0_lme_means_f$emmeans,
               Letters = letters)

##   interact           emmean    SE df lower.CL upper.CL .group
## Non-rizospheric_Dry      60.9 130 26  -207.0     329  a
## Non-rizospheric_Extreme-dry 164.3 138 26  -119.4     448  a
## Roots_Dry                 209.0 138 26   -74.4     492  a
## Roots_Wet                 218.1 173 26  -137.8     574  a
## Non-rizospheric_Wet       374.0 147 26    71.6     676  ab
## Rizospheric_Dry          405.2 158 26    80.0     730  ab
## Rizospheric_Wet          487.2 147 26   184.8     790  ab
## Roots_Extreme-dry         657.7 147 26   355.3     960  ab
## Rizospheric_Extreme-dry   1009.1 138 26   725.8    1292  b
##
## Degrees-of-freedom method: containment
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 9 estimates
## significance level used: alpha = 0.05
## NOTE: Compact letter displays can be misleading
##        because they show NON-findings rather than findings.
##        Consider using 'pairs()', 'pwpp()', or 'pwpm()' instead.

q1_lme_f<-lme(q1~ interact, random=~1 |Plant, data = alpha_f)
q1_lme_means_f<-emmeans(q1_lme_f, pairwise ~ interact)
multcomp::cld(object = q1_lme_means_f$emmeans,
               Letters = letters)

##   interact           emmean    SE df lower.CL upper.CL .group
## Non-rizospheric_Dry      43.8 58.4 26   -76.3     164  a
## Roots_Dry                 91.3 61.8 26   -35.6     218  ab

```

```

## Roots_Wet           93.4 77.6 26   -66.1    253  ab
## Non-rizospheric_Extreme-dry 96.8 61.9 26   -30.4    224  ab
## Rizospheric_Dry      142.8 70.9 26    -2.9    289  ab
## Rizospheric_Wet      148.0 65.9 26    12.5    284  ab
## Non-rizospheric_Wet 201.2 65.9 26    65.6    337  ab
## Rizospheric_Extreme-dry 345.8 61.8 26   218.8    473  b
## Roots_Extreme-dry    361.2 65.9 26   225.7    497  b
##
## Degrees-of-freedom method: containment
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 9 estimates
## significance level used: alpha = 0.05
## NOTE: Compact letter displays can be misleading
##       because they show NON-findings rather than findings.
##       Consider using 'pairs()', 'pwpp()', or 'pwpm()' instead.
q2_lme_f<-lme(q2~ interact, random=~1 |Plant, data = alpha_f)
q2_lme_means_f<-emmeans(q2_lme_f, pairwise ~ interact)
multcomp::cld(object = q2_lme_means_f$emmeans,
               Letters = letters)

## interact              emmean    SE df lower.CL upper.CL .group
## Non-rizospheric_Dry  40.0 40.8 26   -43.88    124  a
## Roots_Wet             70.5 54.0 26   -40.57    182  ab
## Roots_Dry              72.3 43.1 26   -16.28    161  ab
## Non-rizospheric_Extreme-dry 83.4 43.2 26    -5.32    172  ab
## Rizospheric_Dry       113.3 49.4 26   11.74    215  ab
## Rizospheric_Wet       116.8 46.0 26   22.28    211  ab
## Non-rizospheric_Wet  158.2 46.0 26   63.65    253  ab
## Rizospheric_Extreme-dry 245.9 43.1 26   157.24    334  b
## Roots_Extreme-dry     263.0 46.0 26   168.46    357  b
##
## Degrees-of-freedom method: containment
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 9 estimates
## significance level used: alpha = 0.05
## NOTE: Compact letter displays can be misleading
##       because they show NON-findings rather than findings.
##       Consider using 'pairs()', 'pwpp()', or 'pwpm()' instead.

```

Loading files

```

library(tidyverse)
intra_ro_wet_q0<- read.csv(
  "../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=0-table_wet_ro.txt.csv") %>% mutate(
  qs="q0") %>% mutate(type="wet")
intra_ro_wet_q1<- read.csv(
  "../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=1-table_wet_ro.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="wet")
intra_ro_wet_q2<- read.csv(
  "../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=2-table_wet_ro.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="wet")

intra_ri_wet_q0<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=0-table_wet_ri.txt.csv") %>% mutate(
  qs="q0")%>% mutate(type="wet")

```

```

intra_ri_wet_q1<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=1-table_wet_ri.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="wet")
intra_ri_wet_q2<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=2-table_wet_ri.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="wet")

intra_nr_wet_q0<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=0-table_wet_nr.txt.csv") %>% mutate(
  qs="q0")%>% mutate(type="wet")
intra_nr_wet_q1<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=1-table_wet_nr.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="wet")
intra_nr_wet_q2<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=2-table_wet_nr.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="wet")

intra_ro_dry_q0<- read.csv(
  "../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=0-table_dry_ro.txt.csv") %>% mutate(
  qs="q0")%>% mutate(type="dry")
intra_ro_dry_q1<- read.csv(
  "../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=1-table_dry_ro.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="dry")
intra_ro_dry_q2<- read.csv(
  "../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=2-table_dry_ro.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="dry")

intra_ri_dry_q0<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=0-table_dry_ri.txt.csv") %>% mutate(
  qs="q0")%>% mutate(type="dry")
intra_ri_dry_q1<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=1-table_dry_ri.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="dry")
intra_ri_dry_q2<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=2-table_dry_ri.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="dry")

intra_nr_dry_q0<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=0-table_dry_nr.txt.csv") %>% mutate(
  qs="q0")%>% mutate(type="dry")
intra_nr_dry_q1<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=1-table_dry_nr.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="dry")
intra_nr_dry_q2<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=2-table_dry_nr.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="dry")

intra_ro_exdry_q0<- read.csv(
  "../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=0-table_exdry_ro.txt.csv") %>% mutate(
  qs="q0")%>% mutate(type="extreme-dry")
intra_ro_exdry_q1<- read.csv(

```

```

"../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=1-table_exdry_ro.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="extreme-dry")
intra_ro_exdry_q2<- read.csv(
  "../Data/beta/Beta_diversityro/Intra_Beta_Similarity-q=2-table_exdry_ro.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="extreme-dry")

intra_ri_exdry_q0<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=0-table_exdry_ri.txt.csv") %>% mutate(
  qs="q0")%>% mutate(type="extreme-dry")
intra_ri_exdry_q1<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=1-table_exdry_ri.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="extreme-dry")
intra_ri_exdry_q2<- read.csv(
  "../Data/beta/Beta_diversityri/Intra_Beta_Similarity-q=2-table_exdry_ri.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="extreme-dry")

intra_nr_exdry_q0<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=0-table_exdry_nr.txt.csv") %>% mutate(
  qs="q0")%>% mutate(type="extreme-dry")
intra_nr_exdry_q1<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=1-table_exdry_nr.txt.csv") %>% mutate(
  qs="q1")%>% mutate(type="extreme-dry")
intra_nr_exdry_q2<- read.csv(
  "../Data/beta/Beta_diversitynr/Intra_Beta_Similarity-q=2-table_exdry_nr.txt.csv") %>% mutate(
  qs="q2")%>% mutate(type="extreme-dry")

```

Plot

```

library(ggpubr)
my_comparisons <- list( c("wet", "dry"), c("wet", "extreme-dry"), c("dry", "extreme-dry") )

intra_ro<- rbind(intra_ro_wet_q0, intra_ro_dry_q0, intra_ro_exdry_q0,
                  intra_ro_wet_q1, intra_ro_dry_q1, intra_ro_exdry_q1,
                  intra_ro_wet_q2, intra_ro_dry_q2, intra_ro_exdry_q2)

i1<-intra_ro %>% mutate(TurnOver= 1-TurnoverComp) %>% ggpubr::ggboxplot(
  x = "type", y="TurnOver", fill = "type")+
  facet_wrap(~qs, scales = "free") + xlab("") +
  ylab("Ratio of ASVs turnover-Roots") +
  scale_fill_manual(values = c("DarkGreen", "yellow", "red")) +
  stat_compare_means(comparisons = my_comparisons, label = "p.signif") +
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        panel.border = element_rect(color = "black"),
        panel.spacing.x = unit(0,"line"),
        axis.line= element_line(colour = "black"),
        axis.line.y.right = element_line(colour = "black" ),
        axis.text = element_text(colour = "black", size = 10))

intra_ri<- rbind(intra_ri_wet_q0, intra_ri_dry_q0, intra_ri_exdry_q0,
                  intra_ri_wet_q1, intra_ri_dry_q1, intra_ri_exdry_q1,

```

```

    intra_ri_wet_q2, intra_ri_dry_q2, intra_ri_exdry_q2)

i2<-intra_ri %>%mutate(TurnOver= 1-TurnoverComp)%>% ggpubr::ggboxplot(
  x = "type", y="TurnOver", fill = "type")+
  facet_wrap(~qs, scales = "free")+
  xlab("")+ylab("Ratio of ASVs turnover-Rhizosphere")+
  scale_fill_manual(values = c("DarkGreen", "yellow", "red"))+
  stat_compare_means(comparisons = my_comparisons, label = "p.signif")+
  theme_bw() + theme(panel.grid.major = element_blank(),
                     panel.grid.minor = element_blank(),
                     panel.background = element_blank(),
                     panel.border = element_rect(color = "black"),
                     panel.spacing.x = unit(0,"line"),
                     axis.line= element_line(colour = "black"),
                     axis.line.y.right = element_line(colour = "black" ),
                     axis.text = element_text(colour = "black", size = 10))

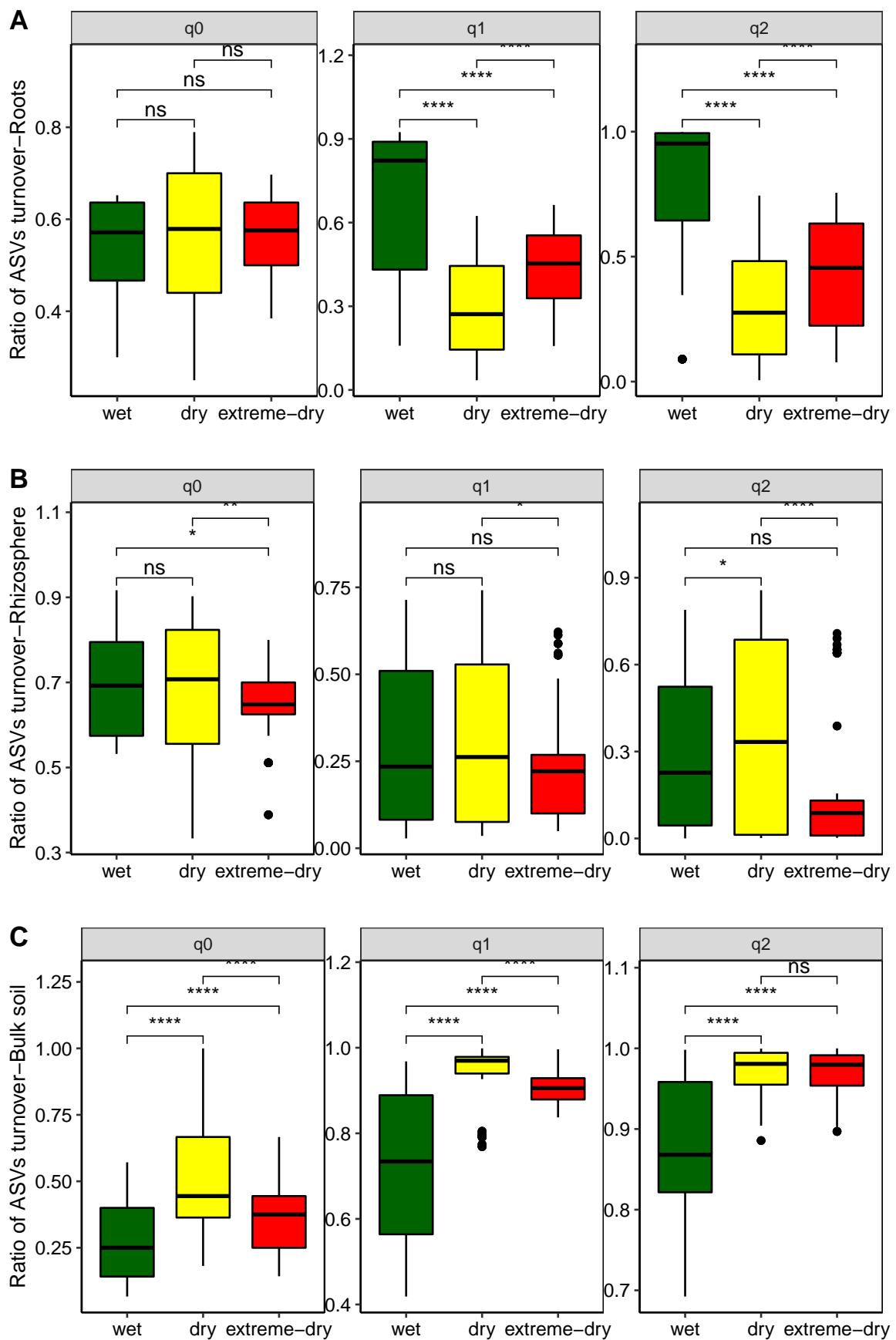
intra_nr<- rbind(intra_nr_wet_q0, intra_nr_dry_q0, intra_nr_exdry_q0,
                   intra_nr_wet_q1, intra_nr_dry_q1, intra_nr_exdry_q1,
                   intra_nr_wet_q2, intra_nr_dry_q2, intra_nr_exdry_q2)

i3<-intra_nr%>%mutate(TurnOver= 1-TurnoverComp) %>% ggpubr::ggboxplot(
  x = "type", y="TurnoverComp", fill = "type")+
  facet_wrap(~qs, scales = "free") + xlab("") +
  ylab("Ratio of ASVs turnover-Bulk soil")+
  scale_fill_manual(values = c("DarkGreen", "yellow", "red"))+
  stat_compare_means(comparisons = my_comparisons, label = "p.signif")+
  theme_bw() + theme( panel.grid.major = element_blank(),
                     panel.grid.minor = element_blank(),
                     panel.background = element_blank(),
                     panel.border = element_rect(color = "black"),
                     panel.spacing.x = unit(0,"line"),
                     axis.line= element_line(colour = "black"),
                     axis.line.y.right = element_line(colour = "black" ),
                     axis.text = element_text(colour = "black", size = 10))

library(cowplot)
p<-plot_grid(
  i1+theme(legend.position = "none"),i2+theme(legend.position = "none"),
  i3+theme(legend.position = "none"), nrow = 3, labels = c("A", "B", "C"))

p

```



```

#ggsave(plot=p, "Figures_final/Fig2.intra-turnover-paired.pdf", width = 10, height = 13)

#loading libraries
library(qiime2R)
library(tidyverse)
library(cowplot)
library(FactoMineR)
library(vegan)
library(ggpubr)
library(zCompositions)
library(compositions)

PCA plot all

#load file and correct zero values
table_240<- read_qza("../Data/merge_table_240_noplant_filtered_nous.qza")$data %>%
  as.data.frame()

taxonomy_240<- read_qza("../Data/taxonomy_blast_240_0.97.qza")$data

#remove non-fungi taxa

taxonomy_filter<- taxonomy_240 %>% filter(
  !str_detect(Taxon, "ozoa"))%>% filter(
  !str_detect(Taxon, "helida")) %>% filter(
  !str_detect(Taxon, "ophyta")) %>% filter(
  !str_detect(Taxon, "Ciliophora")) %>% filter(
  !str_detect(Taxon, "Nucleariidae_and_Fonticula_group")) %>% filter(
  !str_detect(Taxon, "Arthriocida")) %>% filter(
  !str_detect(Taxon, "Labyrinthulomycetes")) %>% filter(
  !str_detect(Taxon, "Apicomplexa")) %>% filter(
  !str_detect(Taxon, "Bicosoecida")) %>% filter(
  !str_detect(Taxon, "Breviatea")) %>% filter(
  !str_detect(Taxon, "Aphelidea")) %>% filter(
  !str_detect(Taxon, "Arthropoda"))

table_filter<- table_240[match(
  taxonomy_filter$Feature.ID, rownames(table_240)),] %>% drop_na()

d.pro.0<- table_filter %>% dplyr::select_at(vars(!contains("US")))
d.pro <- t(cmultRepl(t(d.pro.0), method="CZM", output="p-counts"))

## No. corrected values: 335

# make our compositional and run pca function
d.clr.abund <- t(apply(d.pro, 2, function(x){log(x) - mean(log(x))}))
pcx.abund <- prcomp(d.clr.abund)

#load metadata and taxonomy file
meta<-read_tsv(
  file = "/home/steph/Documents/Documentos/fastas nuevos/18S/MAPPINGS/FINALMAP18S") %>%
  rename(SampleID=`#SampleID`) %>%
  filter(SampleID!="q2:types")

```

```

meta$Compartment<- factor(meta>Type_of_soil,
                           levels = c( "Non-rizospheric", "Rizospheric",
                                      "Roots", "Uncultivated"),
                           labels = c("Bulk soil", "Rhizosphere",
                                      "Roots", "Uncultivated"))
meta$Water_regime<- factor(meta$Treatment,
                            levels = c( "0", "1", "2", "3"),
                            labels = c("Initial","Wet", "Dry", "Extreme dry"))

tax<-read_qza("../Data/taxonomy_blast_240_0.97.qza")$data %>%
  rename(FeatureID=Feature.ID)

#Constructing PCA

#LABELS
PC1 <- paste("PC1", round(sum(pcx.abund$sdev[1] ^ 2) /
                           mvar(d.clr.abund) * 100, 2), "%")
PC2 <- paste("PC2", round(sum(pcx.abund$sdev[2] ^ 2) /
                           mvar(d.clr.abund) * 100, 2), "%")

#SHAPES AND COLOR OF ARROWS

vars_chosen<- c("588a0e16a32970569d10c88aaa93f050",
                "81979356618c524328e9a9fc41c30906",
                "ab657f069eebbddfa32f5fd2609e4b24",
                "0147e304b1ce1be9c4ff15b660605ef2",
                "1db96e3e66ec1535d586d1a3a954cb66")

vars_choosing<- data.frame(pcx.abund$rotation)%>%  rownames_to_column(
  var = "FeatureID")%>%
  filter(FeatureID %in% vars_chosen) %>%  mutate(a=sqrt(PC1^2+PC2^2)) %>%
  mutate(PC1=PC1*40, PC2=PC2*40) %>% left_join(tax)%>% dplyr::select(
    Taxon, PC1, PC2, FeatureID)%>%mutate(tax= str_extract(Taxon, "[^_]+$")) %>%
  mutate_at(c("tax"), funs(tax = case_when(
    tax=="herbarum" ~ "Cladosporium herbarum",
    TRUE~as.character(tax)))))

color_type<- c("#800000", "#808000", "#008000", "#D35400", "#2E4053")

pca<- ggplot() +
  theme_bw()+
  xlab(PC1) +
  ylab(PC2) +
  geom_segment(data = vars_choosing, aes(x = 0, y = 0, xend = PC1, yend = PC2),
               arrow=arrow(length=unit(0.15,"cm")),
               alpha = 0.75, color = 'black', size= 0.6)+
  theme(axis.text = element_text(colour = "black", size = 14),
        axis.title = element_text(colour = "black", size = 14),
        legend.text = element_text(size = 9),
        axis.ticks = element_line(colour = "black"),
        plot.title = element_text(size = 16))

```

