Concentration Trial

Maximilian Fernando Becker

2023-03-30

Bacterial community analysis

Becker *et al.* "Plant health protecting product application"

This document contains all statistical analyses conducted for the manuscript. Note that due to the random iterative nature of some analyses (such as PERMANOVA, CAP & PCA) some of the figure parameters will change slightly during reanalysis, though core results will remain essentially unchanged.

All data to reproduce analysis can be found here: `https://github.com/mfbeuq/becker_etal_PHPPapplication`

Load necessary ecological analysis libraries.

```
library(phyloseq)
library(microbiome)
library(ggord)
library(metagMisc)
library(ggpubr)
library(FSA)
library(knitr)
library(rmarkdown)
library(ape)
library(vegan)
library(philr)
library(compositions)
library(qiime2R)
library(plyr)
library(dplyr)
library(tidyr)
library(PMCMR)
library(tibble)
library(viridis)
library(gridExtra)
library(AcidPlots)
library(grid)
library(colorRamps)
```

```
library(rstatix)
library(dunn.test)
library(pairwiseAdonis)
library(dplyr)
library(ANCOMBC)
library(nlme)
library(tidyverse)
library(compositions)
library(readr)
library(DT)
library(matrixStats)
library(pheatmap)
library(RColorBrewer)
library(dendsort)
library(ComplexHeatmap)
library(circlize)
library(round)
library(lme4)
library(emmeans)
set.seed(225)
```

Contents of this workspace

```
load('concentration_trial.RData')
```

- `ps`: phyloseq object including the metadata, and the ASV table, the taxonomy table and the phylogenetic tree, all three created by QIIME2
- `psL.meta`: Shannon diversity index data of the loosely associated (L) root microbiota sub data set
- `psT.meta`: Shannon diversity index data of the tightly associated (T) root microbiota sub data set
- `RA.ord.L`: DEICODE ordination generated by QIIME2 of the loosely associated (L) root microbiota sub data set
- `RA.ord.T`: DEICODE ordination generated by QIIME2 of the loosely associated (T) root microbiota sub data set
- `RA.dist.L`: DEICODE distance matrix generated by QIIME2 of the loosely associated (L) root microbiota sub data set
- `RA.dist.T`: DEICODE distance matrix generated by QIIME2 of the tightly associated (T) root microbiota sub data set
- `q_meta`: the metadata table
- `deicode_theme`: a theme for generating the plots
- `plot_theme`: a theme for generating the plots
- `p.colors`: a color list for phyla coloring
- `read_distance`: function for importing a DEICODE distance matrix generated by QIIME2
- 'rename.ancombc.output": a function for renaming taxa in an ANCOM-BC table

and various lists for renaming factors

Alpha diversity statistics:

The following does the statistical analysis for the Shannon index values of the L-compartment

```
# Calculate Mean and SD
mean(psL.meta$Shannon)
sd(psL.meta$Shannon)

# Linear Model
Model <- lm(Shannon ~ PHPP_conc, data = psL.meta, na.action = na.omit)
summary(Model)
anova(Model)
```

The following does the statistical analysis for the Shannon index values of the T-compartment

```
# Calculate Mean and SD
mean(psT.meta$Shannon)
sd(psT.meta$Shannon)

# Linear Model
Model <- lm(Shannon ~ PHPP_conc, data = psT.meta, na.action = na.omit)
summary(Model)
anova(Model)
```

Beta Diversity:

Loosely associated root microbiome: First, the distance matrix calculated by QIIME2 needs to be merged with the metadata file:

```
RA.dist <- RA.dist.L #using a temporary variable
tmp_order <- colnames(RA.dist)
row.names(q_meta) <- q_meta$SampleID
de <- merge(q_meta, RA.dist, by=0, all=F, all.y=F)
row.names(de) <- de$SampleID
de <- de[ order(match(row.names(de), tmp_order)), ]
RA.dist <- RA.dist[ order(match(row.names(RA.dist), tmp_order)), ]
```