```

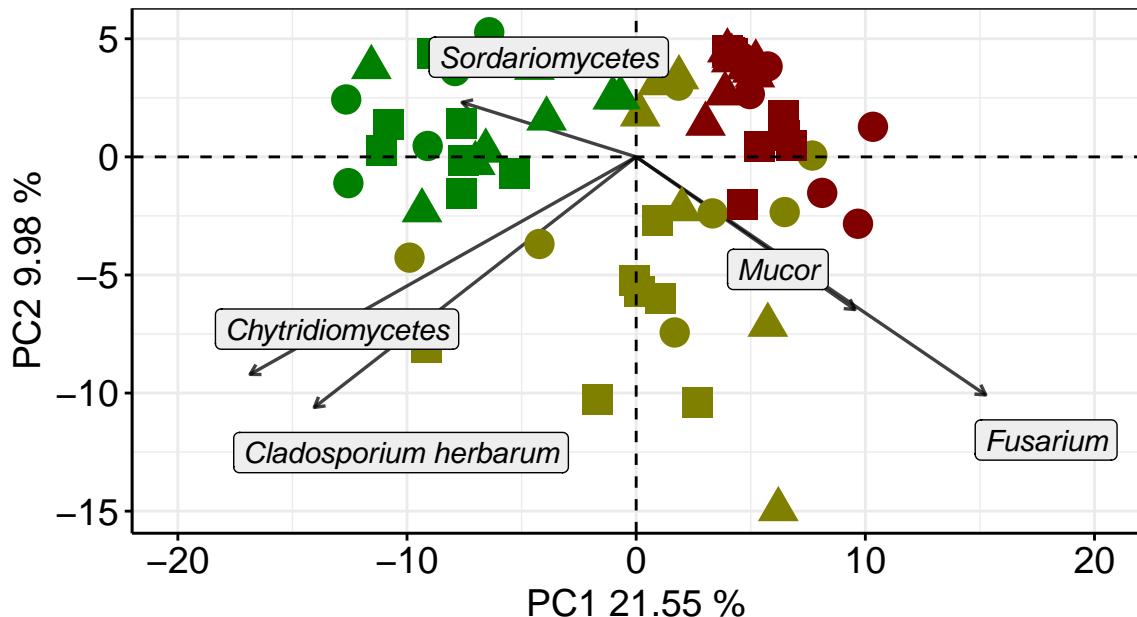
axis.line = element_line(colour = "black"),
axis.text.y.right = element_text(colour = "black"),
axis.text.x.top = element_text(colour = "black"),
legend.position = "top",
legend.box = "vertical",
legend.direction = "horizontal",
legend.spacing.y = unit(0.01, 'cm'),
legend.title = element_blank()) +
geom_point(
  data=data.frame(pcx.abund$x) %>% rownames_to_column(var = "SampleID")%>%
    left_join(meta, by = "SampleID") %>% filter(!Compartiment == "NA"),
  aes(x=PC1, y=PC2, color=Compartiment, shape=Water_regime),
  size=5) + geom_vline(xintercept = 0, linetype = 2) +
  geom_hline(yintercept = 0, linetype = 2) +
  scale_color_manual(values = color_type) +
  scale_x_continuous(limits = c(-20,20)) +
  ggrepel::geom_label_repel(data = vars_choosing, aes(x=PC1, y=PC2, label= tax),
  segment.colour = NA, col = 'black', fill= "#EEEEEE",
  fontface="italic", box.padding = 0.6, size=4)+ theme(
  plot.margin = unit(c(0.5, 1.5, 0.4, 0.1), "cm"))+
  guides(colour = guide_legend(override.aes = list(size=3)),
  shape = guide_legend(override.aes = list(size = 3)))+
  theme(legend.text = element_text(size = 12))

```

pca

● Wet ▲ Dry ■ Extreme dry

● Bulk soil ● Rhizosphere ● Roots



PERMANOVAs

```

d.clr.abund2<- data.frame(d.clr.abund, check.names = F) %>% rownames_to_column(
  var = "ids") %>% filter(!str_detect(ids, 'US')) %>% column_to_rownames(
  var = "ids")

meta_just<- data.frame(d.clr.abund2, check.names = F) %>% rownames_to_column(
  var = "SampleID") %>% inner_join(meta)

perm<- how(nperm = 999)
setBlocks(perm)<- with(meta_just, Plot)
ad_grouped<-adonis2(d.clr.abund2~Water_regime*Compartment,
                      data = meta_just, method = "euclidian",
                      permutations =999) %>% round(
  ., digits = 3) %>%replace(is.na(.), "-")

Permanova_grouped<-data.frame(ad_grouped, check.names = F) %>% rownames_to_column(
  var="Factor") %% ggttexttable(., rows = NULL, theme = ttheme("blank")) %>%
  tab_add_hline(at.row = 1:2, row.side = "top", linewidth = 2)%>%
  table_cell_font(., row = 3, column = 6, face = "bold") %>%
  table_cell_font(., row = 2, column = 6, face = "bold") %>%
  tab_add_hline(at.row = c(6), row.side = "bottom",
                linewidth = 3, linetype = 1) %>%
  tab_add_footnote(
    text = "*p values in Bold are significant using \n an alpha value of 0.05",
    size = 10, face = "italic")

```

Permanova_grouped

Factor	Df	SumOfSqs	R2	F	Pr(>F)
Water_regime	2	463.088	0.039	1.583	0.023
Compartment	2	2594.214	0.216	8.867	0.001
Water_regime:Compartment	4	733.834	0.061	1.254	0.088
Residual	56	8191.646	0.684	-	-
Total	64	11982.782	1.000	-	-

*p values in Bold are significant using
an alpha value of 0.05

```

#ggsave('..../Figures_final/paired_perma.pdf',
#  width = 6, height = 2, dpi = 300, plot =Pairwsie_permanova_grouped)
#ggsave('..../Figures_final/pca_all.pdf',
#  width = 6, height = 2, dpi = 300, plot =pca)

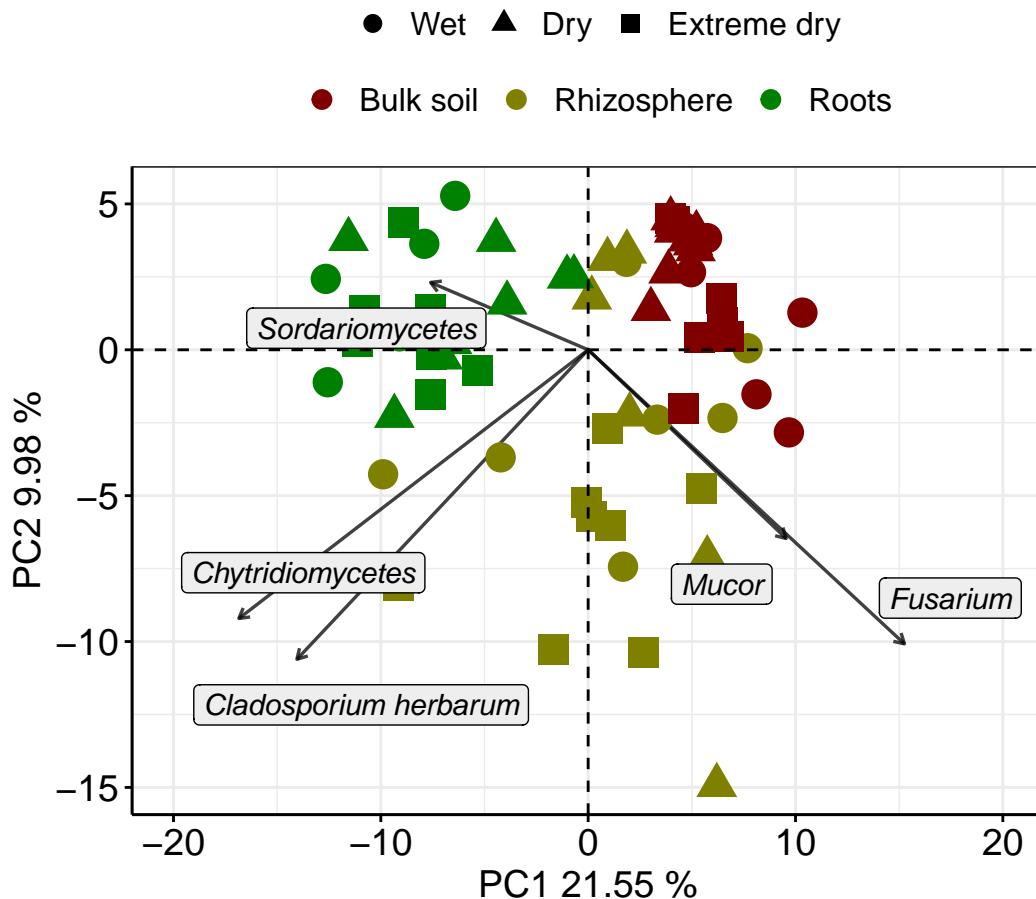
```

```

b<-cowplot:::plot_grid(pca,Permanova_grouped,
  nrow=2, rel_heights = c(1.8,1),
  labels = c("A", "B"), label_size = 18)

```

b

A**B**

Factor	Df	SumOfSqs	R2	F	Pr(>F)
Water_regime	2	463.088	0.039	1.583	0.023
Compartment	2	2594.214	0.216	8.867	0.001
Water_regime:Compartment	4	733.834	0.061	1.254	0.088
Residual	56	8191.646	0.684	-	-
Total	64	11982.782	1.000	-	-

*p values in Bold are significant using
an alpha value of 0.05

```
#ggsave('.../.../Figures_final/beta.png',
#       width = 14, height = 9.5, dpi = 300, plot = c)
```

Water regime PCA AND PERMANOVA

```
#barplots
```

```
table_all<- data.frame(read_qza(
  ".../Data/merge_table_240_noplant_filtered_nous.qza")$data ,
```

```

    check.names = F)

taxonomy<- read_qza("../Data//taxonomy_blast_240_0.97.qza")$data
metadata<- read_tsv("../Data/FINALMAP18S") %>% dplyr::select(
  -Month)

metadata$Compartment<- factor(
  metadata>Type_of_soil,
  levels = c( "non-rizospheric", "Non-rizospheric", "Rizospheric",
             "Roots", "Seeds", "Uncultivated"),
  labels = c("Initials", "Non-rhizospheric" , "Rhizosphere",
            "Roots", "Initials", "Uncultivated"))

metadata$Water_regime<- factor(metadata$Treatment,
                                levels = c( "0", "1", "2", "3"),
                                labels = c("Initials", "Wet" , "Dry", "Extreme-dry"))

#ALL
table_all_ro<- table_filter %>% dplyr::select_at(vars(
  matches("RO")))) %% t()%>% as.data.frame( )%>% rownames_to_column(
  var = "#SampleID") %>% inner_join(metadata) %>% group_by(
  Water_regime) %>% summarise_at(c(2:152), mean) %>% column_to_rownames(
  var = "Water_regime") %>% t()

table_all_ri<- table_filter %>% dplyr::select_at(vars(
  matches("RI")))) %% t()%>% as.data.frame(
) %>% rownames_to_column(
  var = "#SampleID") %>% inner_join(metadata) %>% group_by(
  Water_regime) %>% summarise_at(c(2:152), mean) %>% column_to_rownames(
  var = "Water_regime") %>% t()

table_all_nr<- table_filter%>% dplyr::select_at(vars(
  matches("NR")))) %% t()%>% as.data.frame(
) %>% rownames_to_column(
  var = "#SampleID") %>% inner_join(metadata) %>% group_by(
  Water_regime) %% summarise_at(c(2:152), mean) %>% column_to_rownames(
  var = "Water_regime") %>% t()

#PCA 'S
all_ro<- table_filter %>% dplyr::select_at(vars(
  matches("RO")))) %% t()%>% as.data.frame(
) %>% rownames_to_column(
  var = "#SampleID") %% inner_join(metadata) %>% arrange(
  Water_regime) %>% filter(
  !Water_regime=="TO") %>% dplyr::select(
  -BarcodeSequence:-Water_regime) %>% column_to_rownames(
  var = "#SampleID") %>% t()

all_ri<- table_filter %>% dplyr::select_at(vars(
  matches("RI")))) %% t()%>% as.data.frame(
) %>% rownames_to_column(

```

```

var = "#SampleID") %>% inner_join(metadata) %>%arrange(
  Water_regime) %>% filter(
  !Water_regime=="TO") %>% dplyr::select(
  -BarcodeSequence:-Water_regime) %>% column_to_rownames(
  var = "#SampleID") %>% t()

all_nr<- table_filter %>% dplyr::select_at(vars(
  matches("NR"))) %>% t()%>% as.data.frame(
)%>% rownames_to_column(
  var = "#SampleID") %>% inner_join(metadata) %>%arrange(
  Water_regime) %>% filter(
  !Water_regime=="TO") %>% dplyr::select(
  -BarcodeSequence:-Water_regime) %>% column_to_rownames(
  var = "#SampleID") %>% t()

list_pca<- list(all_ro, all_ri, all_nr)
zero_func <- function(x){ t(cmultRepl(t(x), method="CZM", output="p-counts"))}
clr_func<-function(x){t(CoDaSeq::codaSeq.clr(x ,samples.by.row = F))}

zero_list<- lapply(list_pca, zero_func)

## No. corrected values: 46
clr_list<- lapply(zero_list, clr_func)
pcx.abund_list <- lapply(clr_list, prcomp)

#create the base plot with only the arrows
PC1.f<- function(x,y){paste("PC1", round(sum(x$sdev[1] ^ 2) /
  mvar(y) * 100, 1), "%")}
PC2.f <- function(x,y){paste("PC2", round(sum(x$sdev[2] ^ 2) /
  mvar(y) * 100, 1), "%")}

PC1_all<- mapply(PC1.f, pcx.abund_list, clr_list)
PC2_all<- mapply(PC2.f, pcx.abund_list, clr_list)

list2<- list(pcx.abund_list[[1]],pcx.abund_list[[2]] , pcx.abund_list[[3]])

pca_tables2<- function(tab){ggplot() +
  geom_segment(data=data.frame(tab$rotation) %>%
    rownames_to_column(var = "FeatureID"))%>%
    mutate(a=sqrt(PC1^2+PC2^2)) %>%
    top_n(15, a) %>%
    mutate(PC1=PC1*20, PC2=PC2*20),
    aes(x=0, xend=PC1, y=0, yend=PC2),
    arrow = arrow(length = unit(0.3,"cm")))+

  geom_point(data=data.frame(tab$x) %>%
    rownames_to_column(var = "#SampleID"))%>%
    left_join(metadata, by = "#SampleID"),
    aes(x=PC1, y=PC2, fill=Water_regime),shape=21, size=4) +
  geom_vline(xintercept = 0, linetype = 2) +
  geom_hline(yintercept = 0, linetype = 2) +theme_light()+
  scale_x_continuous(limits = c(-25,25))+
  scale_y_continuous(limits = c(-20,20))+
  scale_fill_manual(values = c("#479330","#FFFF00", "#FF0000"))+

```

```

theme(axis.text = element_text(colour = "black", size = 10),
      axis.title = element_text(colour = "black", size = 10),
      legend.text = element_text(size = 10),
      axis.ticks = element_line(colour = "black"),
      axis.line = element_line(colour = "black"),
      axis.text.y.right = element_text(colour = "black"),
      axis.text.x.top = element_text(colour = "black"),
      legend.position = "top",
      legend.direction = "vertical",
      legend.box = "vertical",
      legend.title = element_blank()  }

figures_pca2<- lapply(list2, pca_tables2)
pca_ro_all<- figures_pca2[[1]]+ xlab(PC1_all[[1]]) +ylab(PC2_all[[1]]) + theme(
  legend.position="top", legend.title=element_blank())
pca_ri_all<- figures_pca2[[2]]+ xlab(PC1_all[[2]]) +ylab(PC2_all[[2]]) + theme(
  legend.position="top", legend.title=element_blank())
pca_nr_all<- figures_pca2[[3]]+ xlab(PC1_all[[3]]) +ylab(PC2_all[[3]]) + theme(
  legend.position="top", legend.title=element_blank())

pca_allthree<- plot_grid(pca_ro_all, pca_ri_all, pca_nr_all, ncol = 3,
  labels = c("C Roots", "E Rhizosphere",
            "G Bulk soil"), rel_widths = c(1,1,1) )

#PERMANOVA'S
clr_function_perma<-function(x){data.frame(x, check.names = F) %>%
  rownames_to_column(
  var = "ids")%>% column_to_rownames(var = "ids")}

meta_just_func2<- function(x){data.frame(x, check.names = F) %>%
  rownames_to_column(
  var = "#SampleID") %>% inner_join(metadata)}
clr_perma<- lapply(clr_list, clr_function_perma)
meta_just_list2<- lapply(clr_perma, meta_just_func2)

library(vegan)
perm<- how(nperm = 999)

setBlocks(perm)<- with(meta_just_list2[[1]], Plot)
a5<-adonis2(clr_perma[[1]]~Water_regime, data = meta_just_list2[[1]], method =
  "euclidian", permutations =perm) %>% round(., digits = 2) %>%replace(
  is.na(.), "-")

setBlocks(perm)<- with(meta_just_list2[[2]], Plot)
a6<-adonis2(clr_perma[[2]]~Water_regime, data = meta_just_list2[[2]], method =
  "euclidian", permutations =perm) %>% round(., digits = 2) %>%replace(

```

```

        is.na(.), "-")
setBlocks(perm) <- with(meta_just_list2[[3]], Plot)
a7<-adonis2(clr_perma[[3]]~Water_regime, data = meta_just_list2[[3]], method =
  "euclidian", permutations =perm) %>% round(., digits = 2) %>%replace(
  is.na(.), "-")

library(ggpubr)

Permanova_a5<-data.frame(a5, check.names = F) %>% rownames_to_column(
  var="Factor") %>% ggtexttable(., rows = NULL, theme = ttheme(
  "blank", base_size = 10)) %>%
  tab_add_hline(at.row = 1:2, row.side = "top", linewidth = 2)%>%
  tab_add_hline(at.row = c(4), row.side = "bottom", linewidth = 3, linetype = 1)

Permanova_a6<-data.frame(a6, check.names = F) %>% rownames_to_column(
  var="Factor") %>% ggtexttable(., rows = NULL, theme = ttheme(
  "blank", base_size = 10)) %>%
  tab_add_hline(at.row = 1:2, row.side = "top", linewidth = 2)%>%
  tab_add_hline(at.row = c(4), row.side = "bottom", linewidth = 3, linetype = 1)

Permanova_a7<-data.frame(a7, check.names = F) %>% rownames_to_column(
  var="Factor") %>% ggtexttable(., rows = NULL, theme = ttheme(
  "blank", base_size = 10)) %>%
  tab_add_hline(at.row = 1:2, row.side = "top", linewidth = 2)%>%
  tab_add_hline(at.row = c(4), row.side = "bottom", linewidth = 3, linetype = 1)