Then, PERMANOVA is calculated on the distance matrix

```
adonis2(formula = RA.dist ~ de$PHPP + de$PHPP:de$Application , permutations = 999)
```

Furthermore, PERMDISP is calculated either for the PHPP treatment and the grouped variable

```
mod <- betadisper(as.dist(RA.dist), de$PHPP); anova(mod) # p: 0.6419
#tukey <- TukeyHSD(mod, which = "group", ordered = F, conf.level = 0.95); tukey$group

mod <- betadisper(as.dist(RA.dist), de$Grouped); anova(mod) # p: 0.838
#tukey <- TukeyHSD(mod, which = "group", ordered = F, conf.level = 0.95); tukey$group
```

Tightly associated root microbiome: First, the distance matrix calculated by QIIME2 needs to be merged with the metadata file:

```
RA.dist <- RA.dist.T #using a temporary variable
tmp_order <- colnames(RA.dist)
row.names(q_meta) <- q_meta$SampleID
de <- merge(q_meta, RA.dist, by=0, all=F, all.y=F)
row.names(de) <- de$SampleID
de <- de[ order(match(row.names(de), tmp_order)), ]
RA.dist <- RA.dist[ order(match(row.names(RA.dist), tmp_order)), ]
```

Then, PERMANOVA is calculated on the distance matrix

```
adonis2(formula = RA.dist ~ de$PHPP + de$PHPP:de$Application , permutations = 999)
```

Furthermore, PERMDISP is calculated either for the PHPP treatment and the grouped variable

```
mod <- betadisper(as.dist(RA.dist), de$PHPP); anova(mod) # p: 0.6419
#tukey <- TukeyHSD(mod, which = "group", ordered = F, conf.level = 0.95); tukey$group

mod <- betadisper(as.dist(RA.dist), de$Grouped); anova(mod) # p: 0.838
#tukey <- TukeyHSD(mod, which = "group", ordered = F, conf.level = 0.95); tukey$group
```

```
# upper panel for L-compartment
RA.ord <- RA.ord.L #using a temporary variable
panelA <- RA.ord$data$Vectors %>%
  dplyr::select(SampleID, PC1, PC2) %>%
  left_join(q_meta) %>%
  mutate(PHPP = fct_relevel(PHPP, names_short)) %>%
  ggplot(aes(x=PC1, y=PC2, color=PHPP, shape=Application)) +
  geom_point(alpha=0.9, size=12) + deicode_theme +
  guides(color = guide_legend(order = 1, override.aes = list(size=12))) +
  geom_hline(yintercept = 0, linetype="dotted") +
  geom_vline(xintercept = 0, linetype="dotted") +
  xlab(paste("PC1 - ",format(round(100*RA.ord$data$ProportionExplained[1,1],digits = 1), nsmall = 1),"%
  ylab(paste("PC2 - ",format(round(100*RA.ord$data$ProportionExplained[1,2],digits = 1), nsmall = 1),"%
  scale_shape_manual(values=c(16,15,16,15,16), name="Application", labels=c("Control", "Double", "Recom
  theme(legend.text.align = 0, legend.spacing.y = unit(1, 'cm'), legend.key.height = unit(1, 'cm')) +
  scale_color_manual(values = col_short, name="PHPP",
                     labels = names_short)
print(panelA)

# upper panel for T-compartment
RA.ord <- RA.ord.T #using a temporary variable
panelB <- RA.ord$data$Vectors %>%
  dplyr::select(SampleID, PC1, PC2) %>%
  left_join(q_meta) %>%
  mutate(PHPP = fct_relevel(PHPP, names_short)) %>%
  ggplot(aes(x=PC1, y=PC2, color=PHPP, shape=Application)) +
  geom_point(alpha=0.9, size=12) + deicode_theme +
```

```
  guides(color = guide_legend(order = 1, override.aes = list(size=12))) +
  geom_hline(yintercept = 0, linetype="dotted") +
  geom_vline(xintercept = 0, linetype="dotted") +
  xlab(paste("PC1 - ",format(round(100*RA.ord$data$ProportionExplained[1,1],digits = 1), nsmall = 1),"%
  ylab(paste("PC2 - ",format(round(100*RA.ord$data$ProportionExplained[1,2],digits = 1), nsmall = 1),"%
  scale_shape_manual(values=c(16,15,16,15,16), name="Application", labels=c("Control", "Double", "Recom
  theme(legend.text.align = 0, legend.spacing.y = unit(1, 'cm'), legend.key.height = unit(1, 'cm')) +
  scale_color_manual(values = col_short, name="PHPP",
                     labels = names_short)
print(panelB)
```

Figure 2:

Phylogenetic Beta Diversity:

Figure S2: Load necessary packages and create theme for plots. Then, betaNTI and betaNRI, as well as tNST, are calculated for the compartments and timepoints individually.

```
library(microeco)
library(file2meco)
library(scales)
library(phyloseq)
library(magrittr)
library(ggplot2)
library(ggpubr)
library(dplyr)

dir.create("stat_results")
dir.create("stat_results/PD")

names_long <- c("Control","r.Aliette","d.Aliette","r.Luna","d.Luna","r.Movento", "d.Movento","w/o.Serena
q_meta$Grouped <- factor(q_meta$Grouped , levels = names_long)
sample_data(ps)$PHPP_conc <- factor(sample_data(ps)$PHPP_conc, levels = names_long)

### ind. timepoints - ssp-----------------------------------------------------

# same species pool
for (comp in c("L","T")) {

  # generate meco object
  rs = "factor"
  pseq <- subset_samples(ps, Compartment==comp)
  dataset <- phyloseq2meco(pseq)
  dataset$tidy_dataset(); dataset

  # generate trans_nullmodel object
  t1 <- trans_nullmodel$new(dataset)


  #### beta NRI ----------------------------------------------------------
  method <- "bNRI"
  # see null.model parameter for other null models
```

```r
# null model run 500 times for the example
t1$cal_ses_betampd(runs=999, abundance.weighted = TRUE)
# return t1$res_ses_betampd

# add betaNRI matrix to beta_diversity list
dataset$beta_diversity[["betaNRI"]] <- t1$res_ses_betampd
# create trans_beta class, use measure "betaNRI"
t2 <- trans_beta$new(dataset = dataset, group = "PHPP_conc", measure = "betaNRI")
# transform the distance for each group
t2$cal_group_distance(); t2$res_group_distance

# export data
t2$res_group_distance %>%
  write.table( file = paste0("../stat_results/PD/",method,"_",comp,"_",rs,"_factor_rawdata_ssp.tsv"),

#calculate Mean and SD
t2$res_group_distance %>% group_by(PHPP_conc) %>% summarise(Average=mean(Value), SD=sd(Value)) %>%
  write.table( file = paste0("../stat_results/PD/",method,"_",comp,"_",rs,"_factor_mean_ssp.tsv"), ro

# export ANOVA and post-hoc tests
sink(file = paste0("../stat_results/PD/",method,"_",comp,"_",rs,"_factor_ANOVA_ssp.txt"))
one.way <- aov(Value ~ PHPP_conc, data = t2$res_group_distance)
agricolae::SNK.test(one.way, "PHPP_conc", alpha = 0.05, group=TRUE, main = NULL,console=TRUE)
sink(file = NULL)


# prepare the plot
tmp <- t2$res_group_distance
tmp$PHPP_conc <- factor(tmp$PHPP_conc)
anova <- aov(Value ~ PHPP_conc, data = tmp)
tukey <- TukeyHSD(anova)

cld <- multcompView::multcompLetters4(anova, tukey)

# table with factors and 3rd quantile
dt <- group_by(tmp, PHPP_conc) %>%
  dplyr::summarise(w=mean(Value), sd = sd(Value)) %>%
  arrange(desc(w))

# extracting the compact letter display and adding to the Tk table
cld <- as.data.frame.list(cld$PHPP_conc)
dt$cld <- cld$Letters
dt$PHPP_conc <- factor(dt$PHPP_conc, levels = c("Control", "r.Aliette", "d.Aliette", "r.Luna", "d.Luna
tmp$PHPP_conc <- factor(tmp$PHPP_conc, levels = c("Control", "r.Aliette", "d.Aliette", "r.Luna", "d.L

# plot the results
plot = ggplot(tmp, aes(x=PHPP_conc, y=Value, fill=PHPP_conc)) +