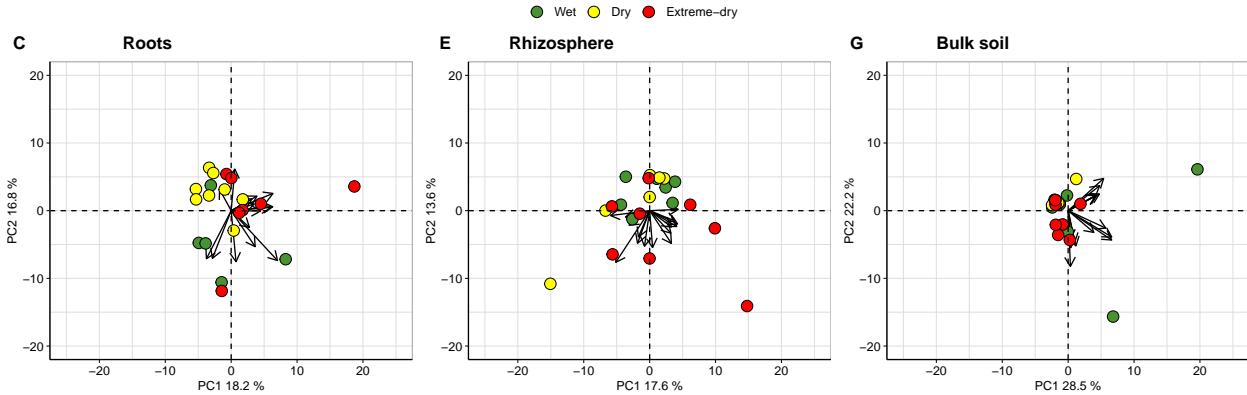
#all
perm_all_three<- plot_grid(Permanova_a5, Permanova_a6, Permanova_a7,
                           ncol = 3, labels = c("D", "F", "H"), label_y = 0.8)

pca_ro_all<- pca_ro_all+ theme(legend.position="top",
                                 legend.direction="horizontal")
legends<- get_legend(pca_ro_all)
pca_allthree<- plot_grid(NULL,legends, NULL,
                          pca_ro_all+theme(legend.position = "none"),
                          pca_ri_all+theme(legend.position = "none"),
                          pca_nr_all+theme(legend.position = "none"),
                          nrow = 2, ncol = 3, rel_heights = c(0.5,2),
                          labels = c("", "", "", "C Roots",
                                     "E Rhizosphere",
                                     "G Bulk soil"),
                          rel_widths = c(1,1,1), label_x = -0.16,
                          label_y = 1, hjust = -0.5, vjust = -0.5 )

plot_all_three<- plot_grid( pca_allthree, perm_all_three,
                             nrow = 2, rel_heights = c(1.5,1))

plot_all_three

```



D

Factor	Df	SumOfSqs	R2	F	Pr(>F)
Water_regime	2	535.35	0.17	1.69	0.01
Residual	17	2686.93	0.83	—	—
Total	19	3222.27	1.00	—	—

F

Factor	Df	SumOfSqs	R2	F	Pr(>F)
Water_regime	2	471.47	0.11	1.08	0.28
Residual	18	3929.93	0.89	—	—
Total	20	4401.40	1.00	—	—

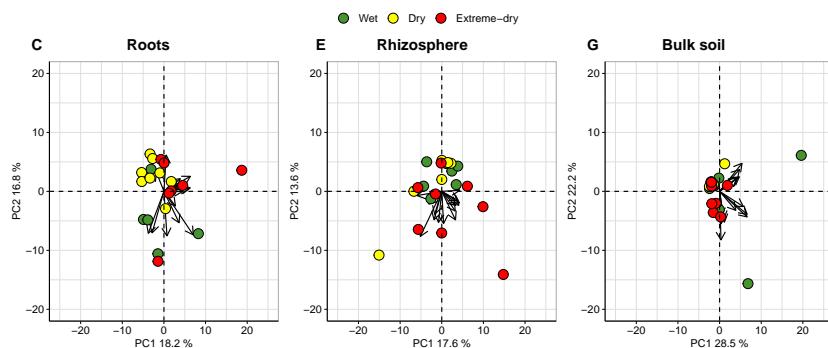
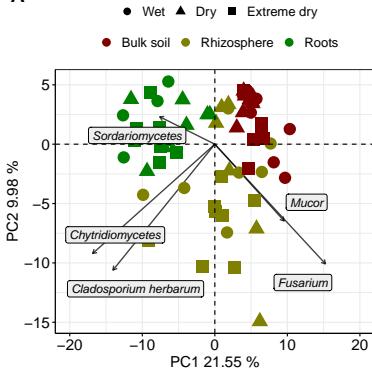
H

Factor	Df	SumOfSqs	R2	F	Pr(>F)
Water_regime	2	188.52	0.11	1.29	0.1
Residual	21	1532.48	0.89	—	—
Total	23	1721.00	1.00	—	—

#joining plots

```
per_pca<-plot_grid(b,plot_all_three, rel_widths = c(1,2))
per_pca
```

A



B

Factor	Df	SumOfSqs	R2	F	Pr(>F)
Water_regime	2	463.088	0.039	1.583	0.023
Compartment	2	2594.214	0.216	8.867	0.001
Water_regime:Compartment	4	733.834	0.061	1.254	0.088
Residual	56	8191.646	0.684	—	—
Total	64	11982.782	1.000	—	—

D

Factor	Df	SumOfSqs	R2	F	Pr(>F)	Factor	Df	SumOfSqs	R2	F	Pr(>F)	Factor	Df	SumOfSqs	R2	F	Pr(>F)
Water_regime	2	535.35	0.17	1.69	0.01	Water_regime	2	471.47	0.11	1.08	0.28	Water_regime	2	188.52	0.11	1.29	0.1
Residual	17	2686.93	0.83	—	—	Residual	18	3929.93	0.89	—	—	Residual	21	1532.48	0.89	—	—
Total	19	3222.27	1.00	—	—	Total	20	4401.40	1.00	—	—	Total	23	1721.00	1.00	—	—

F

H

```
#ggsave('..../Figures_final/Fig2.Permanova_PCA_NEW.png',
# width = 16, height = 8, dpi = 300, plot =per_pca)
#ggsave('..../Figures_final/Permanova-pairwise_filt.pdf',
# width = 6, height = 4, dpi = 300, plot =Pairwsie_permanova)
```

Aldex analysis

```
#Loading libraries
library(dplyr)
library(stringr)
library(purrr)
library(tidyverse)
library(tibble)
```

```

library(tidyr)
library(qiime2R)
library(ALDEx2)
library(tidyverse)
library(ComplexHeatmap)
library(viridis)
library(circlize)
library(RColorBrewer)
library(cowplot)
library(tidyverse)
library(qiime2R)
library(cowplot)
require(compositions)
require(zCompositions)
library(ggrepel)
library(vegan)
library(ggpubr)
library(RVAideMemoire)

relabunda<- function(x){(t(t(x)/colSums(x)))*100}

#Importing data
table_240<- read_qza("../Data/grouped_240_fil_noplant_nous.qza")$data
metadata<- read_tsv("../Data/FINALMAP_GROUPED.txt")
table_240<- table_240 %>% t() %>% data.frame(
  .,check.names = F) %>% rownames_to_column(
  var = "#SampleID") %>% inner_join(
  metadata) %>% arrange(Type_of_soil) %>% dplyr::select(
  -Type_of_soil:-Month) %>% column_to_rownames(
  var = "#SampleID") %>% t() %>% as.data.frame()
table_240_t<- table_240 %>% t()
taxonomy_240<- read_qza("../Data/taxonomy_blast_240_0.97.qza")$data

#remove non-fungi taxa

taxonomy_filter<- taxonomy_240 %>% filter(
  !str_detect(Taxon, "ozoa"))%>% filter(
  !str_detect(Taxon, "helida")) %>% filter(
  !str_detect(Taxon, "ophyta")) %>% filter(
  !str_detect(Taxon, "Ciliophora")) %>% filter(
  !str_detect(Taxon, "Nucleariidae_and_Fonticula_group")) %>% filter(
  !str_detect(Taxon, "Arthrioida")) %>% filter(
  !str_detect(Taxon, "Labyrinthulomycetes")) %>% filter(
  !str_detect(Taxon, "Apicomplexa")) %>% filter(
  !str_detect(Taxon, "Bicosoecida")) %>% filter(
  !str_detect(Taxon, "Breviatea")) %>% filter(
  !str_detect(Taxon, "Aphelidea")) %>% filter(
  !str_detect(Taxon, "Arthropoda"))

table_filter<- table_240[match(taxonomy_filter$Feature.ID,
                                rownames(table_240)),] %>% drop_na()

#by taxonomic levels

```

```

parse_taxa_240<- parse_taxonomy(taxonomy_filter)

summarize_taxa_240<-summarize_taxa(table_filter,
                                      parse_taxa_240)
Phylum_240<- summarize_taxa_240$Phylum
Class_240<- summarize_taxa_240$Class
Genus_240<- summarize_taxa_240$Genus

#analysis kruskal wallis all levels
conds<- c(rep("NR", 9), rep("RI", 9), rep("RO", 9))

#auss
aldex_analysis<- aldex.clr(table_filter,
                           mc.samples = 1000, denom="all", verbose = TRUE, conds )

differentials <- aldex.kw(aldex_analysis, useMC = F, verbose = F)
aldex_240<- differentials %>% rownames_to_column(
  var = "Feature.ID") %>% left_join(taxonomy_240)%>% filter(kw.eBH < 0.05)

#taxonomic levels
aldex_list<- list(Phylum_240, Class_240, Genus_240)

aldex_analysis_function<-function(x){ aldex.clr(x,
mc.samples = 1000, denom="all", verbose = TRUE, conds)}

aldex_analysis<- lapply(aldex_list, aldex_analysis_function)

differentials_function <- function(x){aldex.kw(x, useMC = F, verbose = F)}

differentials_tables_class <- lapply(aldex_analysis, differentials_function)

aldex_phylum<- differentials_tables_class[[1]] %>% rownames_to_column(
  var = "Feature.ID") %>% filter(kw.eBH<0.05)
aldex_class<- differentials_tables_class[[2]] %>% rownames_to_column(
  var = "Feature.ID") %>% filter(kw.eBH<0.05)
aldex_genus<- differentials_tables_class[[3]] %>% rownames_to_column(
  var = "Feature.ID") %>% filter(kw.eBH<0.05)

#Paired test just with asv's
table_240_t<- table_filter %>% t()
otutable1<- data.frame(t(table_240_t),
                        check.names = F, stringsAsFactors = F) %>% dplyr::select_at(
  vars(contains("Roots")))
otutable2<- data.frame(t(table_240_t),
                        check.names = F, stringsAsFactors = F) %>% dplyr::select_at(
  vars(starts_with("Ri")))
otutable3<- cbind(otutable1, otutable2)

otutable4<- data.frame(t(table_240_t),

```

```

            check.names = F, stringsAsFactors = F) %>% dplyr::select_at(
            vars(contains("Roots")))
otutable5<- data.frame(t(table_240_t),
            check.names = F, stringsAsFactors = F) %>% dplyr::select_at(
            vars(starts_with("N")))
otutable6<- cbind(otutable4, otutable5)

otutable7<- data.frame(t(table_240_t),
            check.names = F, stringsAsFactors = F)%>%dplyr::select_at(
            vars(contains("Non")))
otutable8<- data.frame(t(table_240_t),
            check.names = F, stringsAsFactors = F) %>% dplyr::select_at(
            vars(starts_with("Ri")))
otutable9<- cbind(otutable7, otutable8)

conds<- c(rep("A", 9), rep("B", 9))

aldex_list_paired<- list(otutable3, otutable6, otutable9)

aldex_analysis_function_paired<-function(x){ aldex(x,
    mc.samples = 1000, denom="all", verbose = TRUE, conds, effect = T, test = "t")}

aldex_analysis_paired<- lapply(aldex_list_paired, aldex_analysis_function_paired)

aldex_ro_ri<-aldex_analysis_paired[[1]] %>% rownames_to_column(
  var = "Feature.ID") %>% filter(wi.eBH < 0.05) %>% left_join(taxonomy_filter)
aldex_ro_nr<- aldex_analysis_paired[[2]] %>% rownames_to_column(
  var = "Feature.ID") %>% filter(wi.eBH < 0.05) %>% left_join(taxonomy_filter)
aldex_nr_ri<- aldex_analysis_paired[[3]] %>% rownames_to_column(
  var = "Feature.ID") %>% filter(wi.eBH < 0.05) %>% left_join(taxonomy_filter)

#write.table(aldex_nr_ri, "dif_aldex_nr_ri.tsv", sep = "\t", row.names = T)

#Aldex for functional predictions of Picrust2

EC_predicted <- read.delim("../Data/pathways_table_all.txt",
  check.names = F) %>% dplyr::select(
  -"#OTU ID") %>% column_to_rownames(
  var = "taxonomy")

Alpha.t_asv_table<- read.delim("../Data/alpha_all.tsv")

table<- EC_predicted %>% t() %>% as.data.frame() %>% rownames_to_column(
  var = "sampleid")
table_meta<- table %>% inner_join(Alpha.t_asv_table) %>% group_by(
  Type, Treatment, Plot) %>% summarise_if(is.numeric, sum) %>% unite(
  "ids",Type:Plot, sep = "_") %>% column_to_rownames(
  var = "ids") %>% dplyr::select(-Loc:-value) %>% t() %>%
  as.data.frame() %>% mutate_if(

```

```

    is.numeric, as.integer)

table_nr<- table_meta %>% dplyr::select_at(vars(contains("Bulk")))
table_ri<- table_meta %>% dplyr::select_at(vars(contains("Rhizosphere")))
table_ro<- table_meta %>% dplyr::select_at(vars(contains("Root")))

table_nr_ri<- cbind(table_nr, table_ri)
table_nr_ro<- cbind(table_nr, table_ro)
table_ro_ri<- cbind(table_ro, table_ri)

conds1<- c(rep("bs", 9), rep("ri", 9))
conds2<- c(rep("bs", 9), rep("ro", 9))
conds3<- c(rep("ro", 9), rep("ri", 9))

library(ALDEEx2)
aldex_nr_ri<-aldex(table_nr_ri, conditions = conds1, mc.samples = 1000,
                     test = "t", effect = T, denom = "all")
aldex_nr_ro<-aldex(table_nr_ro, conditions = conds2, mc.samples = 1000,
                     test = "t", effect = T, denom = "all")
aldex_ro_ri<-aldex(table_nr_ri, conditions = conds3, mc.samples = 1000,
                     test = "t", effect = T, denom = "all")

#write.table(aldex_nr_ri, "aldex_nr_ri_funct.txt", sep = "\t")
#write.table(aldex_nr_ro, "aldex_nr_ro_funct.txt", sep = "\t")
#write.table(aldex_ro_ri, "aldex_ro_ri_funct.txt", sep = "\t")

```

Let's plot heatmap with aldex values!

```

#loading files
meta<-read_tsv("../Data/FINALMAP_GROUPED.txt")
table_240<- read_qza("../Data/grouped_240_fil_noplant_nous.qza")$data %>% as.data.frame()
taxonomy_240<- read_qza("../Data/taxonomy_blast_240_0.97.qza")$data
#remove non-fungi taxa
taxonomy_filter<- taxonomy_240 %>% filter(
  !str_detect(Taxon, "ozoa"))%>% filter(
  !str_detect(Taxon, "helida")) %>% filter(
  !str_detect(Taxon, "ophyta")) %>% filter(
  !str_detect(Taxon, "Ciliophora")) %>% filter(
  !str_detect(Taxon, "Nucleariidae_and_Fonticula_group")) %>% filter(
  !str_detect(Taxon, "Arthrioida")) %>% filter(
  !str_detect(Taxon, "Labyrinthulomycetes")) %>% filter(
  !str_detect(Taxon, "Apicomplexa")) %>% filter(
  !str_detect(Taxon, "Bicosoecida")) %>% filter(
  !str_detect(Taxon, "Breviatea")) %>% filter(
  !str_detect(Taxon, "Aphelidea")) %>% filter(
  !str_detect(Taxon, "Arthropoda"))

table_filter<- table_240[match(taxonomy_filter$Feature.ID, rownames(table_240)),] %>% drop_na()

#by taxonomic levels
parse_taxa_240<- parse_taxonomy(taxonomy_filter)

summarize_taxa_240<-summarize_taxa(table_filter, parse_taxa_240)

```

```

Phylum_240<- summarize_taxa_240$Phylum
Class_240<- summarize_taxa_240$Class
Genus_240<- summarize_taxa_240$Genus

#PHYLUM
phylum_240<- summarize_taxa_240$Phylum %>% mutate(means = rowMeans(.)) %>% mutate_at(
  "means", as.numeric) %>% arrange(-means)
phylum_240_abund<-data.frame( relabunda(phylum_240[1:28]) , check.names = F)

phylum_240_relab<-phylum_240_abund %>% dplyr::select_at(
  vars(!contains("0"))) %>% dplyr::slice(
  c(1:7)) %>% rownames_to_column(var = "#OTU ID") %>% filter(
  !`#OTU ID` == "d_Eukaryota; Phragmoplastophyta") %>% dplyr::select(-means)

phylum_aldex_240<- read_tsv("../Data/aldex_results/dif_240_phylum.tsv") %>%
  rename("#OTU ID"=Feature.ID) %>%
  right_join( phylum_240_relab)%>%dplyr::mutate_at(
    c(1),~str_extract(., "[^;]+$")) %>% column_to_rownames(var = "#OTU ID") %>%
  mutate_at(c("kw.eBH"), funs(p.value = case_when(
    . <= 0.001 ~ "<0.001",
    . > 0.001 & . <= 0.01 ~ "<0.01",
    . > 0.01 & . < 0.05 ~ "<0.05",
    . >= 0.05 ~ ">0.05")))%>% rownames_to_column(
    var = "ids") %>% filter(!ids == " NA") %>% arrange(
    ids) %>% column_to_rownames(var = "ids")

heatmap<- phylum_240_relab %>% column_to_rownames(
  var = "#OTU ID") %>% t() %>% as.data.frame() %>% rownames_to_column(
  var = "#SampleID") %>% inner_join(meta) %>% group_by(
    Type_of_soil, Treatments) %>% summarise_if(
    is.numeric, mean) %>% unite(
    "ids", Type_of_soil:Treatments) %>% column_to_rownames(
    var = "ids") %>%dplyr::select_at(vars(
      contains("d_")),~str_extract(
        ., "[^;]+$")) %>% t() %>% as.data.frame( ) %>% rownames_to_column(
        var = "ids") %>% arrange(ids) %>% column_to_rownames(var = "ids")

#CLASS
class_240<- summarize_taxa_240$Class %>% mutate(
  means = rowMeans(.)) %>% mutate_at(
  "means", as.numeric) %>% arrange(-means)
class_240_abund<-data.frame(relabunda(class_240[1:28]) , check.names = F)
class_240_relab<-class_240_abund %>% dplyr::select_at(
  vars(!contains("0"))) %>% dplyr::slice(
  c(1:20)) %>% rownames_to_column(
  var = "#OTU ID") %>% filter(
  !`#OTU ID` == "d_Eukaryota; Phragmoplastophyta") %>% dplyr::select(-means)

remove<- c(" Pezizomycetes", " Aconoidasida" , " Chromadorea" ,
          " Aphelidea", " Discosea" , " Imbricatea" )

```

```

class_aldex_240<- read_tsv(
  "../Data/aldex_results/dif_240_class.tsv") %>% rename(
  "#OTU ID"=Feature.ID) %>% right_join(
  class_240_relab)%>%dplyr::mutate_at(c(1),~str_extract(., "[^;]+$"))%>% filter(
  !`#OTU ID` == " Incertae_Sedis" & !`#OTU ID` == " NA" & !
  `#OTU ID` == " Eurotiomycetes" ) %>% arrange(
  `#OTU ID`)%>% column_to_rownames(var = "#OTU ID") %>%
  mutate_at(c("kw.eBH"), funs(p.value = case_when(
    . <= 0.001 ~ "<0.001",
    . > 0.001 & . <= 0.01 ~ "<0.01",
    . > 0.01 & . < 0.05 ~ "<0.05",
    . >= 0.05 ~ ">0.05")))) %>% rownames_to_column(
  var = "ids" ) %>% filter(!ids %in% remove) %>% arrange(
  ids) %>% column_to_rownames(var = "ids")

heatmap_class<- class_240_relab %>% column_to_rownames(
  var = "#OTU ID") %>% t() %>% as.data.frame(
) %>% rownames_to_column(
  var = "#SampleID") %>% inner_join(meta) %>% group_by(
  Type_of_soil, Treatments) %>% summarise_if(
  is.numeric, mean) %>% unite(
  "ids", Type_of_soil:Treatments) %>% column_to_rownames(
  var = "ids") %>%dplyr::select(
  -"d_Eukaryota; Zoopagomycota; Incertae_Sedis" ,
  -"d_Eukaryota; Mucoromycota; Incertae_Sedis") %>% dplyr::select_at(vars(
  contains("d_")),~str_extract(
  ., "[^;]+$")) %>% t() %>% as.data.frame( ) %>% rownames_to_column(
  var = "ids") %>% filter(!ids %in% remove) %>% arrange(
  ids) %>% filter(!ids==" Eurotiomycetes") %>% column_to_rownames(var = "ids")