  geom_boxplot() +
  geom_hline(yintercept = -2, linetype = 2) + geom_hline(yintercept = 2, linetype = 2) +
  scale_fill_manual(values=c("gray10", "red1", "red4", "purple1", "purple4", "orange1", "orange3", "pa
  xlab("") + ylab("betaNRI") + theme(legend.position = "") +
  coord_cartesian(ylim = c(-6, 4)) +
  scale_y_continuous(breaks = seq(-6, 4, by = 2)) +
```

```
  plot_theme +
  geom_text(data = dt, aes(x = PHPP_conc, y = w, label = cld), size = 12, color = "black", hjust = 0.5
plot$layers[[2]]$aes_params$textsize <- 16
print(plot)


### beta NTI ----------------------------------------------------------
method <- "bNTI"
# null model run 500 times
t1$cal_ses_betamntd(runs=999, abundance.weighted = TRUE, nworker = 10, use_iCAMP_force = FALSE)
# add betaNRI matrix to beta_diversity list
dataset$beta_diversity[["betaNTI"]] <- t1$res_ses_betamntd
# create trans_beta class, use measure "betaNRI"
t2 <- trans_beta$new(dataset = dataset, group = "PHPP_conc", measure = "betaNTI")
# transform the distance for each group
t2$cal_group_distance(); t2$res_group_distance

# export data
t2$res_group_distance %>%
  write.table( file = paste0("../stat_results/PD/",method,"_",comp,"_",rs,"_factor_rawdata_ssp.tsv"),

#calculate Mean and SD
t2$res_group_distance %>% group_by(PHPP_conc) %>% summarise(Average=mean(Value), SD=sd(Value)) %>%
  write.table( file = paste0("../stat_results/PD/",method,"_",comp,"_",rs,"_factor_mean_ssp.tsv"), ro

# export ANOVA and post-hoc tests
sink(file = paste0("../stat_results/PD/",method,"_",comp,"_",rs,"_factor_ANOVA_ssp.txt"))
one.way <- aov(Value ~ PHPP_conc, data = t2$res_group_distance); summary(one.way)
agricolae::SNK.test(one.way, "PHPP_conc", alpha = 0.05, group=TRUE, main = NULL,console=TRUE)
sink(file = NULL)

# prepare the plot
tmp <- t2$res_group_distance
anova <- aov(Value ~ PHPP_conc, data = tmp)
tukey <- TukeyHSD(anova)

cld <- multcompView::multcompLetters4(anova, tukey)

# table with factors and 3rd quantile
dt <- group_by(tmp, PHPP_conc) %>%
  dplyr::summarise(w=mean(Value), sd = sd(Value)) %>%
  arrange(desc(w))

# extracting the compact letter display and adding to the Tk table
cld <- as.data.frame.list(cld$PHPP_conc)
dt$cld <- cld$Letters
dt$PHPP_conc <- factor(dt$PHPP_conc, levels = c("Control", "r.Aliette", "d.Aliette", "r.Luna", "d.Luna
tmp$PHPP_conc <- factor(tmp$PHPP_conc, levels = c("Control", "r.Aliette", "d.Aliette", "r.Luna", "d.Lu

# plot the results
plot = ggplot(tmp, aes(x=PHPP_conc, y=Value, fill=PHPP_conc)) +
  geom_boxplot() +
  geom_hline(yintercept = -2, linetype = 2) + geom_hline(yintercept = 2, linetype = 2) +
```

```
    scale_fill_manual(values=c("gray10", "red1", "red4", "purple1", "purple4", "orange1", "orange3", "pa
    xlab("") + ylab("betaNTI") + theme(legend.position = "") +
    coord_cartesian(ylim = c(-6, 4)) +
    scale_y_continuous(breaks = seq(-6, 4, by = 2)) +
    plot_theme +
    geom_text(data = dt, aes(x = PHPP_conc, y = w, label = cld), size = 12, color = "black", hjust = 0.5
  plot$layers[[2]]$aes_params$textsize <- 16
  print(plot)
}
```