#GENUS
genus_240<- summarize_taxa_240$Genus %>% mutate(
  means = rowMeans(.) ) %>%mutate_at(
  "means", as.numeric) %>% arrange(-means)

genus_240_abund<-data.frame( relabunda(genus_240[1:28]) , check.names = F)
genus_240_relab<-genus_240_abund %>% dplyr::select_at(vars(!contains("0")))) %>% dplyr::slice(
  c(1:27)) %>% rownames_to_column(var = "#OTU ID") %>% filter(
  !`#OTU ID` == "d_Eukaryota; Phragmoplastophyta")
remove2<- c(" Ophiostoma", " Naganishia" , " Aspergillus" , " Magnoliophyta" )

genus_aldex_240<- read_tsv("../Data/aldex_results/dif_240_genus.tsv")%>%
  rename("#OTU ID"=Feature.ID) %>% right_join(
  genus_240_relab)%>%dplyr::mutate_at(c(1),~str_extract(., "[^;]+$"))%>% filter(
  !`#OTU ID` == " Incertae_Sedis" & !`#OTU ID` == " NA" ) %>% arrange(
  `#OTU ID`)%>% column_to_rownames(var = "#OTU ID") %>%
  mutate_at(c("kw.eBH"), funs(p.value = case_when(
    . <= 0.001 ~ "<0.001",
    . > 0.001 & . <= 0.01 ~ "<0.01",
    . > 0.01 & . < 0.05 ~ "<0.05",
    . >= 0.05 ~ ">0.05")))) %>% rownames_to_column(

```

```

var = "ids") %>% filter(
  !ids %in% remove2) %>% arrange(ids) %>% mutate(ids=case_when(
    ids == " uncultured" ~ " uncultured Glomerales",
    TRUE~ as.character(ids))) %>%
column_to_rownames(var = "ids")

heatmap_genus<- genus_240_relab %>% column_to_rownames(
  var = "#OTU ID") %>% t() %>% as.data.frame(
) %>% rownames_to_column(
  var = "#SampleID") %>% inner_join(meta) %>% group_by(
  Type_of_soil, Treatments) %>% summarise_if(
  is.numeric, mean) %>% unite(
  "ids", Type_of_soil:Treatments) %>% column_to_rownames(
  var = "ids") %>% dplyr::select_at(vars(
  -ends_with("NA"))) %>% dplyr::select_at(vars(
  contains("d_")),~str_extract(
    ., "[^;]+$")) %>% t() %>% as.data.frame( ) %>% rownames_to_column(
  var = "ids") %>% filter(
    !ids %in% remove2) %>% arrange(ids) %>% mutate(ids=case_when(
    ids == " uncultured" ~ " uncultured Glomerales",
    TRUE~ as.character(ids))) %>%
column_to_rownames(var = "ids")

#ASV's

asv_240<- data.frame(table_filter, check.names = F) %>% mutate(
  means = rowMeans(..)) %>% mutate_at(
  "means", as.numeric) %>% arrange(-means)%>% rownames_to_column(
  var = "Feature.ID")%>% inner_join(
  taxonomy_240)%>% dplyr::select(-Consensus)

asv_240_abund<-data.frame(asv_240[c(1,30)],
                           relabunda(asv_240[2:28]) , check.names = F)

names<- asv_240_abund[2]

asv_aldex_240<- read.delim("../Data/aldex_results/dif_240_asv.tsv") %>%
  rename("#OTU ID"=Feature.ID) %>% filter(
  kw.eBH < 0.05) %>% dplyr::mutate_at(c("Taxon"),~str_extract(., "[^;]+$"))%>% filter(
  !`#OTU ID` == " Incertae_Sedis" & !`#OTU ID` == " NA" ) %>% arrange(
  `#OTU ID`)%>% remove_rownames() %>% column_to_rownames(var = "#OTU ID") %>%
  mutate_at(c("kw.eBH"), funs(p.value = case_when(
    . <= 0.001 ~ "<0.001",
    . > 0.001 & . <= 0.01 ~ "<0.01",
    . > 0.01 & . < 0.05 ~ "<0.05",
    . >= 0.05 ~ ">0.05")))%>% rownames_to_column(
  var = "Feature.ID") %>% arrange(Taxon) %>% drop_na(.) %>% remove_rownames(
  )%>% column_to_rownames(var = "Taxon")

```

```

asv_240_relab<-asv_240_abund %>% dplyr::select_at(
  vars(!contains("0")) ) %>% right_join(
    asv_aldex_240, by = c("Feature.ID")) %>% dplyr::select(
      Taxon, `Non-rizospheric22`Roots11) %>% drop_na(.) %>% column_to_rownames(var = "Taxon")

heatmap_asv<- asv_240_relab %>% t() %>% as.data.frame(
) %>% rownames_to_column(
  var = "#SampleID") %>% inner_join(meta) %>% group_by(Type_of_soil, Treatments) %>% summarise_if(
  is.numeric, mean) %>% unite(
    "ids", Type_of_soil:Treatments) %>% column_to_rownames(
      var = "ids") %>% dplyr::select_at(vars(
    contains("__")), ~str_extract(., "[^;]+$")) %>% t() %>% as.data.frame(
  ) %>% rownames_to_column(var = "ids") %>% arrange(
    ids) %>% column_to_rownames(var = "ids")

```

Heatmap!

```

col_fun2 = colorRamp2(c(0, 1, 1+1e-5, 10, 10+1e-5, 50, 50+1e-5, 100),
                      viridis(8, option = "D", direction = -1))

```

```

#pvalue annotation
mypalette<-brewer.pal(10,"BrBG")
cols_pvalue <- list('Adj. p-value' = c("<0.001" = '#AB0000',
                                         "<0.01" = '#FF0000',
                                         "<0.05" = "#EC7063",
                                         ">0.05" = "#F9EBEA"))

```

```

split = rep(1:3, each = 3)
treats<- c("1.Wet", "2.Dry", "3.Extreme-dry")
cols_ho<- list("Water regime" = c("1.Wet" = '#479330',
                                     "2.Dry" = '#FFFF00',
                                     "3.Extreme-dry" = "#FF0000"))

```

```

annP = HeatmapAnnotation("Adj. p-value" = phylum_aldex_240$p.value,
                        which = "row", col = cols_pvalue,
                        show_legend = F, gp = gpar(col = "white"),
                        show_annotation_name = F)

```

```

ha = HeatmapAnnotation(foo = anno_block(gp = gpar(
  fill = c("#800000", "#808000", "#008000", "#D35400")),
  labels = c("Bulk soil", "Rhizosphere", "Roots"),
  labels_gp = gpar(col = "white", fontsize = 7, fontface= "bold")))

```

```

ho = HeatmapAnnotation("Water regime" = c(rep(treats, 3)),
                       which = "col", col = cols_ho,
                       annotation_name_gp = gpar(fontsize=10),
                       show_legend = T, gp = gpar(col = "white", fontize=12),
                       show_annotation_name = T)

```

```

ht1<-Heatmap(heatmap,col = col_fun2,
             heatmap_legend_param = list(direction = "horizontal",

```

```

                    title = "Relative \n abundance (%)",
                    at = c(0,1, 10,  50, 100), break_dist = 1),
rect_gp = gpar(col = "white", lwd = 1),
cluster_columns = F, cluster_rows = T,
show_heatmap_legend = TRUE, top_annotation = c(ha, ho),
height = nrow(heatmap)*unit(6, "mm"),
row_dend_width = unit(0.5, "cm"),right_annotation = c(annP),
column_order = sort(colnames(heatmap)),width = ncol(heatmap)*unit(3, "mm"),
column_split = split, column_title = NULL,
row_names_gp = gpar(fontsize=12),row_title = "Phylum",
column_title_gp = gpar(
fill = c("#800000" , "#808000" , "#008000", "#D35400", "#2E4053" )))

class_aldex<- class_aldex_240 %>% drop_na(.)
annP2 = HeatmapAnnotation("Adj. p-value" = class_aldex$p.value,
                           which = "row", col = cols_pvalue,name = ".",
                           show_annotation_name = F,
                           show_legend = T, gp = gpar(col = "white"))

heat_class<- heatmap_class %>% rownames_to_column(var = "ids") %>% inner_join(
  class_aldex %>% mutate(ids = rownames(class_aldex)),
  by = 'ids') %>% column_to_rownames(var = "ids") %>% dplyr::select(
  `Non-rizospheric_TC`:`Roots_TSD`)
ht2<-Heatmap(heat_class,col = col_fun2,
            heatmap_legend_param = list(direction = "horizontal",
                                         col_fun = col_fun2,
                                         title = "Relative \n abundance (%)",
                                         at = c(0,1, 10,  50, 100),
                                         break_dist = 1),
            rect_gp = gpar(col = "white", lwd = 1),
            height = nrow(ht1)*unit(8, "mm"),
            row_title = "Class",width = ncol(heatmap)*unit(3, "mm"),
            row_dend_width = unit(0.5, "cm"),cluster_columns = F,
            cluster_rows = T,show_heatmap_legend = FALSE,
            row_names_gp = gpar(fontsize=12), right_annotation = c(annP2))

annP3 = HeatmapAnnotation("Adj. p-value" = genus_aldex_240$p.value,
                           which = "row", col = cols_pvalue,name = ".",
                           show_annotation_name = F,
                           show_legend = T, gp = gpar(col = "white"))

ht3<-Heatmap(heatmap_genus,col = col_fun2,
            heatmap_legend_param = list(direction = "horizontal",
                                         col_fun = col_fun2,
                                         title = "Relative \n abundance (%)",
                                         at = c(0,1, 10,  50, 100),
                                         break_dist = 1),
            rect_gp = gpar(col = "white", lwd = 1),
            height = nrow(ht1)*unit(8, "mm"),
            row_title = "Genus",width = ncol(heatmap)*unit(3, "mm"),
            row_dend_width = unit(0.5, "cm"),cluster_columns = F,
            cluster_rows = T,show_heatmap_legend = FALSE,
            row_names_gp = gpar(fontsize=12), right_annotation = c(annP3))

```

```

                        at = c(0,1, 10, 50, 100),
                        break_dist = 1),
rect_gp = gpar(col = "white", lwd = 1), cluster_columns = F,
cluster_rows = T, show_heatmap_legend = FALSE, row_title = "Genus",
height = nrow(ht1)*unit(9, "mm"),
width = ncol(heatmap)*unit(1, "mm"),
row_dend_width = unit(0.5, "cm"), right_annotation = c(annP3),
column_order = sort(colnames(heatmap_genus)),
row_names_gp = gpar(fontsize=12),
column_split = split, column_title = NULL)

annP4 = HeatmapAnnotation('Adj. p-value' = asv_aldex_240$p.value,
                         which = "row", col = cols_pvalue, name = ".",
                         annotation_name_gp = gpar(fontsize=10),
                         show_legend = F, gp = gpar(col = "white"),
                         show_annotation_name = F)

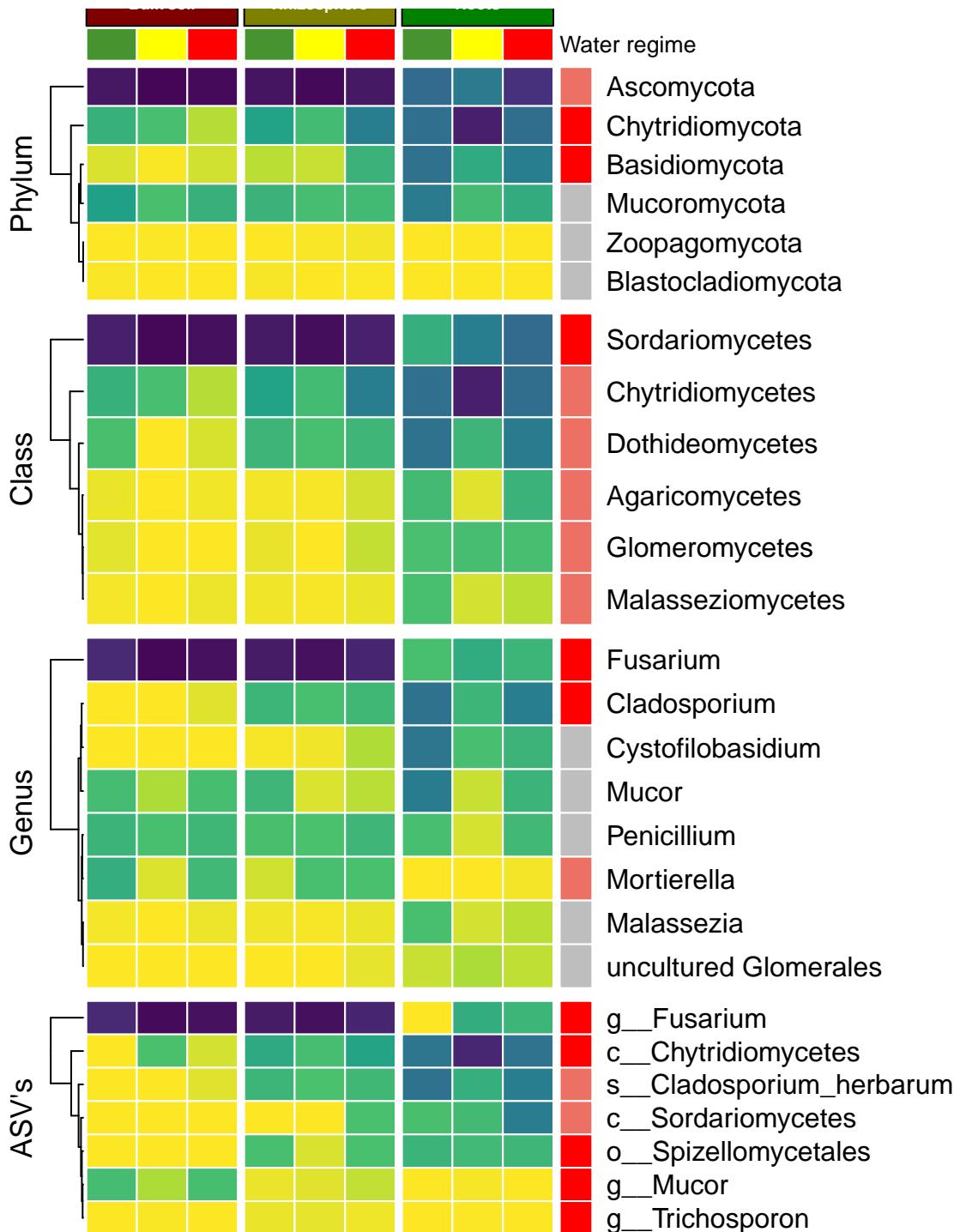
ht4<-Heatmap(heatmap_asv, col = col_fun2,
              heatmap_legend_param = list(direction = "horizontal",
                                            col_fun = col_fun2,
                                            title = "Relative \n abundance (%)",
                                            at = c(0,1, 10, 50, 100),
                                            break_dist = 1),
              rect_gp = gpar(col = "white", lwd = 1), row_title = "ASV's",
              cluster_columns = F, cluster_rows = T,
              show_heatmap_legend = FALSE, row_names_gp = gpar(fontsize=12),
              width = ncol(heatmap)*unit(4, "mm"),
              height = nrow(ht1)*unit(6, "mm"),
              row_dend_width = unit(0.5, "cm"), right_annotation = c(annP4),
              column_order = sort(colnames(heatmap_asv)), show_column_names = F)

#ht1%v%ht2%v%ht3%v%ht4
ht_list = ht1 %v% ht2 %v% ht3 %v% ht4

heatm<-grid.grabExpr(draw(ht_list, heatmap_legend_side = "bottom",
                           annotation_legend_side = "bottom",
                           merge_legend=T, width = ncol(ht1)*unit(8, "mm")))

plot_grid(heatm)

```



Water regime	Relative	Adj. p-value	Aldex paired plots
#load and format files			
aldex_ro_ri<- read.delim("../Data/aldex_results/dif_aldex_ro_ri.tsv",			
) %>%			
mutate_at(c("diff btw"), funs(Type = case_when(

```

. < 0 ~ "Roots",
. > 0 ~ "Rhizosphere")) )%>% arrange(diff.btw)%>%mutate(
  taxa= str_extract(Taxon, "[^_]+$"))%>%
mutate(taxa=case_when(
  taxa=="Sordariomycetes" ~ "c__Sordariomycetes",
  taxa=="Trichosporon" ~ "g__Trichosporon",
  taxa=="Fusarium" ~ "g__Fusarium",
  taxa=="Mucor" ~ "g__Mucor" )) %>%
mutate_at(c("wi.eBH"), funs(p.value = case_when(
  . <= 0.001 ~ "<0.001",
  . > 0.001 & . <= 0.01 ~ "<0.01",
  . > 0.01 & . < 0.05 ~ "<0.05",
  . >= 0.05 ~ ">0.05")))

```



```

aldex_nr_ri<- read.delim("../Data/aldex_results/dif_aldex_nr_ri.tsv" ) %>%
  mutate_at(c("diff.btw"), funs(Type = case_when(
    . < 0 ~ "Non-rhizospheric",
    . > 0 ~ "Rhizosphere")) )%>% arrange(diff.btw)%>%mutate(
  taxa= str_extract(Taxon, "[^_]+$"))%>%
mutate(taxa=case_when(
  taxa=="Spizellomycetales" ~ "o__Spizellomycetales",
  taxa=="Trichosporon" ~ "g__Trichosporon",
  taxa=="Chytridiomycetes" ~ "c__Chytridiomycetes")) %>%
mutate_at(c("wi.eBH"), funs(p.value = case_when(
  . <= 0.001 ~ "<0.001",
  . > 0.001 & . <= 0.01 ~ "<0.01",
  . > 0.01 & . < 0.05 ~ "<0.05",
  . >= 0.05 ~ ">0.05")))

```



```

aldex_ro_nr<- read.delim("../Data/aldex_results/dif_aldex_ro_nr.tsv") %>%
  mutate_at(c("diff.btw"), funs(Type = case_when(
    . < 0 ~ "Roots",
    . > 0 ~ "Non-rhizospheric")) )%>% arrange(diff.btw)%>%mutate(
  taxa= str_extract(Taxon, "[^_]+$"))%>%
mutate_at(c("wi.eBH"), funs(p.value = case_when(
  . <= 0.001 ~ "<0.001",
  . > 0.001 & . <= 0.01 ~ "<0.01",
  . > 0.01 & . < 0.05 ~ "<0.05",
  . >= 0.05 ~ ">0.05")))%>%
mutate_at(c("taxa"), funs(taxa = case_when(
  taxa=="herbarum" ~ "s__Cladosporium herbarum",
  taxa=="Chytridiomycetes" ~ "c__Chytridiomycetes",
  taxa=="Fusarium" ~ "g__Fusarium",
  taxa=="Spizellomycetales" ~ "o__Spizellomycetales",
  taxa=="Sordariomycetes" ~ "c__Sordariomycetes",
  taxa=="Mucor" ~ "g__Mucor" )))

```



```

cols_pvalue <- list('Adj. p-value' = c(
  "<0.001" = '#C70039',

```

```

"<0.01"="#FF5733",
"<0.05"="#FFC300",
">0.05"="#DAF7A6"))
effect_col_fun =colorRamp2(
  c(-3.5, 0, 3.5),
  c("lightsalmon4", "white", "lightseagreen"))

aldex_nr_ro_funct<- read.delim("../Data/aldex_nr_ro_funct.txt" ) %>% rownames_to_column(
  var = "Taxon") %>% filter(
  wi.eBH<0.05) %>% mutate_at(c("diff.btw"), funs(Type = case_when(
    . < 0 ~ "Non-rhizospheric",
    . > 0 ~ "Roots")))%>% arrange(diff.btw)%>%mutate(
  taxa= str_extract(Taxon, "[^_]+$"))%>%
  mutate_at(c("wi.eBH"), funs(p.value = case_when(
    . <= 0.001 ~ "<0.001",
    . > 0.001 & . <= 0.01 ~ "<0.01",
    . > 0.01 & . < 0.05 ~ "<0.05",
    . >= 0.05 ~ ">0.05")))
}

heat<- aldex_ro_nr %>% dplyr::select(
  taxa, "RO"= rab.win.A, "BS"= rab.win.B)
heat1<- heat[c(-5),] %>% remove_rownames() %>% column_to_rownames(var = "taxa")
heat2<- aldex_nr_ri %>% dplyr::select(
  taxa, "BS"= rab.win.A, "R"= rab.win.B) %>% remove_rownames() %>% column_to_rownames(var = "taxa")
heat3<- aldex_ro_ri %>% dplyr::select(
  taxa, "RO"= rab.win.A, "R"= rab.win.B) %>% remove_rownames() %>% column_to_rownames(var = "taxa")
heat4<- aldex_nr_ro_funct %>% dplyr::select(
  taxa, "BS"= rab.win.bs, "RO"= rab.win.ro) %>% remove_rownames() %>% column_to_rownames(var = "taxa")

```

Let's plot!

```

treatment_col = structure(c("#008000", "#800000"),
                           names = c("Roots", "Non-rhizospheric"))

barpl = rowAnnotation("difference \nbetween groups" = anno_barplot(
  aldex_ro_nr[c(-5),]$diff.btw, which = "row",
  gp = gpar(fill = treatment_col[aldex_ro_nr[c(-5),]$Type]),
  width = unit(3, "cm")), show_annotation_name = F,
  annotation_name_gp =gpar(fontsize = 0),
  annotation_name_rot = 0)

annP = HeatmapAnnotation("Adj. p-value" =aldex_ro_nr[c(-5),]$p.value,
                        simple_anno_size = unit(0.35, "cm"),
                        annotation_name_gp =gpar(fontsize = 7),
                        which = 'row',
                        annotation_legend_param = list(
                          title_gp = gpar(fontsize = 9, fontface="bold"),
                          labels_gp = gpar(fontsize = 9),
                          direction ="vertical"),
                        col = cols_pvalue,
                        show_legend = F, gp = gpar(col = "white"),
                        show_annotation_name = T)

```

```

annE = rowAnnotation("Effect size" = aldex_ro_nr[c(-5),]$effect,
                     annotation_name_gp = gpar(fontsize = 9),
                     col = list("Effect size" = effect_col_fun),
                     simple_anno_size = unit(0.35, "cm"),
                     annotation_legend_param = list(title_gp = gpar(fontsize = 9,
                                                                     fontface="bold"),
                                                     labels_gp = gpar(fontsize = 9),
                                                     direction ="vertical"),
                     show_legend = F, gp = gpar(col = "white"), show_annotation_name = T)

```

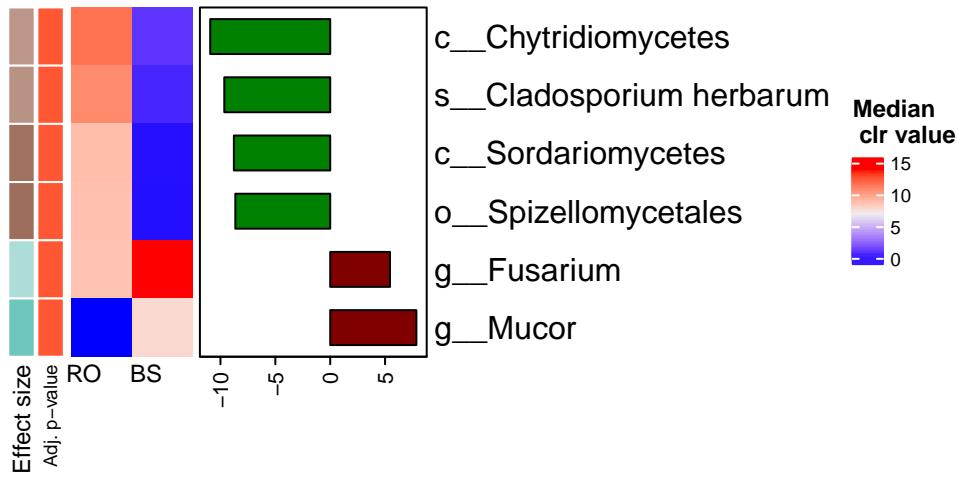
```

H1<-Heatmap(heat1, cluster_rows = F,
             cluster_columns = F, width = ncol(heat1)*unit(8, "mm"),
             height = nrow(heat1)*unit(7.7, "mm"), column_names_rot = 0,
             left_annotation = c(annE, annP),
             right_annotation=c(barpl),
             name = "Median \n clr value",
             heatmap_legend_param = list(direction = "vertical" ,
                                           labels_gp = gpar(fontsize = 7),
                                           title_gp = gpar(fontsize = 9, fontface="bold"),
                                           legend_height = unit(1.4, "cm")),
             column_names_gp = gpar(fontsize = 9),
             row_names_gp = gpar(fontsize = 12), show_heatmap_legend = T)

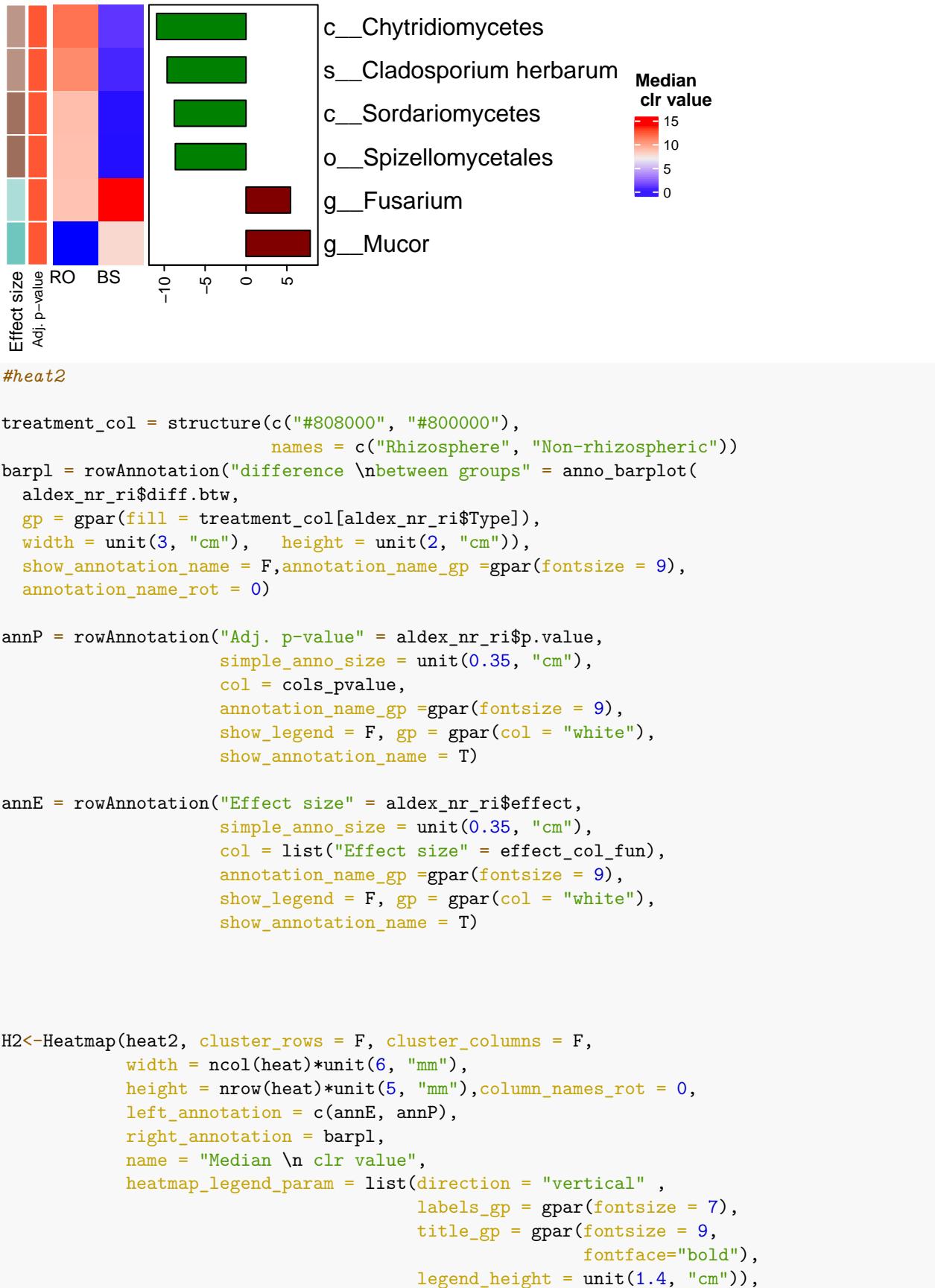
```

Warning: The input is a data frame, convert it to the matrix.

```
H1.1<- draw(H1, heatmap_legend_side = "right")
```



H1.1



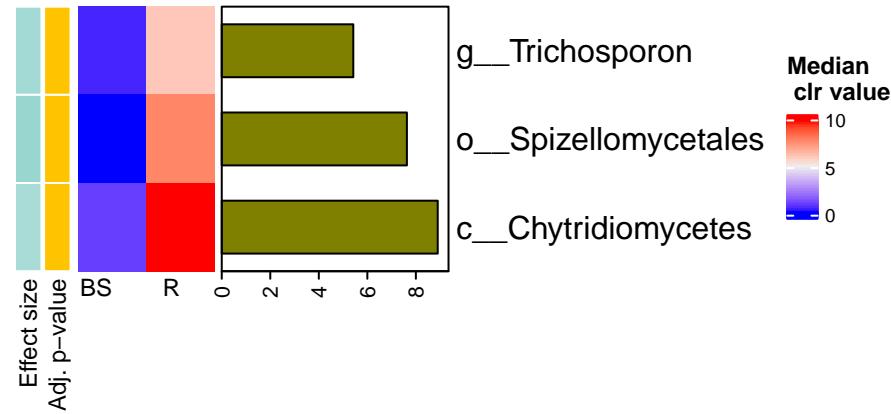
```

column_names_gp = gpar(fontsize = 9),
row_names_gp = gpar(fontsize = 12))

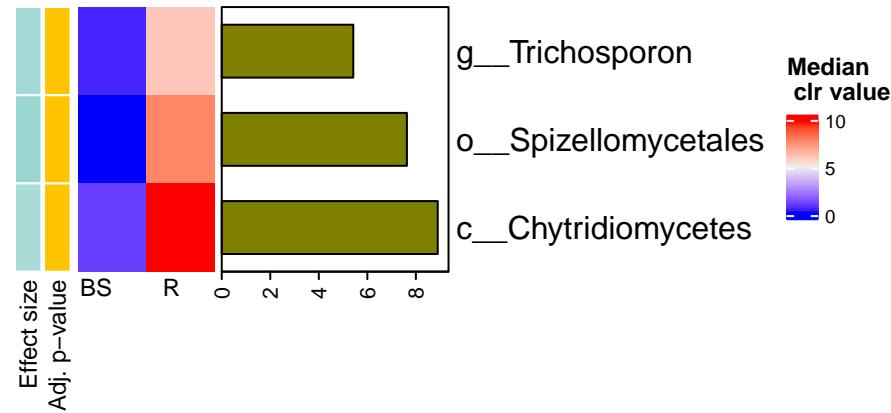
## Warning: The input is a data frame, convert it to the matrix.

H2.1<- draw(H2, heatmap_legend_side = "right")

```



H2.1



#heat3

```

treatment_col = structure(c("#008000", "#808000"),
                           names = c("Roots", "Rhizosphere"))
barpl = rowAnnotation("difference \nbetween groups" = anno_barplot(
  aldex_ro_ri$diff.btw,
  gp = gpar(fill = treatment_col[aldex_ro_ri$type]),
  width = unit(3.5, "cm"), height = unit(2, "cm")),
  show_annotation_name = F,
  annotation_name_gp = gpar(fontsize = 9),
  annotation_name_rot = 0)

annP = rowAnnotation("Adj. p-value" = aldex_ro_ri$p.value,
                     annotation_name_gp = gpar(fontsize = 9),
                     simple_anno_size = unit(0.35, "cm"),
                     annotation_legend_param = list(title_gp = gpar(fontsize = 9,
                                                                     fontface = "bold"),
                                                     labels_gp = gpar(fontsize = 9),
                                                     direction = "horizontal"),

```

```

    col = cols_pvalue, show_legend = T,
    gp = gpar(col = "white"), show_annotation_name = T)

annE = rowAnnotation("Effect size" = aldex_ro_ri$effect,
                     annotation_name_gp = gpar(fontsize = 9),
                     col = list("Effect size" = effect_col_fun),
                     simple_anno_size = unit(0.35, "cm"),
                     annotation_legend_param = list(title_gp = gpar(fontsize = 9,
                                                                     fontface="bold"),
                                                     labels_gp = gpar(fontsize = 9),
                                                     direction ="horizontal"),
                     show_legend = T, gp = gpar(col = "white"), show_annotation_name = T)

H3<-Heatmap(heat3, cluster_rows = F, cluster_columns = F,
             width = ncol(heat)*unit(5, "mm"),
             height = nrow(heat)*unit(4.5, "mm"), column_names_rot = 0,
             left_annotation = c(annE, annP), name = "Median \n clr value",
             right_annotation = barpl,
             heatmap_legend_param = list(direction = "vertical" , labels_gp = gpar(fontsize = 7),
                                           title_gp = gpar(fontsize = 9, fontface="bold"),
                                           legend_height = unit(1.4, "cm")),
             column_names_gp = gpar(fontsize = 9),
             row_names_gp = gpar(fontsize = 12)) %v% NULL

## Warning: The input is a data frame, convert it to the matrix.
H3.1<- draw(H3, annotation_legend_side = "top", heatmap_legend_side="right")

## Warning: 'legend_height' you specified is too small, use the default minimal
## height.

## Warning: 'legend_height' you specified is too small, use the default minimal
## height.

## Warning: 'legend_height' you specified is too small, use the default minimal
## height.




Effect size



|    |    |   |   |   |
|----|----|---|---|---|
| -4 | -2 | 0 | 2 | 4 |
|----|----|---|---|---|



Adj. p-value



|       |
|-------|
| <0.01 |
| <0.05 |



Median clr value



|     |
|-----|
| -20 |
| -10 |
| 0   |
| 10  |
| 20  |



Annotations:



- c__Sordariomycetes
- g__Trichosporon
- g__Fusarium
- g__Mucor



Effect size



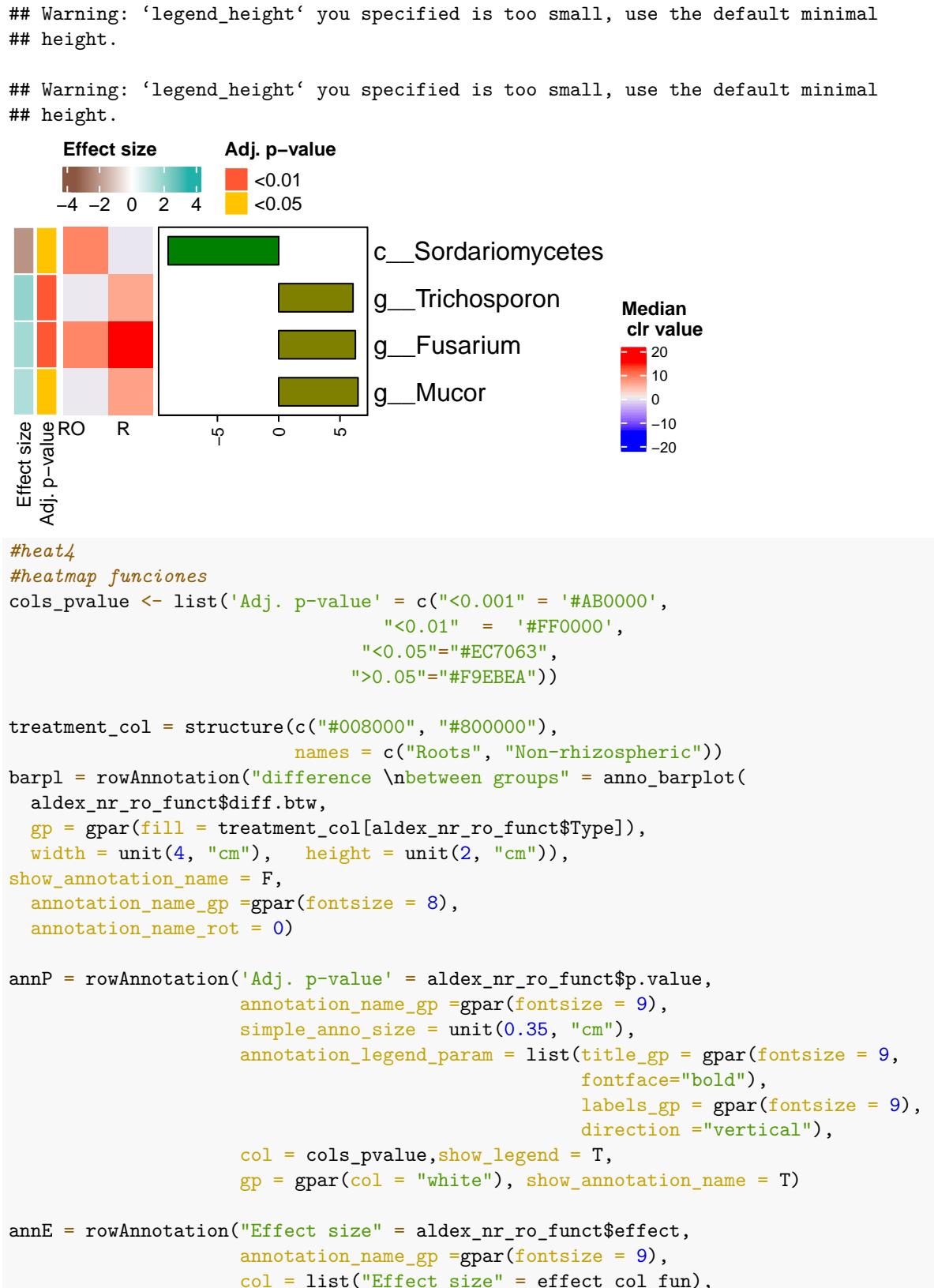
Adj. p-value



RO R


```

H3.1



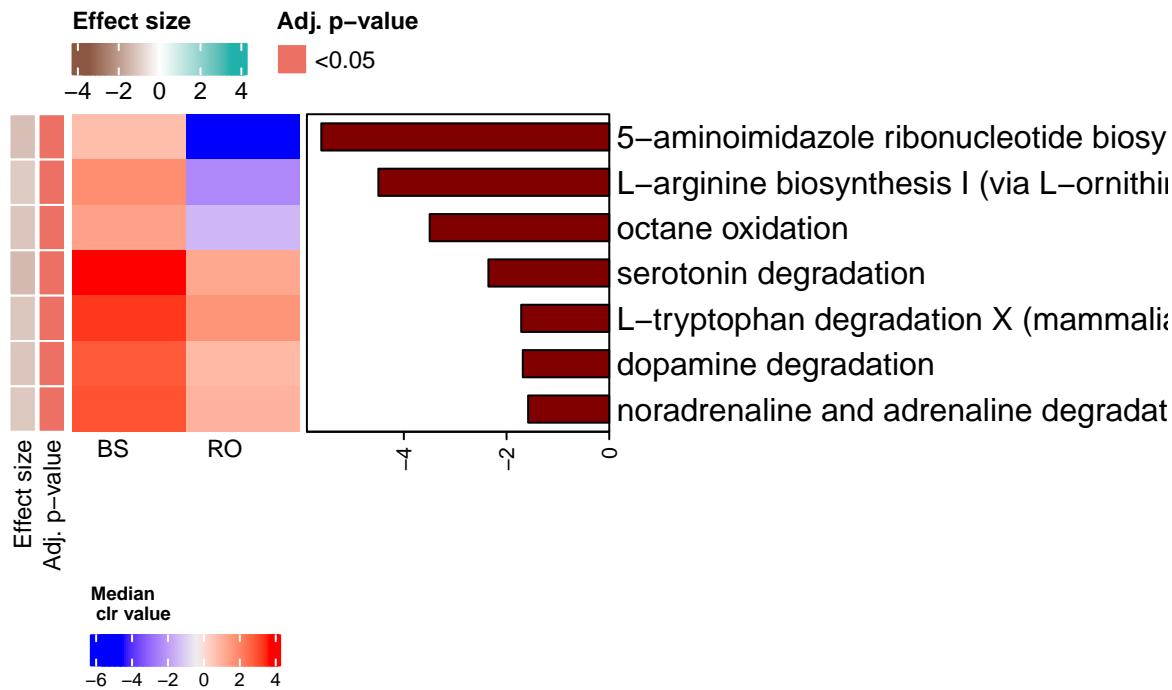
```

simple_anno_size = unit(0.35, "cm"),
annotation_legend_param = list(title_gp = gpar(fontsize = 9,
                                                fontface="bold"),
                                labels_gp = gpar(fontsize = 9),
                                direction ="horizontal"),
show_legend = T, gp = gpar(col = "white"), show_annotation_name = T)