Figure S7: tNST is calculated for each compartment and timepoint individually with both Bray's and Jaccard's distance.

```
for (comp in c("L","T")) {
  for (dist in c("jaccard", "bray")) {

    method <- "NST"
    rs <- "factor"
    pseq <- subset_samples(ps, Compartment==comp)

    meta.df = as(sample_data(pseq), "matrix")
    meta.df = as.data.frame(meta.df)
    meta.df = meta.df[,"PHPP_conc", drop=FALSE]

    OTU1 = as(otu_table(pseq), "matrix")
    if(taxa_are_rows(pseq)){OTU1 <- t(OTU1)}
    comm.df = as.data.frame(OTU1)

    # calculate NST
    cal_nst <- NST::tNST(comm = comm.df, group= meta.df, rand = 999, nworker = 8, between.group = FALSE

    # bootstrap NST
    cal_nst.boot <- NST::nst.boot(nst.result = cal_nst, group = meta.df, rand = 999, trace = TRUE, two.

    cal_nst.boot$compare[,-c(7,9:10)] %>% write.table( file = paste0("../stat_results/PD/",method,"_",c

    # PERMANOVA
    cal_nst.permanova <- NST::nst.panova(nst.result = cal_nst, group = meta.df, rand = 999, trace=TRUE,

    cal_nst.permanova %>% write.table( file = paste0("../stat_results/PD/",method,"_",comp,"_",rs,"_",di

    tmp <- cal_nst.boot$detail$NST.boot
    df <- plyr::ldply (tmp,data.frame)

    # export data
    df %>% write.table( file = paste0("../stat_results/PD/",method,"_",comp,"_",rs,"_",dist,"_factor_rau

    df$.id <- factor(df$.id, levels = c("Control", "r.Aliette", "d.Aliette", "r.Luna", "d.Luna" ,"r.Move

    # plot
    p_tNSTI <- ggplot(df, aes(x=.id, y=X..i.., fill=.id)) +
      geom_boxplot() +
      xlab("") + ylab("tNST") +
```

```
      coord_cartesian(ylim = c(0, 1), expand = T) +
      plot_theme + theme(legend.position = "") +
      scale_y_continuous(breaks = seq(0, 1, by = 0.2)) +
      geom_hline(yintercept = 0, linetype = 1) + geom_hline(yintercept = 1, linetype = 2) +
      scale_fill_manual(values=c("gray10", "red1", "red4", "purple1", "purple4", "orange1", "orange3",
    p_tNSTI$layers[[2]]$aes_params$textsize <- 20
    print(p_tNSTI)
  }
}
```