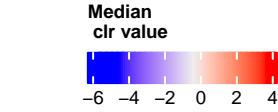
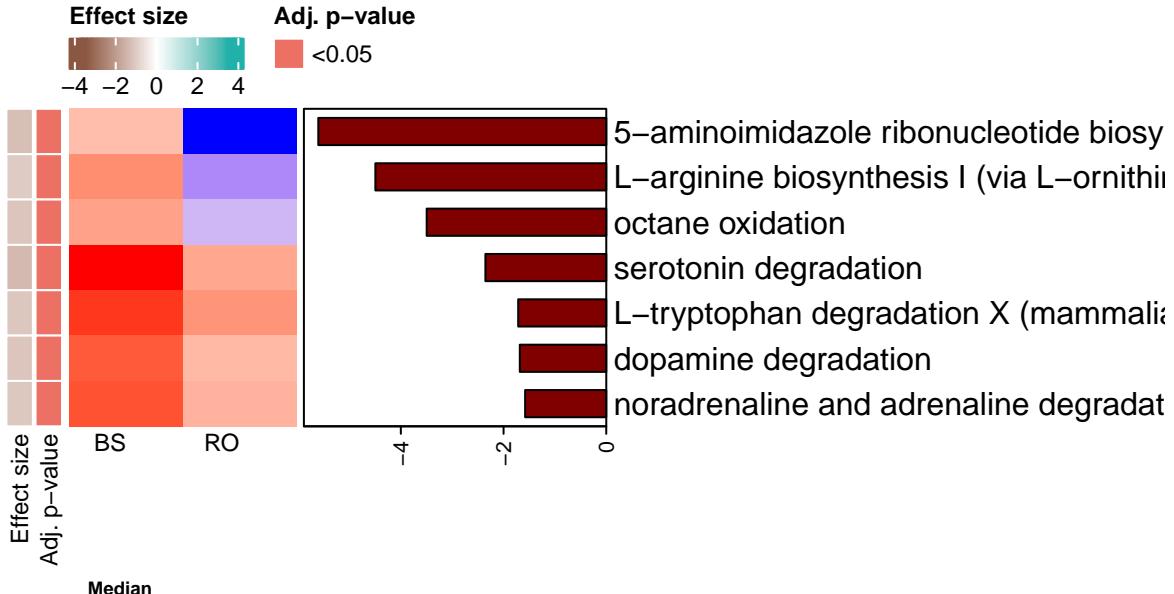
H4<-Heatmap(heat4, cluster_rows = F, cluster_columns = F, width = ncol(heat)*unit(10, "mm"),
height = nrow(heat)*unit(6, "mm"), column_names_rot = 0,
left_annotation = c(annE, annP), name = "Median \n clr value",
right_annotation = barpl,
heatmap_legend_param = list(direction = "horizontal" , labels_gp = gpar(fontsize = 7),
                               title_gp = gpar(fontsize = 7, fontface="bold"),
                               legend_height = unit(1.4, "cm")),
column_names_gp = gpar(fontsize = 9),
row_names_gp = gpar(fontsize = 12)) %v% NULL

## Warning: The input is a data frame, convert it to the matrix.
H4.1<- draw(H4, annotation_legend_side = "top", heatmap_legend_side="bottom")

```



H4.1



```
aldex_H1<- grid.grabExpr(draw(H1.1))
aldex_H2<- grid.grabExpr(draw(H2.1))
aldex_H3<- grid.grabExpr(draw(H3.1))
```

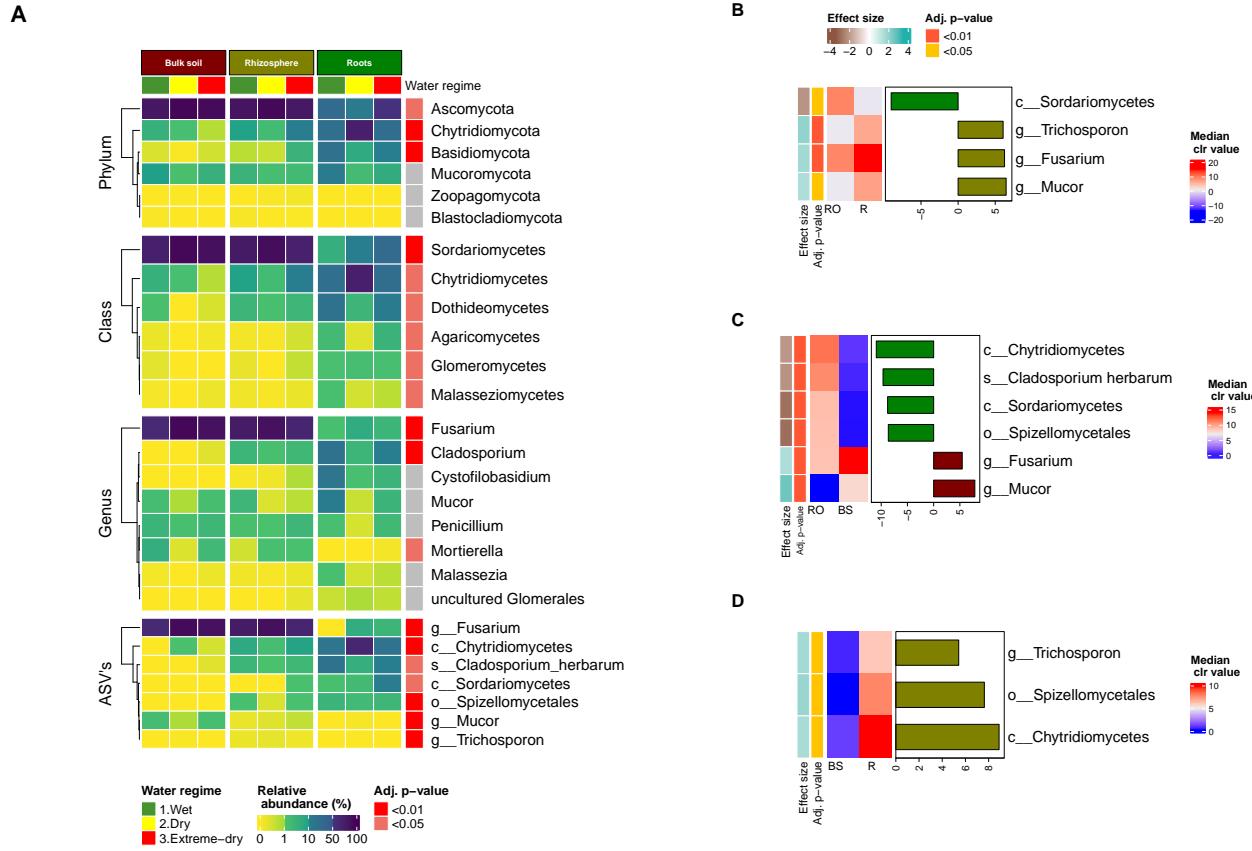
```
## Warning: 'legend_height' you specified is too small, use the default minimal
## height.
```

```
## Warning: 'legend_height' you specified is too small, use the default minimal
## height.
```

```
aldex_H4<- grid.grabExpr(draw(H4.1))
```

```
paired_grouped<-plot_grid(aldex_H3, aldex_H1, aldex_H2, nrow = 3,
                           rel_heights = c(1.1, 1, 1),
                           labels = c("B", "C", "D"))+ theme(
                             plot.margin = unit(c(0, 0, 0, 0), "cm"))
```

```
c<-cowplot:::plot_grid(heatm ,paired_grouped, ncol = 2, rel_widths = c(1.8,1.4), labels = c("A", NULL)
c
```



```
#ggsave(''./../Figures_final/Fig4.compartments_treatment_filt_nobarplot.pdf',
#      width = 12.5, height = 9, dpi = 300, plot =c)
#ggsave(''./../Figures_final/FigSX.Aldex_functional_mod.pdf',
#      width = 10, height = 6, dpi = 300, plot =aldex_H4)
```

```
#load libraries and files
library(imager)
library(cowplot)
library(ggpubr)
library(tidyverse)
library(magrittr)
library(ggpubr)
library(readxl)
library(viridis)
arbus2<-load.image(file = "../Figures/arbus_raw.png")
photo_panel <- ggdraw() + draw_image(arbus2, scale = 1)

#grafica arbusculos
```

```
dosmeses<- readxl::read_excel("../Data/MICOR ARBUS.xlsx", sheet = 2, range = "A1:O28")
cuatromeses<- readxl::read_excel("../Data/MICOR ARBUS.xlsx", sheet = 3, range = "A1:O29")
seismeses<- readxl::read_excel("../Data/MICOR ARBUS.xlsx", sheet = 4, range = "A1:O28")

arbus<- bind_rows(list(m2 = dosmeses, m4 = cuatromeses, m6=seismeses), .id = "Tiempos")
arbusc<- arbus %>% dplyr::select(L, P, "T", C, Tiempos, Porcentaje=%")
```

```

#ggblooxplot de todas

arbusc$Time <- factor(arbusc$Tiempos,
                      levels = c("0", "m2", "m4", "m6"),
                      labels = c("day0", "2 months", "4 months", "6 months"))
arbusc$Treatment <- factor(arbusc$T,
                           levels = c("1", "2", "3"),
                           labels = c("Wet", "Dry", "Extreme dry"))

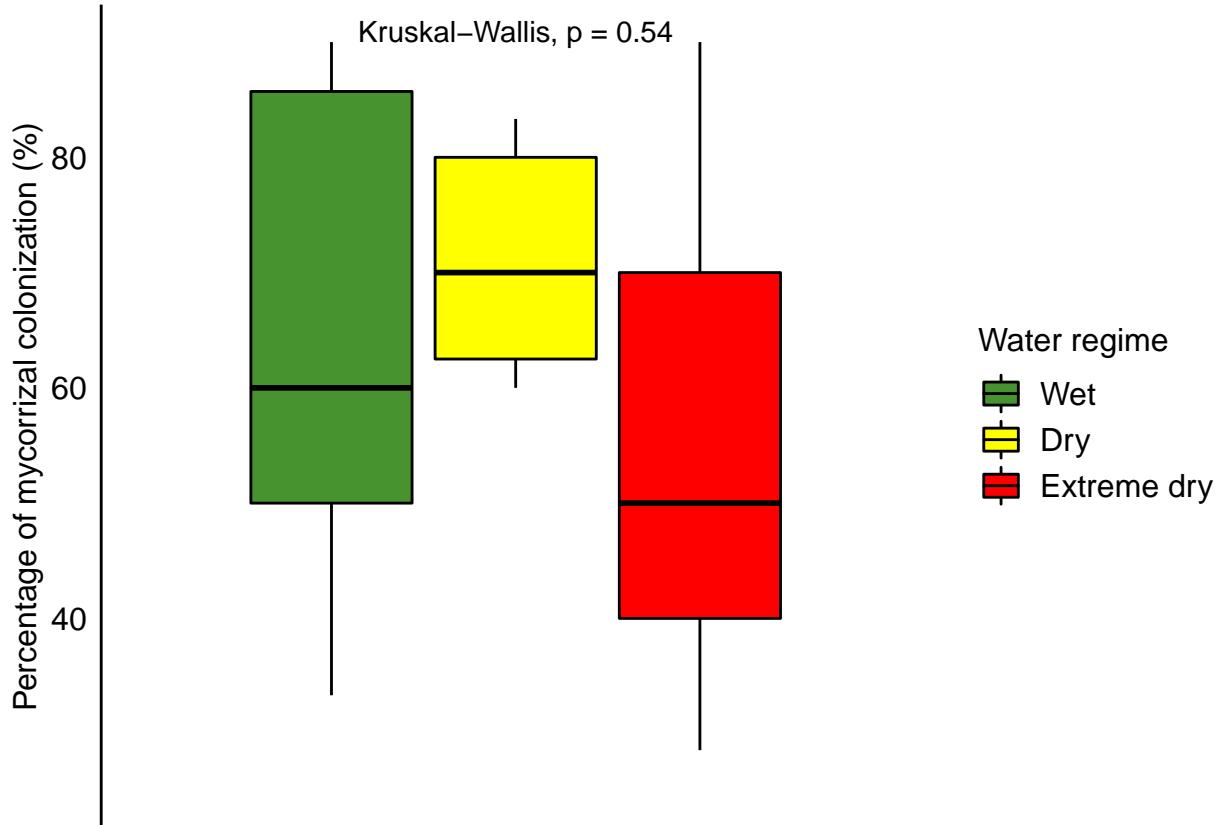
pallete = c("#479330", "#FFFF00", "#FF0000")

p<-arbusc %>% filter(Time == "2 months") %>% ggboxplot(
  ., x = "Time", y = "Porcentaje", #add = "mean_se",
  palette = pallete, color= "black", fill = "Treatment",outlier.shape = NA,
  # position = position_dodge(0.8) ,
  xlab = "Time",
  ylab = "Percentage of mycorrizal colonization (%)")

p1<-p + stat_compare_means(aes(group= Treatment))+
  # ,label= paste0(..method..,"\\n", "p=",..p.format..))+ 
  stat_compare_means(label.y = 130, show.legend = F )+ font("legend.title", size = 12)+ 
  font("legend.text", size = 12)+theme(axis.title.x = element_blank(),
                                       axis.text.x = element_blank(),
                                       axis.ticks = element_blank())+theme(legend.position = "right")+
  theme(plot.title = element_text(size = 14))

p1

```



```
#Guilds files

#Guild level
filo<- read_tsv("../Data/merge_table_240_noplant_filtered.guilds.txt") %>%
  rename( OTUID = "#OTU ID")
metadata<- read.csv("../Data/FINALMAP18S_2.csv") %>% dplyr::select(
  key=sampleid, groups, Treatment, Type_of_soil)

relabunda<- function(x){(t(t(x)/colSums(x)))*100}

filo2<- data.frame(filo[1], relabunda(filo[2:66]),
                     filo[67:76] , check.names = F)

guilds_gather<-filo2%>%dplyr::select(`1.2RO`:`2.10NR`, `Guild` )%>% gather(
  -`Guild`,key = "key", value = "value")

guilds_gather_sum<- guilds_gather %>% group_by(
  key,`Guild`)%>% summarise(prop =sum(value)) %>%
  mutate(Type = case_when(
    str_detect(key, "RI") ~ "Rhizosphere",
    str_detect(key, "US") ~ "Uncultivated",
    str_detect(key, "RO") ~ "Roots",
    str_detect(key, "SE") ~ "Initials",
    str_detect(key, "P") ~ "Initials",
    str_detect(key, "NR") ~ "Non-Rhizosphere")) %>% inner_join(metadata)
```

```

#Trophic level

trophic_gather<-filo2%>%dplyr::select(`1.2RO`:`7.23NR`, `Trophic Mode` )%>% gather(
  -`Trophic Mode`,key = "key", value = "value")

trophic_gather_sum<-trophic_gather %>% group_by(
  key, `Trophic Mode`)%>% summarise(prop =sum(value)) %>%
  mutate(Type = case_when(
    str_detect(key, "RI") ~ "Rhizosphere",
    str_detect(key,"US") ~ "Uncultivated",
    str_detect(key,"RO") ~ "Roots",
    str_detect(key,"SE") ~ "Initials",
    str_detect(key,"P") ~ "Initials",
    str_detect(key,"NR") ~ "Non-Rhizosphere"))%>% inner_join(metadata)

#growth level

growth_gather<-filo2%>%dplyr::select(`1.2RO`:`7.23NR`, `Growth Morphology` )%>% gather(
  -`Growth Morphology`,key = "key", value = "value")

growth_gather_sum<-growth_gather %>% group_by(
  key, `Growth Morphology`)%>% summarise(prop =sum(value)) %>%
  mutate(Type = case_when(
    str_detect(key, "RI") ~ "Rhizosphere",
    str_detect(key,"US") ~ "Uncultivated",
    str_detect(key,"RO") ~ "Roots",
    str_detect(key,"SE") ~ "Initials",
    str_detect(key,"P") ~ "Initials",
    str_detect(key,"NR") ~ "Non-Rhizosphere"))%>% inner_join(metadata)

```

Let's plot!