Differential abundance analysis:

```
library(dplyr)
library(ANCOMBC)
library(nlme)
library(tidyverse)
library(compositions)
library(readr)
library(DT)
library(matrixStats)
library(pheatmap)
library(RColorBrewer)
library(viridis)
library(dendsort)
library(qiime2R)
library(phyloseq)
library(microbiome)
library(ComplexHeatmap) #Heatmap function
library(circlize)
library(round)
library(tidyr)
library(readr)
library(tibble)

ps <- readRDS("phyloseq/ps.RDS")
dir.create("stat_results/ANCOMBC/")
dir.create("stat_results/ANCOMBC/pairwise")
dir.create("stat_results/ANCOMBC/pairwise/genus")


names_short <- c("CONTROL","ALIETTE","LUNA","MOVENTO","SERENADE")
names_long <- c(names_long <- c("Control","r.Aliette","d.Aliette","r.Luna","d.Luna","r.Movento", "d.Move



# Timepoints individually: ------------------------------------------------------------------
for (comp in c("L","T")){

  pseq <- subset_samples(ps, Compartment==comp)
  genus_data <- tax_glom(pseq, taxrank <- rank_names(pseq)[6], NArm = FALSE)
  var1 <- "PHPP_conc"
```

```
  tax="genus"

  ## to control
  var2="Control"
  sample_data(genus_data)$PHPP_conc <- factor(sample_data(genus_data)$PHPP_conc, levels = names_long)

  # Run ancombc function
  out = ancombc(phyloseq = genus_data, formula = "PHPP_conc",
                p_adj_method = "fdr", zero_cut = 0.90, lib_cut = 10000,
                group = var1, struc_zero = TRUE, neg_lb = FALSE,
                tol = 1e-5, max_iter = 100, conserve = TRUE,
                alpha = 0.1, global = FALSE)
  res = out$res

  #Coefficients
  tab_coef = res$W
  colnames(tab_coef)
  col_name = c("Control - r.Aliette", "Control - d.Aliette", "Control - r.Luna", "Control - d.Luna",  "
  colnames(tab_coef) = col_name

  source("ancom_bc_rename_variables.R", echo = T, spaced = T)
  tmp <- tab_diff
  tmp2 <- merge(tmp, tab_w, by=0)
  write_csv(tmp2, file = paste0("stat_results/ANCOMBC/pairwise/",tax,"/",comp,"-",var2,".csv"))

  var2="Control";R1 <- read.csv(file = paste0("stat_results/ANCOMBC/pairwise/",tax,"/",comp,"-",var2,".
  tmp <- R1
  rownames(tmp) <- tmp$Row.names; tmp <- tmp[,-1]

  pseq <- subset_samples(ps, Compartment==comp)
  genus_data <- tax_glom(pseq, taxrank <- rank_names(pseq)[6], NArm = FALSE)
  PGroup <- transform_sample_counts(genus_data, function(x)100* x / sum(x))
  OTUg <- otu_table(PGroup)
  AverageD <- as.data.frame(rowMeans(OTUg))
  names(AverageD) <- "Mean"
  SD <- as.data.frame(rowSds(OTUg),na.rm = T)
  names(SD) <- "SD"
  tmp2 <- cbind(AverageD,SD)
  tmp3 <- merge(tmp,tmp2, by=0, all=TRUE)

  write_csv(tmp3, file = paste0("stat_results/ANCOMBC/pairwise/",tax,"/",comp,"-",var1,"_results.csv"))
}


for (comp in c("L","T")){

  ###ANCOM results
  tmp <- read.csv(file = paste0("stat_results/ANCOMBC/pairwise/",tax,"/",comp,"-",var1,"_results.csv"))

  rownames(tmp) <- tmp$Row.names; tmp$Row.names = NULL
  pseq <- subset_samples(ps, Compartment==comp)
  genus_data <- tax_glom(pseq, taxrank <- rank_names(pseq)[6], NArm = FALSE)
```

```
  ttable <- as.data.frame(tax_table(genus_data))
  GTable <- merge(ttable,tmp, by=0)
  rownames(GTable) <- GTable$Row.names; GTable$Row.names = NULL
  head(GTable)

  ancom <- GTable[(GTable$Mean>=0.1),] %>% drop_na(P)

  ancom <- ancom[,-c(1)]
  ancom$Phylum <- ancom$P
  ancom <- merge(ancom, p.colors, by="Phylum", all.y = F)
  ancom$G[ancom$G=="uncultured"] <- "Unclassified"
  ancom$G[is.na(ancom$G)] <- "Unclassified"