```

#colors plot
color_type<- c("#800000", "#808000", "#008000", "#D35400", "#2E4053")

guilds_gather_sum_join<- guilds_gather_sum  %>%
  rename(Level = Guild) %>% mutate(
  type = "Guild")%>%filter(
Level %in% c("Leaf Saprotroph" , "Arbuscular Mycorrhizal"))
trophic_gather_sum_join<-trophic_gather_sum %>%  rename(
  Level =`Trophic Mode` ) %>% mutate(
  type = "Trophic")%>%filter(
Level  %in% c("Saprotroph" , "Symbiotroph"))

gather_sum<- rbind(guilds_gather_sum_join, trophic_gather_sum_join) %>%
  mutate(Treatments = case_when(
    str_detect(Treatment, "1") ~ "Wet",
    str_detect(Treatment, "2") ~ "Dry",
    str_detect(Treatment, "3") ~ "Extreme-dry"))
gather_sum$Treatments<- factor(gather_sum$Treatments, levels = c(

```

```

    "Wet", "Dry", "Extreme-dry"))

#compartment effect
library(lme4)
library(nlme)
library(pgirmess)
meta<- read.delim("../Data/FINALMAP18S_plant.csv",
                  check.names = F) %>% dplyr::select("#SampleID", "Plant")

gather_sum_plant<- gather_sum %>% inner_join(meta, by = c("key"="#SampleID"))

guild_wet_mico<- gather_sum_plant %>% filter(type=="Guild") %>% filter(
  Treatments == "Wet") %>% filter(Level=="Arbuscular Mycorrhizal")
guild_wet_leafsap<- gather_sum_plant %>% filter(type=="Guild") %>% filter(
  Treatments == "Wet") %>% filter(Level=="Leaf Saprotoph")

guild_dry_mico<- gather_sum_plant %>% filter(type=="Guild") %>% filter(
  Treatments == "Dry") %>% filter(Level=="Arbuscular Mycorrhizal")
guild_dry_leafsap<- gather_sum_plant %>% filter(type=="Guild") %>% filter(
  Treatments == "Dry") %>% filter(Level=="Leaf Saprotoph")

guild_exdry_mico<- gather_sum_plant %>% filter(type=="Guild") %>% filter(
  Treatments == "Extreme-dry") %>% filter(Level=="Arbuscular Mycorrhizal")
guild_exdry_leafsap<- gather_sum_plant %>% filter(type=="Guild") %>% filter(
  Treatments == "Extreme-dry") %>% filter(Level=="Leaf Saprotoph")

guild_wet_mico1<-lme(prop~ Type_of_soil, random=~1 |Plant, data = guild_wet_mico)
guild_wet_mico2<-PermTest(guild_wet_mico1)
guild_wet_mico2

## 
## Monte-Carlo test
##
## Call:
## PermTest.lme(obj = guild_wet_mico1)
##
## Based on 1000 replicates
## Simulated p-value:
##           p.value
## (Intercept) 0.853
## Type_of_soil 0.233

guild_wet_leafsap1<-lme(prop~ Type_of_soil, random=~1 |Plant, data = guild_wet_leafsap)
guild_wet_leafsap2<-PermTest(guild_wet_leafsap1)

guild_dry_mico1<-lm(prop~ Type_of_soil, data = guild_dry_mico)
guild_dry_mico2<-PermTest(guild_dry_mico1)
guild_dry_leafsap1<-lm(prop~ Type_of_soil, data = guild_dry_leafsap)
guild_dry_leafsap2<-PermTest(guild_dry_leafsap1)

guild_exdry_mico1<-lme(prop~ Type_of_soil, random=~1 |Plant, data = guild_exdry_mico)
guild_exdry_mico2<-PermTest(guild_exdry_mico1)

```

```

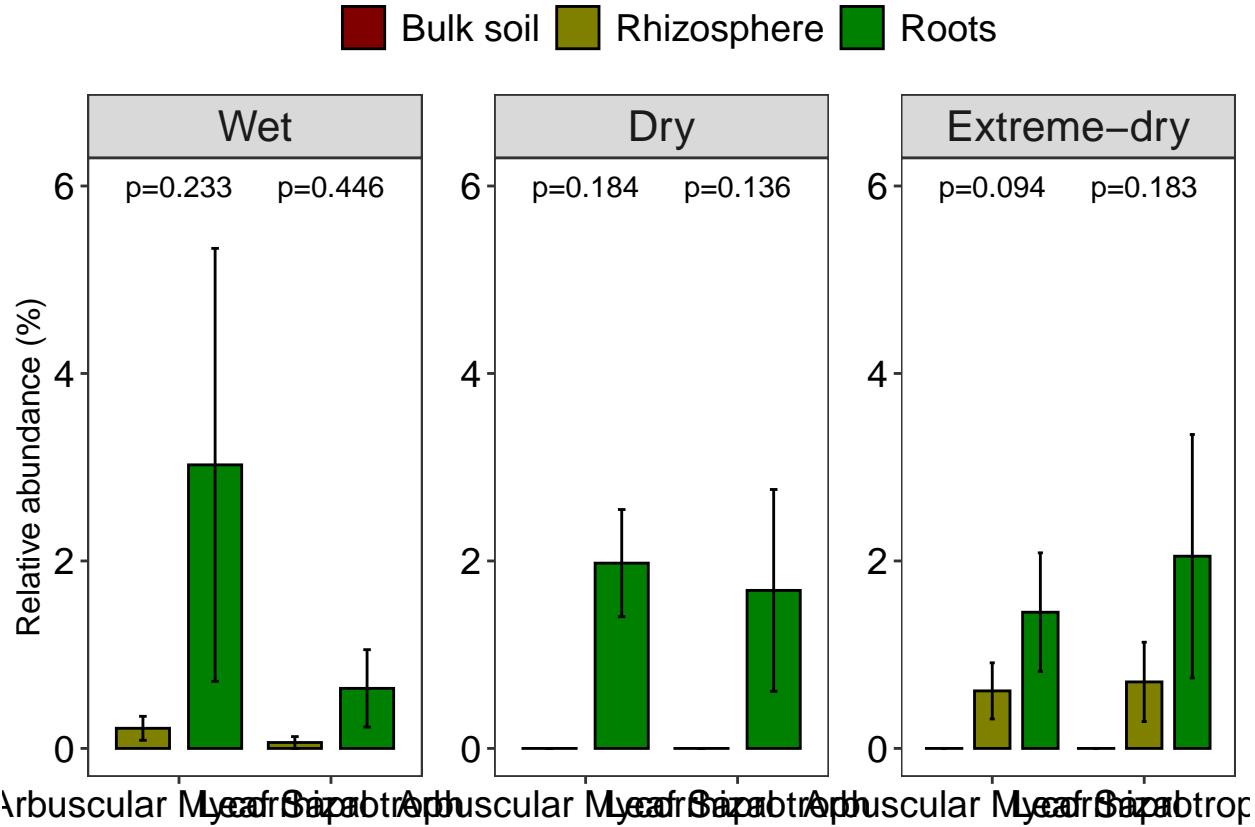
guild_exdry_leafsap1<-lme(prop~ Type_of_soil, random=~1 |Plant, data = guild_exdry_leafsap)
guild_exdry_leafsap2<-PermTest(guild_exdry_leafsap1)

annot_df_guild<- data.frame(
  Level= c("Arbuscular Mycorrhizal", "Leaf Saprotroph",
          "Arbuscular Mycorrhizal", "Leaf Saprotroph",
          "Arbuscular Mycorrhizal", "Leaf Saprotroph"),
  prop=c(6,6,6, 6, 6,6),
  Treatments=c("Wet", "Dry", "Extreme-dry"),
  label=c(
    paste0("p=",guild_wet_mico2$resultats$p.value[2]),
    paste0("p=",guild_wet_leafsap2$resultats$p.value[2]),
    paste0("p=",guild_dry_mico2$resultats$p.value[1]),
    paste0("p=",guild_dry_leafsap2$resultats$p.value[1]),
    paste0("p=",guild_exdry_mico2$resultats$p.value[2]),
    paste0("p=",guild_exdry_leafsap2$resultats$p.value[2])), check.names = F)

annot_df_guild$Treatments<- factor(annot_df_guild$Treatments,
                                      levels =c("Wet", "Dry", "Extreme-dry") )

phyloplot4 <- gather_sum %>% mutate(Type=case_when(
  Type == "Non-Rhizosphere"~"Bulk soil",
  TRUE ~ as.character(Type))
)%>% filter(!key=="6.13R0"&!key=="5.14R0") %>% filter(
  type=="Guild") %>%
  ggbarplot(x = "Level", y="prop", color = "black", fill = "Type",
             position = position_dodge(), add = "mean_se",
             facet.by = "Treatments")+
  facet_wrap(~Treatments, scales = "free", ncol=3, strip.position = "top")+
  theme_bw()+
  theme(panel.spacing=unit(1,"lines"),
        strip.text.x = element_text(size = 16),
        axis.text = element_text(colour = "black", size = 14),
        axis.title.x = element_blank(),
        axis.title.y = element_text(size = 12),
        legend.title = element_blank(),
        legend.text = element_text(size=14),
        #axis.text.x = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        legend.position = "top")+scale_fill_manual(values =color_type)+
  ylab("Relative abundance (%)")+geom_text(data = annot_df_guild,label=annot_df_guild$label)
phyloplot4

```



```
#compartment effect
library(lme4)
library(nlme)
library(pgirmess)
meta<- read.delim("../Data/FINALMAP18S_plant.csv",
  check.names = F) %>% dplyr::select("#SampleID", "Plant")
gather_sum_plant<- gather_sum %>% inner_join(meta, by = c("key"="#SampleID"))

guild_wet_mico<- gather_sum_plant %>% filter(type=="Trophic") %>% filter(
  Treatments == "Wet") %>% filter(Level=="Symbiotroph")
guild_wet_leafsap<- gather_sum_plant %>% filter(type=="Trophic") %>% filter(
  Treatments == "Wet") %>% filter(Level=="Saprotoph")

guild_dry_mico<- gather_sum_plant %>% filter(type=="Trophic") %>% filter(
  Treatments == "Dry") %>% filter(Level=="Symbiotroph")
guild_dry_leafsap<- gather_sum_plant %>% filter(type=="Trophic") %>% filter(
  Treatments == "Dry") %>% filter(Level=="Saprotoph")

guild_exdry_mico<- gather_sum_plant %>% filter(type=="Trophic") %>% filter(
  Treatments == "Extreme-dry") %>% filter(Level=="Symbiotroph")
guild_exdry_leafsap<- gather_sum_plant %>% filter(type=="Trophic") %>% filter(
  Treatments == "Extreme-dry") %>% filter(Level=="Saprotoph")

guild_wet_mico1<-lme(prop~ Type_of_soil, random=~1 |Plant, data = guild_wet_mico)
guild_wet_mico2<-PermTest(guild_wet_mico1)
guild_wet_mico2
```

```

##  

## Monte-Carlo test  

##  

## Call:  

## PermTest.lme(obj = guild_wet_mico1)  

##  

## Based on 1000 replicates  

## Simulated p-value:  

##           p.value  

## (Intercept) 0.583  

## Type_of_soil 0.206  

guild_wet_leafsap1<-lme(prop~ Type_of_soil, random=~1 |Plant, data = guild_wet_leafsap)
guild_wet_leafsap2<-PermTest(guild_wet_leafsap1)

guild_dry_mico1<-lm(prop~ Type_of_soil, data = guild_dry_mico)
guild_dry_mico2<-PermTest(guild_dry_mico1)

guild_dry_leafsap1<-lm(prop~ Type_of_soil, data = guild_dry_leafsap)
guild_dry_leafsap2<-PermTest(guild_dry_leafsap1)

guild_exdry_mico1<-lme(prop~ Type_of_soil, random=~1 |Plant, data = guild_exdry_mico)

guild_exdry_mico2<-PermTest(guild_exdry_mico1)
guild_exdry_leafsap1<-lme(prop~ Type_of_soil, random=~1 |Plant, data = guild_exdry_leafsap)
guild_exdry_leafsap2<-PermTest(guild_exdry_leafsap1)

annot_df_guild<- data.frame(  

  Level= c("Symbiotroph", "Saprotoph", "Symbiotroph",  

          "Saprotoph", "Symbiotroph", "Saprotoph"),  

  prop=c(6,6,6, 6, 6,6),  

  Treatments=c("Wet", "Dry", "Extreme-dry"),  

  label=c(  

  paste0("p=",guild_wet_mico2$resultats$p.value[2]),  

  paste0("p=",guild_wet_leafsap2$resultats$p.value[2]),  

  paste0("p=",guild_dry_mico2$resultats$p.value[1]),  

  paste0("p=",guild_dry_leafsap2$resultats$p.value[1]),  

  paste0("p=",guild_exdry_mico2$resultats$p.value[2]),  

  paste0("p=",guild_exdry_leafsap2$resultats$p.value[2])), check.names = F)  

annot_df_guild$Treatments<- factor(annotation_df_guild$Treatments,  

                                         levels =c("Wet", "Dry", "Extreme-dry") )  

phyloplot5 <- gather_sum %>% mutate(Type=case_when(  

  Type == "Non-Rhizosphere" ~ "Bulk soil",  

  TRUE ~ as.character(Type)) %>% filter(!key=="6.13R0" & !key=="5.14R0") %>% filter(type=="Trophic") %>%  

  ggbarplot(x = "Level", y="prop", color = "black", fill = "Type",  

            position = position_dodge(), add = "mean_se",  

            facet_by = "Treatments") +  

  facet_wrap(~Treatments, scales = "free",  

            ncol=3, strip.position = "top") +  

  theme_bw() +

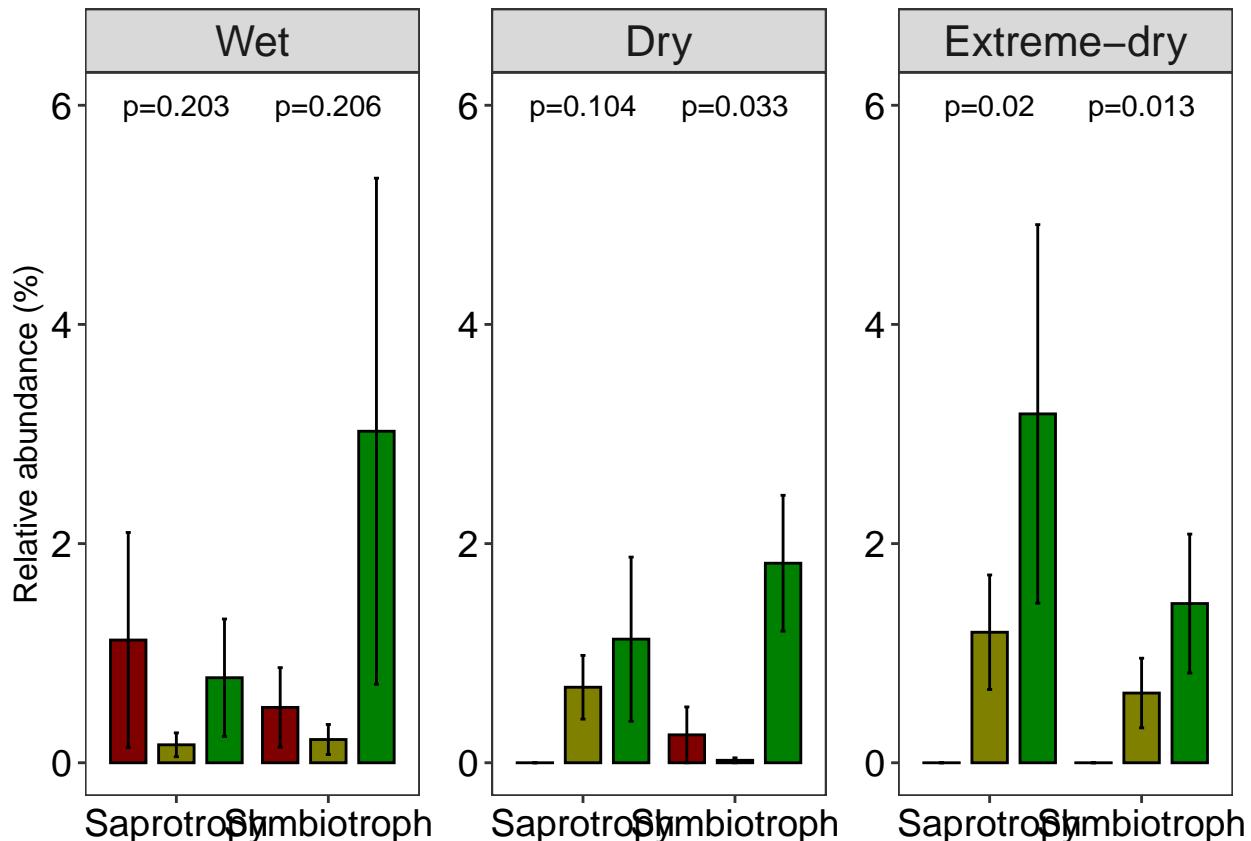
```

```

theme(panel.spacing=unit(1,"lines"),
      strip.text.x = element_text(size = 16),
      axis.text = element_text(colour = "black", size = 14),
      axis.title.x = element_blank(),
      axis.title.y = element_text(size = 12),
      legend.title = element_blank(),
      legend.text = element_text(size=14),
      panel.grid.major = element_blank(),
      panel.grid.minor = element_blank(),
      legend.position = "top")+
scale_fill_manual(values =color_type)+
ylab("Relative abundance (%)")+
geom_text(data = annot_df_guild,label=annot_df_guild$label)+
```

theme(legend.position = "none")

phyloplot5

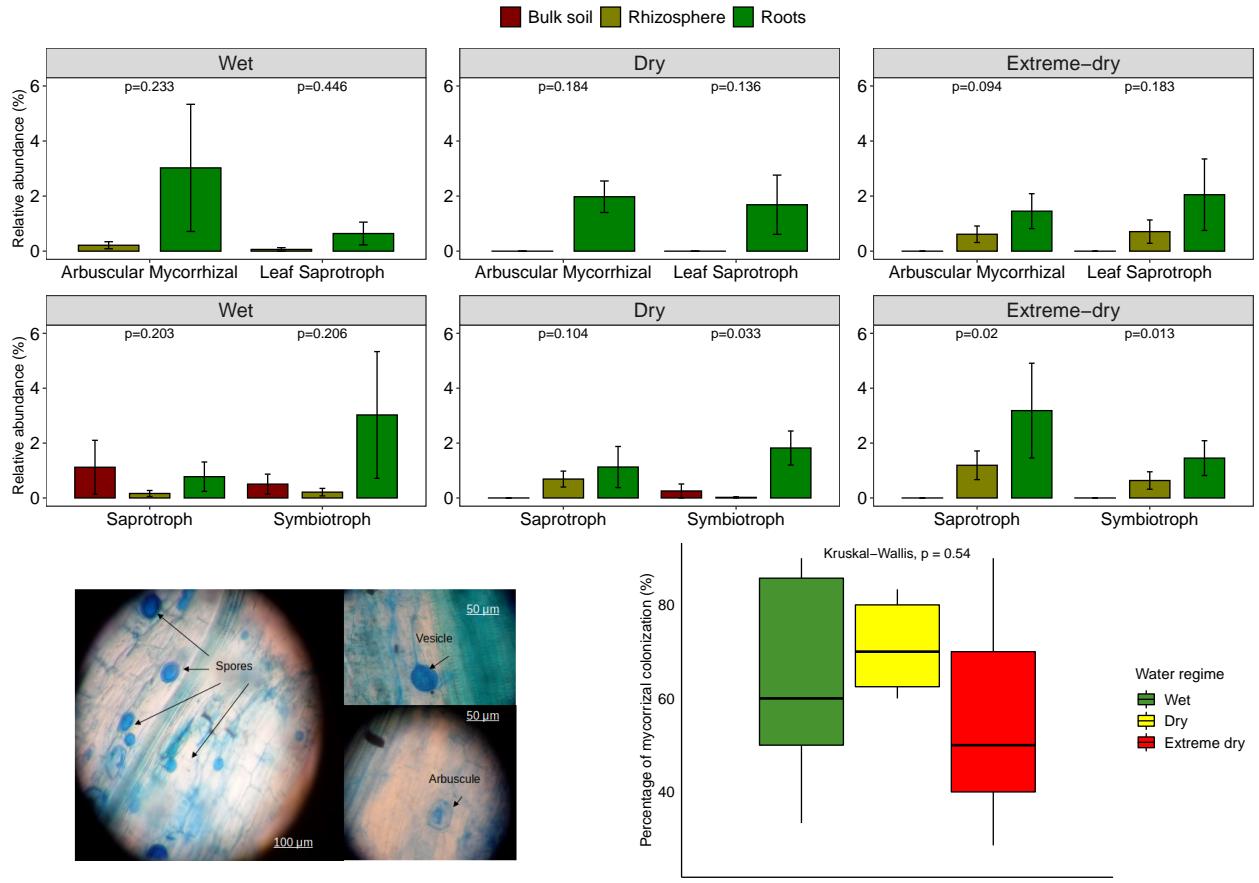


```

library(cowplot)
a1<-plot_grid(photo_panel, p1)
```

```

## Warning: Computation failed in 'stat_compare_means()':
## argument "x" is missing, with no default
a2<- plot_grid(phyloplot4,phyloplot5,nrow = 2, rel_heights = c(1.2,1))
b1<- plot_grid( a2,a1 ,nrow = 2, rel_heights = c(2,1.3))
b1
```



```
#ggsave(''./../Figures_final//Fig5.guilds_arbusc_modif.pdf',
width = 14, height = 10, dpi = 300, plot = b1)
```

```
#ARBUSCLES IMAGE
library(imager)
library(cowplot)
library(viridis)
library(tidyverse)
library(ggpubr)
library(tidyverse)
library(qiime2R)
library(circlize)
library(viridis)
library(ggpubr)
library(ComplexHeatmap)

metadata2<- read_tsv("../Data/meta_dos.txt")
otutable<- read_qza("../Data/merge_table_240_noplant_filtered_nous.qza")$data
taxonomy<- read_qza("../Data/taxonomy_blast_240_0.97.qza")$data

parse<- qiime2R::parse_taxonomy(taxonomy)
phyl<- qiime2R::summarize_taxa(features = otutable, taxonomy = parse)$Genus
phy.ra <- function(x){(t(x)/colSums(x))}

phyl_ro<- phyl %>% dplyr::select_at(vars(contains("RO")))) %>% filter_all(
., any_vars(. != 0)) %>% phy.ra
```

```

.) %>% as.data.frame() %>% rownames_to_column(var = "SampleID")

phy_met<- phyl_ro %>% inner_join(metadata2) %>% column_to_rownames(var = "SampleID") %>% dplyr::select(
  contains("d__"), hojas, raices, Arbus_per)

library(Hmisc)
cors<- phy_met
corr <- rcorr(as.matrix(cors), type=c("spearman"))
#print(corr)
cor.out <-corr$r
#write.table (cor.out, "yenCorrgen", sep="\t")

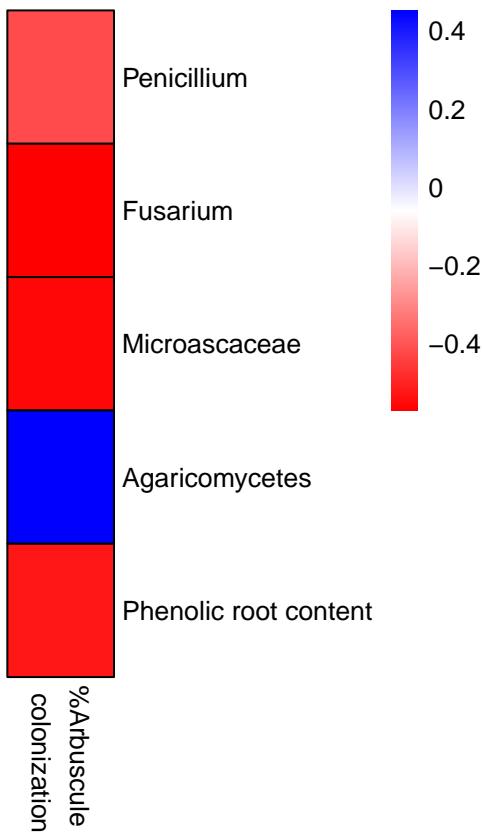
cor.plot<- cor.out %>% as.data.frame() %>% dplyr::select(Arbus_per) %>% filter(abs(Arbus_per)>=abs(0.4))

cor_heat<-cor.plot%>% rownames_to_column(var="id") %>% mutate_at(
  "id", str_replace, "; NA", "")%>% mutate_at(
  "id", str_replace, "; NA", "")%>% mutate_at(
  "id", str_replace, "; NA", "")%>% mutate(tax= str_extract(id, "[^; ]+$")) %>% mutate(
  tax= case_when(
    tax=="raices"~ "Phenolic root content",
    TRUE~as.character(tax))) %>% filter(!id=="Arbus_per") %>%
dplyr::select(tax, "%Arbuscule \n colonization"=Arbus_per) %>%
column_to_rownames(var = "tax")

test_labels<- cor_heat[1]

my_palette <- colorRampPalette(c("red", "white","blue"))(n=599)
library(pheatmap)
heats<-pheatmap(cor_heat, color = my_palette,legend = TRUE,cellheight = 50,
                 cellwidth = 40,border_color = "black",
                 cluster_cols = F, cluster_rows = F, fontsize = 10)

```



```

heat<- grid.grabExpr(heats)

otu_grouped<- read.delim("/home/steph/Documents/Documentos/fastas nuevos/18S/demultiplexed/demux/table_paired_end.fastq.gz") %>% skip = 1, check.names = F) %>% column_to_rownames(var = "#OTU ID") %>% dplyr::select(-taxonomy)

taxo<- read_qza("/home/steph/Documents/Documentos/fastas nuevos/18S//demultiplexed/demux/seqs_and_taxonomy.qza")
phyra<-t(otu_grouped)/colSums(otu_grouped) *100

metadata<- read_tsv("/home/steph/Documents/Documentos/fastas nuevos/18S//MAPPINGS/FINALMAP_GROUPED.txt")

## Rows: 41 Columns: 27
## -- Column specification -----
## Delimiter: "\t"
## chr  (3): #SampleID, Type_of_soil, Treatments
## dbl (23): T, Loc, FW, Root_L, Stem_L, Root_DW, Treatment, TOC, Root_FW, Leaf_Mass, Leaf_N, Leaf_P, Leaf_K, Leaf_C, Leaf_H, Leaf_A, Leaf_S, Leaf_Mg, Leaf_Cl, Leaf_B, Leaf_Zn, Leaf_Mn, Leaf_Fe, Leaf_Mo
## lgl   (1): Month
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
otu_grouped_relab<- phyra %>% t() %>% as.data.frame() %>% rownames_to_column(
  var = "Feature.ID") %>% inner_join(taxonomy) %>% dplyr::select(-Feature.ID, -Consensus)

## Joining, by = "Feature.ID"
otu_glm<-otu_grouped_relab %>%
  filter(str_detect(Taxon, 'Glom')) %>% rownames_to_column(var = "rows") %>% unite(
  
```

```

names, c(rows, Taxon), sep = " ") %>% column_to_rownames(
  var = "names") %>% t() %>% as.data.frame(
) %>% rownames_to_column(var = "#SampleID") %>% inner_join(metadata)

## Joining, by = "#SampleID"

rel<-otu_glm %>% group_by(Type_of_soil, Treatments) %>% summarise_if(
  is.numeric, mean) %>% dplyr::select(
  Type_of_soil,Treatments,contains("d_")) %>% unite(
  "ids", Type_of_soil:Treatments) %>% column_to_rownames(var = "ids") %>%
t() %>% as.data.frame() %>% rownames_to_column(var = "taxa") %>% mutate(
  tax= str_extract(taxa, "[^_]+$")) %>% mutate(
  taxon = case_when(
  tax=="sp."~"s_Glomeromycotina",
  tax=="uncultured"~"g_uncultured_Glomerales",
  tax=="Glomeraceae"~"f_Glomeraceae",
  tax=="Rhizophagus"~"g_Rhizophagus",
  tax=="Glomeromycetes"~"c_Glomeromycetes",
    TRUE ~ as.character(tax))) %>% filter(!taxon=="Plectosphaerellaceae")

relab<-otu_glm %>% dplyr::select("#SampleID",contains("d_")) %>% column_to_rownames(
  var = "#SampleID") %>%
t() %>% as.data.frame() %>% rownames_to_column(var = "taxa") %>% mutate(
  tax= str_extract(
  taxa, "[^_]+$")) %>% mutate(
  taxon = case_when(
  tax=="sp."~"s_Glomeromycotina",
  tax=="uncultured"~"g_uncultured_Glomerales",
  tax=="Glomeraceae"~"f_Glomeraceae",
  tax=="Rhizophagus"~"g_Rhizophagus",
  tax=="Glomeromycetes"~"c_Glomeromycetes",
    TRUE ~ as.character(tax))) %>% filter(
    !taxon=="Plectosphaerellaceae")
rel_sum<- rel %>% group_by(taxon) %>% summarise_if(
  is.numeric, sum) %>% column_to_rownames(
  var = "taxon")

target <- c("f_Glomeraceae","c_Glomeromycetes",
  "g_Rhizophagus" , "g_uncultured_Glomerales","s_Glomeromycotina")
rel_sum2<-rel_sum %>%rownames_to_column(var = "name") %>% arrange(
  factor(name, levels = target)) %>% column_to_rownames(var = "name")

#other heatmap
col_fun2 = colorRamp2(c(0, 0.5, 0.5+1e-5, 1,1.5), viridis(5, option = "D", direction = -1))
split = rep(1:3, each = 3)
treats<- c("1.Wet", "2.Dry", "3.Extreme-dry")
cols_ho<- list("Water regime" = c("1.Wet" = '#479330',
                                     "2.Dry" = '#FFFF00',
                                     "3.Extreme-dry"="#FF0000"))
ha = HeatmapAnnotation(foo = anno_block(gp = gpar(
  fill = c("#800000" , "#808000" , "#008000", "#D35400")),
  labels = c("Bulk soil", "Rhizosphere", "Roots"),
  labels_gp = gpar(col = "white", fontsize = 7, fontface= "bold")))

```

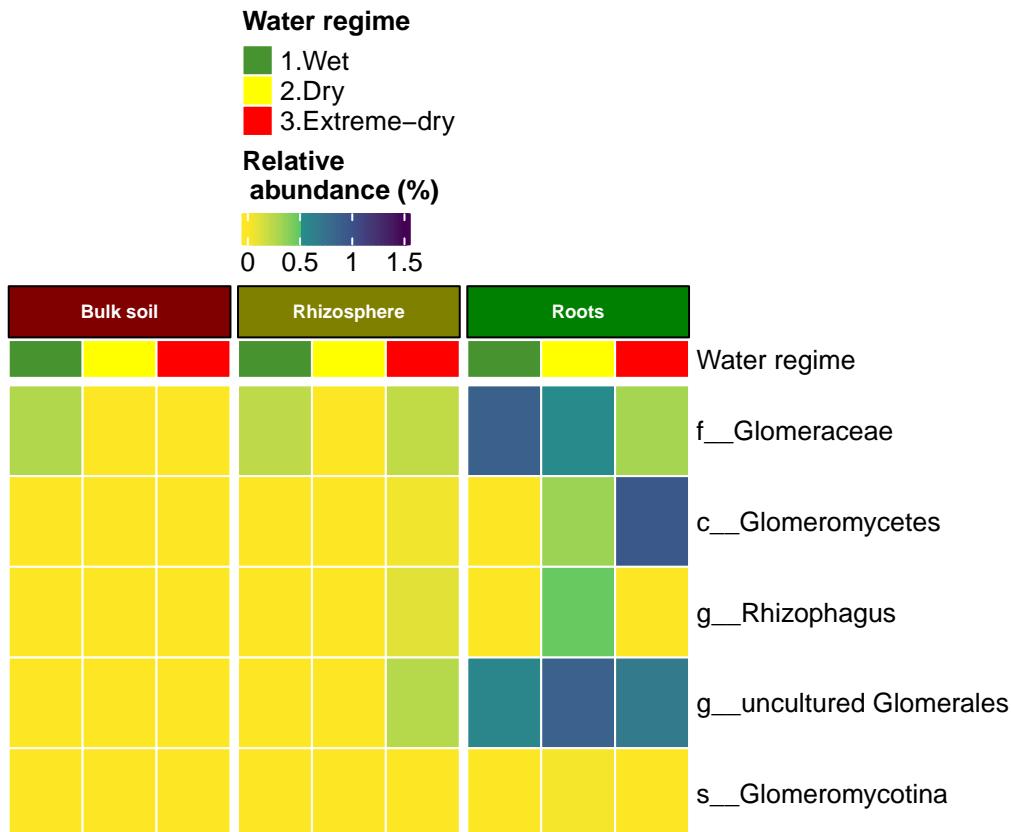
```

ho = HeatmapAnnotation("Water regime" = c(rep(treats, 3)),
                      which = "col", col = cols_ho,
                      annotation_name_gp = gpar(fontsize=10),
                      show_legend = T, gp = gpar(col = "white",
                                                 fontize=12),
                      show_annotation_name = T)

ht<-Heatmap(rel_sum2, col = col_fun2,
            heatmap_legend_param = list(direction = "horizontal",
                                         col_fun = col_fun2, title = "Relative \n abundance (%)",
                                         at = c(0,0.5, 1, 1.5), break_dist = 1, rect_gp = gpar(col = "white", lwd = 1),
                                         top_annotation = c(ha, ho),column_split = split,
                                         cluster_columns = F, cluster_rows = F,show_heatmap_legend = TRUE,
                                         show_column_names = FALSE,
                                         width = ncol(rel_sum2)*unit(10, "mm"),
                                         height = nrow(rel_sum2)*unit(12, "mm"),
                                         column_title = NULL, row_names_gp = gpar(fontsize=10))+
guides(fill=guide_legend(title="Water regime"))

## Warning: The input is a data frame, convert it to the matrix.
abund<-draw(ht, heatmap_legend_side = "top", annotation_legend_side = "top")

```



```

abunds<- grid.grabExpr(draw(abund))

one<-plot_grid(abunds, heat, rel_widths = c(1.5,1), labels = c("A", "B"))

```

```

#ggsave('..../Figures_final/Fig3.cor_glomus.pdf',
#"      width = 10, height = 6, dpi = 300, plot =one)

#PHENOLIC CONTENT
library(tidyverse)
library(ggpubr)
library(viridis)
library(cowplot)
pallete = c("#479330", "#FFFF00", "#FF0000")

fenoles<- readxl::read_excel("../Data/FENOLES_FINAL.xlsx", sheet = 1, range = "A1:G82")
fenoles$Time <- factor(fenoles$tiempo,
                       levels = c("0", "2", "4", "6"),
                       labels = c("day0", "2 months", "4 months", "6 months"))
fenoles$Treatment <- factor(fenoles$T,
                            levels = c("1", "2", "3"),
                            labels = c("Wet", "Dry", "Extreme dry"))

q<- fenoles %>% filter(Time == "2 months") %>% ggboxplot(
., x = "Treatment", y = "hojas", outlier.shape = NA,
palette = pallete, color= "black", fill = "Treatment",
xlab = "Time" ,
ylab = "mg GAE / g dry of leaves")

q1<-q + stat_compare_means(aes(group= Treatment), label.y = 85)+
  font("legend.title", size = 8) +
  font("legend.text", size = 8) + theme(
    axis.title.x = element_blank(),
    legend.position = "none",
    axis.ticks = element_blank())

r<- fenoles %>% filter(Time == "2 months") %>%
  ggboxplot(
., x = "Treatment", y = "raices",
palette = pallete, color= "black", fill = "Treatment",
legend="none",outlier.shape = NA,
xlab = "Time", ylab = "mg GAE / g dry of roots")
r1<-r + stat_compare_means(aes(group= Treatment), label.y = 30) +
  theme(
    axis.title.x = element_blank(),
    axis.ticks = element_blank())+scale_y_continuous(limits = c(0,35))

pallete2<- viridis_pal()(3)

fenoles2<- fenoles %>% gather(raices:hojas ,key = "parte", value = "fenoles")
colnames(fenoles2)

## [1] "L"          "P"          "T"          "C"          "tiempo"     "Time"

```

```

## [7] "Treatment" "parte"      "fenoles"
head(fenoles2)

## # A tibble: 6 x 9
##   L     P     T     C tiempo Time Treatment parte fenoles
##   <dbl> <dbl> <dbl> <dbl> <dbl> <fct>   <fct>   <chr>    <dbl>
## 1    7     1     1     1     2 2 months Wet      raices   9.24
## 2    4     3     1     1     2 2 months Wet      raices   21.8
## 3    6     1     2     1     6 6 months Dry     raices   31.6
## 4    1     2     2     1     4 4 months Dry     raices   NA
## 5    2     2     2     1     6 6 months Dry     raices   30.3
## 6    1     2     3     2     2 2 months Extreme dry raices   9.80

fenoles2$Part <- factor(fenoles2$parte,
                         levels = c("raices", "hojas"),
                         labels = c("Roots", "Leaves"))

s<-fenoles2 %>% filter(Time == "2 months") %>% ggboxplot(
  ., x = "Part", y = "fenoles", outlier.shape = NA,
  palette = pallete2, color= "black", fill = "Part",
  xlab = "Part of the plant",
  ylab = "mg GAE/ g dry weight")

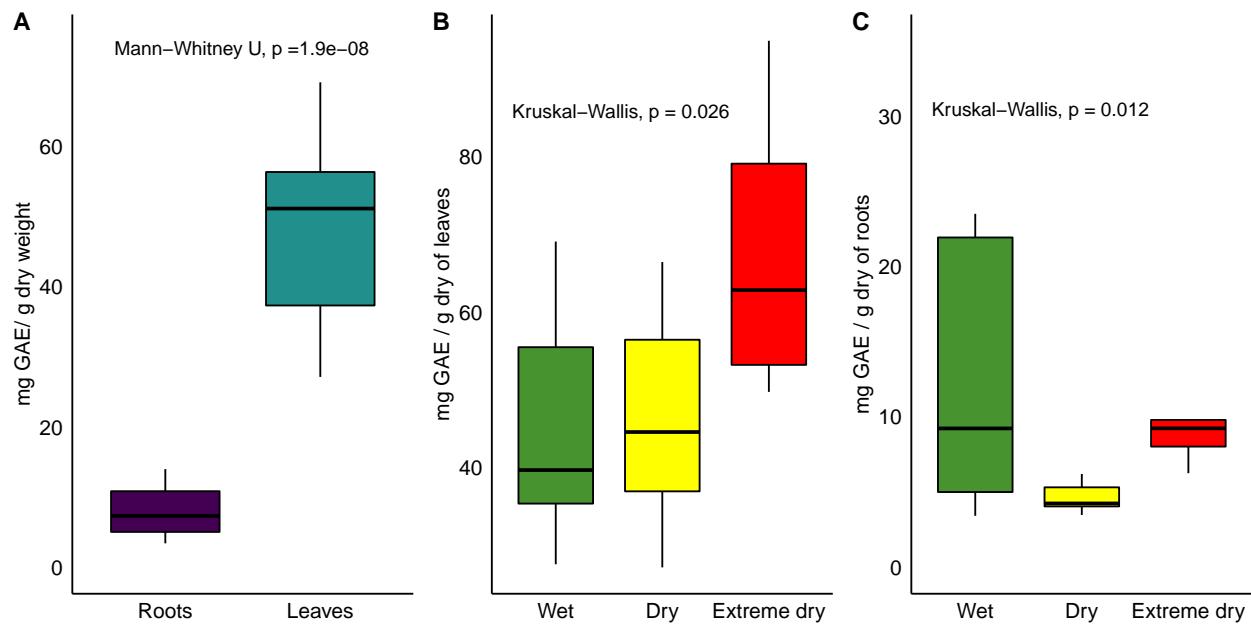
s1<-s + stat_compare_means(label.y = 73, show.legend = F, aes(
  label = paste0("Mann-Whitney U, p =", ..p.format..))+ 
  font("legend.title", size = 9)+ 
  font("legend.text", size = 9) +
  theme(legend.key.height= unit(4, 'mm'), legend.title = element_blank(),
        legend.key.width= unit(4, 'mm'), legend.position = "none")+
  theme( axis.title.x = element_blank(),
        axis.ticks = element_blank())+
  scale_y_continuous(limits = c(0,75))

#joining plots

third<- plot_grid(s1, q1,
                   r1, ncol = 3, align = "vh",axis = "l",
                   labels = c( "A","B", "C"))

third

```



```
#ggsave('..../Figures_final/Fig6.Phenolic_content_mod.pdf', width = 10, height = 5, dpi = 300, plot =
```