  ancom <- add_column(ancom, Name = ancom$G, .after = "G")
  ancom <- ancom[(rowSums(ancom[,c(8:11)])!=0),] %>% drop_na(Phylum)
  head(ancom)
  ancom <- ancom[,-c(2)]
  colnames(ancom)[1:5] <- c("Phylum","Class","Order","Family","Genus")

  tax.clean <- ancom
  colnames(tax.clean)

  tax.clean <- rename.ancombc.output(ancom)
  write_csv(tax.clean, file = paste0("stat_results/ANCOMBC/pairwise/",tax,"/",comp,"_subplots_combined.
  file.copy(from = paste0("stat_results/ANCOMBC/pairwise/",tax,"/",comp,"_subplots_combined.csv"), to =
  #redo in excel! -> add
}


for (comp in c("L","T")){

  mat.df <- read.csv(file = paste0("stat_results/ANCOMBC/pairwise/",tax,"/",comp,"_subplots_mod.csv"), 
  str(mat.df)

  cat(paste(shQuote(unique(colnames(mat.df)), type="cmd"), collapse=", "))
  colnames(mat.df) <- c("Phylum", "Control->Aliette.sig->", "Control->Luna.sig", "Control->Movento.sig"

  #IMPORTANT: specify ONLY the columns with the differentials
  col.order <- c("Control-> Aliette", "Control -> Luna", "Control -> Movento", "Control -> Serenade")
  mat.diff <- as.matrix(mat.df[,c(6:9)])
  sig_mat <- as.matrix(mat.df[,c(2:5)])
  #rownames(mat.diff) <- mat.df$Taxonomy
  min_lc <- min(mat.diff, na.rm = T)
  max_lc <- max(mat.diff, na.rm = T)
  colors <- colorRamp2(c(min_lc, 0, max_lc), c("blue", "white", "red"))

  hb = rowAnnotation('Mean Abundance [%]' = anno_barplot(as.vector(mat.df$Mean), gp = gpar(fill = "blac

  lgd = Legend(at = c(round(min_lc+min_lc*0.05,digits = 1), 0, round(max_lc+max_lc*0.05, digits = 1)),

  ht = Heatmap(mat.diff , name = "logfold change",
               column_gap = unit(5, "mm"),
               row_split = mat.df$Phylum,
               row_gap = unit(5, "mm"),
```

```
                row_names_gp = gpar(fontsize = 32, fontface = "bold",
                                    col = mat.df$P.color),
                #column_split = c("A","A","B","B","C","C","D","D"),
                na_col = "grey",
                col = colors,
                rect_gp = gpar(col = "white", lwd = 2), #white spaces in between
                border = TRUE,
                cluster_rows = FALSE, #remove cluster
                show_column_dend = FALSE, #remove cluster
                show_heatmap_legend = FALSE, #r emove legend
                row_names_side = "right",
                row_names_max_width = unit(2, "cm"),
                row_names_rot = 0,
                row_names_centered = FALSE,
                row_title_gp = gpar(fontsize = 28),
                row_title_rot = 0,
                column_title = paste0("ANCOM-BC W-value"),
                column_title_gp = gpar(fill = "black", col = "white", fontsize = 28, fontface = "bold", l
                column_title_side = "top",
                column_names_max_height = unit(6, "cm"),
                column_names_gp = gpar(fontsize = 30, fontface = "bold", col = "black"),
                column_names_rot = 90,
                column_names_centered = FALSE,
                column_order = col.order,
                show_parent_dend_line = FALSE,
                cell_fun = function(j, i, x, y, width, height, fill) {
                  if(sig_mat[i, j] == "TRUE")
                    grid.text("*", x, y, gp = gpar(fontsize = 28, fontface = "bold"))
                },
                right_annotation = c(hb))
  draw(ht, padding = unit(c(10, 1, 1, 26), "cm")) # add space for titles
  draw(lgd, x = unit(45, "cm"), y = unit(8, "cm"))
}
```

Figure S10: