Combined trial figures

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(plyr)
library(phyloseq)
library(dplyr)
library(ggplot2)
library(RColorBrewer)
library(colorRamps)
library(data.table)
library(readxl)
library(data.table)
library(forcats)


set.seed(225)
```

This beginning workspace contains:

- `temp`: phyloseq object including the metadata, and the ASV table, the taxonomy table and the phylogenetic tree, all three created by QIIME2, for the temporal trial
- `conc`: phyloseq object including the metadata, and the ASV table, the taxonomy table and the phylogenetic tree, all three created by QIIME2, for the concentration trial
- `straw`: phyloseq object including the metadata, and the ASV table, the taxonomy table and the phylogenetic tree, all three created by QIIME2, for the strawberry trial
- `colorcodes`: a list of colors assigned to the phyla, classes and orders

```
load('combined_plots.RData')
```

Load functions neccessary:

```
source('functions.R')
```

Relative abundance barplot at family level:

Figure S13

First transform the absolute abundances to relative abundances and merge the samples to their regarding root compartment (L- or T-compartment). Then glomerate the taxonomic levels at order level. Then extract only the necessary metadata columns into order to bind the three datasets into one large table. Furthermore add a column which specifies to which compartment and trial the data belongs to. Also assign each order with a relative abundance of less then 2% as “other”.

```
# Transformation
temp <- temp %>%
  microbiome::transform("compositional") %>%
  merge_samples("compartment") %>%
  transform_sample_counts(function(x) 100 * x/sum(x)) %>%
  tax_glom(taxrank = "O", NArm = FALSE)
conc <- conc %>%
  microbiome::transform("compositional") %>%
  merge_samples("Compartment") %>%
  transform_sample_counts(function(x) 100 * x/sum(x)) %>%
  tax_glom(taxrank = "O", NArm = FALSE)
straw <- straw %>%
  subset_samples( Compartment != "Soil")%>%
  microbiome::transform("compositional") %>%
  merge_samples("Compartment") %>%
  transform_sample_counts(function(x) 100 * x/sum(x)) %>%
  tax_glom(taxrank = "O", NArm = FALSE)

# Merge
dat_taxa <- data.table(psmelt(temp))
dat_taxa$O <- as.character(dat_taxa$O)
medians <- dat_taxa[, mean := mean(Abundance, na.rm = TRUE), by = "O"]
dat_taxa[is.na(dat_taxa)] <- "Unclassified"
dat_taxa <- dat_taxa[,-c(1,4:18)]
dat_taxa$trial <- as.character("temp")
dat_taxa$axis <- as.character(paste0(dat_taxa$trial,"-",dat_taxa$Sample))
remainder1 <- dat_taxa[(mean <= 1), O := "Other"]

dat_taxa <- data.table(psmelt(conc))
dat_taxa$O <- as.character(dat_taxa$O)
medians <- dat_taxa[, mean := mean(Abundance, na.rm = TRUE), by = "O"]
dat_taxa[is.na(dat_taxa)] <- "Unclassified"
dat_taxa <- dat_taxa[,-c(1,4:13)]
dat_taxa$trial <- as.character("conc")
```

```
dat_taxa$axis <- as.character(paste0(dat_taxa$trial,"-",dat_taxa$Sample))
remainder2 <- dat_taxa[(mean <= 1), O := "Other"]

dat_taxa <- data.table(psmelt(straw))
dat_taxa$O <- as.character(dat_taxa$O)
medians <- dat_taxa[, mean := mean(Abundance, na.rm = TRUE), by = "O"]
dat_taxa[is.na(dat_taxa)] <- "Unclassified"
dat_taxa <- dat_taxa[,-c(1,4:16)]
dat_taxa$trial <- as.character("straw")
dat_taxa$axis <- as.character(paste0(dat_taxa$trial,"-",dat_taxa$Sample))
remainder3 <- dat_taxa[(mean <= 1), O := "Other"]

#merge
remainder <- rbind(remainder1, remainder2, remainder3)

remainder$axis <- mapvalues(remainder$axis, from = c("temp-L", "temp-T", "temp-B", "conc-T", "conc-L",

level_order1 <- factor(remainder$axis, level = c("Temporal-B","Concentration-B","Strawberry-B", "Tempora
```

Rename the columns and only filter phyla of interest:

- `Acidobacteriota`
- `Actinobacteriota`
- `Firmicutes`
- `Proteobacteria`
- `Unclassified`: all unclassified order
- `Other Phyla`: phyla besides the ones listed above were merged here.

Furthermore, order columns are renamed to be more readable

```
tmp1 <- remainder
tmp2 <- tmp1[(mean <= 2), O := "Other"]

df <- tmp2
names(df)[names(df) == "P"] <- "Phylum"
names(df)[names(df) == "O"] <- "Order"
df$group <- paste0(df$Phylum, "-", df$Order, sep = "")
sort(unique(df$group))
cat(paste(shQuote(sort(unique(df$Phylum)), type="cmd"), collapse=", "))

#test
phylums <- c("Acidobacteriota","Actinobacteriota","Firmicutes","Proteobacteria")

df$Order[df$Phylum=="Acidobacteriota" & df$Order=='Acidobacteriota-Other'] <- "Other Acidobacteria"

df$Order[df$Phylum=="Actinobacteriota" & df$Order=='Actinobacteriota-Other'] <- "Other Actinobacteriota

df$Order[df$Phylum=="Firmicutes" & df$Order=='Firmicutes-Other'] <- "Other Firmicutes"

df$Order[df$Phylum=="Proteobacteria" & df$Order=='Proteobacteria-Other'] <- "Other Proteobacteria"

df$group[df$group=='Unclassified-Other'] <- "Unclassified"
```

```
df$group[!df$Phylum %in% phylums] <- "Other Phyla"


df2 <- dplyr::select(df, axis, Phylum, Order, Abundance, group) %>%
  dplyr::mutate(Phylum=factor(Phylum, levels=c(phylums, "Others")),
                Order=fct_reorder(Order, 10*as.integer(Phylum) + grepl("Others", Order)))
```

```
sort(unique(df2$group))
colours <- c("chartreuse1","chartreuse2","chartreuse3","chartreuse4", #Acidobacteriota
             "chocolate1","chocolate3","chocolate4", #Actinobacteriota
             "hotpink1","hotpink3", #Firmicutes
             "khaki", #Other
             plasma(8)) #Proteobacteria

ggplot(df2, aes(x=level_order1, y=Abundance, fill=group, order=group)) +
  geom_bar(aes(fill=group), stat="identity", position="stack") +
  mytheme + theme(axis.text.x = element_text(angle=75, vjust = 0.5)) +
  scale_fill_manual("", values=colours) +
  scale_x_discrete("") +
  theme(axis.text.x = element_text(angle=75, vjust=0.5, size=24),
        axis.text.y = element_text(size = 24),
        axis.title.y = element_text(size = 24)) +
  scale_y_continuous("Relative abundance [%]", breaks=seq(0,100,5), limits = c(0, 101)) +
  guides(fill=guide_legend(nrow=length(unique(df2$group)))) + labs(fill = "Family") +
  geom_vline(xintercept = c(3.5,6.5), color = "black", size=2.5)
```

Colors are chosen for the families and the barplot created:

Differential abundance analysis at phylum level:

Statisical anaylsis

Load necessary ecological analysis libraries.

```
library(ggplot2)
library(phyloseq)
library(microbiome)
library(ggord)
library(metagMisc)
library(nlme)
library(ggpubr)
library(FSA)
library(knitr)
library(rmarkdown)
library(ape)
```

```r
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(lme4)
library(lmerTest)
library(forcats)
library(emmeans)
library(multcomp)
library(RColorBrewer)

load('combined_plots.RData')
dir.create("Data")
dir.create("Plots")

set.seed(225)
```

Figure S14: First the statistical analysis is done for each trial at order level and the exported as .csv file

```r
## Temporal: -------------------------------------------------------------------
ps <- temp
pseq <- ps
genus_data <- tax_glom(pseq, taxrank <- rank_names(pseq)[4], NArm = FALSE)
var1 <- "compartment"


# Bulk Soil
sample_data(genus_data)$compartment <- factor(sample_data(genus_data)$compartment, levels =
                                          c("B" ,"L", "T"))

### Run ancombc function
out = ancombc(phyloseq = genus_data, formula = var1,
              p_adj_method = "holm", zero_cut = 0.90, lib_cut = 10000,
              group = var1, struc_zero = TRUE, neg_lb = FALSE,
              tol = 1e-5, max_iter = 100, conserve = TRUE,
              alpha = 0.05, global = FALSE)
res = out$res

### Coefficients
tab_coef = res$W
```

```r
colnames(tab_coef)
col_name = c("B - L", "B - T")
colnames(tab_coef) = col_name
source("ancom_bc_rename_variables.R", echo = T, spaced = T)
tmp <- tab_diff
bulk <- merge(tmp, tab_w, by=0)
rownames(bulk) <- bulk$Row.names
bulk$Row.names = NULL


# Loosely
sample_data(genus_data)$compartment <- factor(sample_data(genus_data)$compartment, levels =
                                                c("L" ,"B", "T"))

### Run ancombc function
out = ancombc(phyloseq = genus_data, formula = var1,
              p_adj_method = "holm", zero_cut = 0.90, lib_cut = 10000,
              group = var1, struc_zero = TRUE, neg_lb = FALSE,
              tol = 1e-5, max_iter = 100, conserve = TRUE,
              alpha = 0.05, global = FALSE)
res = out$res

### Coefficients
tab_coef = res$W
colnames(tab_coef)
col_name = c("L - B", "L - T")
colnames(tab_coef) = col_name
source("ancom_bc_rename_variables.R", echo = T, spaced = T)
tmp <- tab_diff
loosely <- merge(tmp, tab_w, by=0)
rownames(loosely) <- loosely$Row.names
loosely$Row.names = NULL
loosely <- loosely[,-c(1,3)]


tmp <- merge(bulk, loosely, by=0); rownames(tmp) <- tmp$Row.names; tmp$Row.names = NULL
tmp3 <- as.data.frame(tax_table(genus_data))
tmp4 <- merge(tmp3, tmp, by=0); rownames(tmp4) <- tmp4$Row.names; tmp4$Row.names = NULL
tmp4 <- tmp4[,-c(1,5:6)]
head(tmp4)


PGroup <- transform_sample_counts(genus_data, function(x)100* x / sum(x))
OTUg <- otu_table(PGroup)
TAXg <- tax_table(PGroup)
AveragedD <- as.data.frame(rowMeans(OTUg))
names(AveragedD) <- c("Mean")
SD <- as.data.frame(rowSds(OTUg),na.rm = T)
names(SD) <- c("SD")
tmp <- cbind(AveragedD,SD)
GTable <- merge(TAXg, tmp, by=0, all=TRUE); rownames(GTable) <- GTable$Row.names; GTable$Row.names = NUL
GTable <- GTable[,-c(1,5:6)]
head(GTable)
```

```
tmp5 <- merge(tmp4, GTable, by=0); row.names(tmp5) <- tmp5$Row.names; tmp5$Row.names = NULL
tmp6 <- tmp5[,-c(10:12)]
temp <- tmp6[order(tmp6$Mean, decreasing = TRUE),]
View(temp)
write_csv(temp, file = "Data/temporal_top_families.csv")


## Concentration: ----------------------------------------------------------------
ps <- conc
pseq <- ps
genus_data <- tax_glom(pseq, taxrank <- rank_names(pseq)[4], NArm = FALSE)
var1 <- "Compartment"

# Bulk Soil
sample_data(genus_data)$Compartment <- factor(sample_data(genus_data)$Compartment, levels =
                                                c("B" ,"L", "T"))

### Run ancombc function
out = ancombc(phyloseq = genus_data, formula = var1,
              p_adj_method = "holm", zero_cut = 0.90, lib_cut = 10000,
              group = var1, struc_zero = TRUE, neg_lb = FALSE,
              tol = 1e-5, max_iter = 100, conserve = TRUE,
              alpha = 0.05, global = FALSE)
res = out$res

### Coefficients
tab_coef = res$W
colnames(tab_coef)
col_name = c("B - L", "B - T")
colnames(tab_coef) = col_name
source("ancom_bc_rename_variables.R", echo = T, spaced = T)
tmp <- tab_diff
bulk <- merge(tmp, tab_w, by=0)
rownames(bulk) <- bulk$Row.names
bulk$Row.names = NULL


# Loosely
sample_data(genus_data)$Compartment <- factor(sample_data(genus_data)$Compartment, levels =
                                                c("L" ,"B", "T"))

### Run ancombc function
out = ancombc(phyloseq = genus_data, formula = var1,
              p_adj_method = "holm", zero_cut = 0.90, lib_cut = 10000,
              group = var1, struc_zero = TRUE, neg_lb = FALSE,
              tol = 1e-5, max_iter = 100, conserve = TRUE,
              alpha = 0.05, global = FALSE)
res = out$res

### Coefficients
tab_coef = res$W
colnames(tab_coef)
```

```
col_name = c("L - B", "L - T")
colnames(tab_coef) = col_name
source("ancom_bc_rename_variables.R", echo = T, spaced = T)
tmp <- tab_diff
loosely <- merge(tmp, tab_w, by=0)
rownames(loosely) <- loosely$Row.names
loosely$Row.names = NULL
loosely <- loosely[,-c(1,3)]


tmp <- merge(bulk, loosely, by=0); rownames(tmp) <- tmp$Row.names; tmp$Row.names = NULL
tmp3 <- as.data.frame(tax_table(genus_data))
tmp4 <- merge(tmp3, tmp, by=0); rownames(tmp4) <- tmp4$Row.names; tmp4$Row.names = NULL
tmp4 <- tmp4[,-c(1,5:6)]
head(tmp4)


PGroup <- transform_sample_counts(genus_data, function(x)100* x / sum(x))
OTUg <- otu_table(PGroup)
TAXg <- tax_table(PGroup)
AverageD <- as.data.frame(rowMeans(OTUg))
names(AverageD) <- c("Mean")
SD <- as.data.frame(rowSds(OTUg),na.rm = T)
names(SD) <- c("SD")
tmp <- cbind(AverageD,SD)
GTable <- merge(TAXg, tmp, by=0, all=TRUE); rownames(GTable) <- GTable$Row.names; GTable$Row.names = NUL
GTable <- GTable[,-c(1,5:6)]
head(GTable)


tmp5 <- merge(tmp4, GTable, by=0); row.names(tmp5) <- tmp5$Row.names; tmp5$Row.names = NULL
tmp6 <- tmp5[,-c(10:12)]
conc <- tmp6[order(tmp6$Mean, decreasing = TRUE),]
View(conc)
write_csv(conc, file = "Data/concentration_top_families.csv")



## Strawberry: -----------------------------------------------------------------
ps <- straw
pseq <- ps
genus_data <- tax_glom(pseq, taxrank <- rank_names(pseq)[4], NArm = FALSE)
var1 <- "Compartment"

# Bulk Soil
sample_data(genus_data)$Compartment <- factor(sample_data(genus_data)$Compartment, levels =
                                                c("B" ,"L", "T"))

### Run ancombc function
out = ancombc(phyloseq = genus_data, formula = var1,
              p_adj_method = "holm", zero_cut = 0.90, lib_cut = 10000,
              group = var1, struc_zero = TRUE, neg_lb = FALSE,
              tol = 1e-5, max_iter = 100, conserve = TRUE,
```

```r
              alpha = 0.05, global = FALSE)
res = out$res

### Coefficients
tab_coef = res$W
colnames(tab_coef)
col_name = c("B - L", "B - T")
colnames(tab_coef) = col_name
source("ancom_bc_rename_variables.R", echo = T, spaced = T)
tmp <- tab_diff
bulk <- merge(tmp, tab_w, by=0)
rownames(bulk) <- bulk$Row.names
bulk$Row.names = NULL


# Loosely
sample_data(genus_data)$Compartment <- factor(sample_data(genus_data)$Compartment, levels =
                                                c("L" ,"B", "T"))

### Run ancombc function
out = ancombc(phyloseq = genus_data, formula = var1,
              p_adj_method = "holm", zero_cut = 0.90, lib_cut = 10000,
              group = var1, struc_zero = TRUE, neg_lb = FALSE,
              tol = 1e-5, max_iter = 100, conserve = TRUE,
              alpha = 0.05, global = FALSE)
res = out$res

### Coefficients
tab_coef = res$W
colnames(tab_coef)
col_name = c("L - B", "L - T")
colnames(tab_coef) = col_name
source("ancom_bc_rename_variables.R", echo = T, spaced = T)
tmp <- tab_diff
loosely <- merge(tmp, tab_w, by=0)
rownames(loosely) <- loosely$Row.names
loosely$Row.names = NULL
loosely <- loosely[,-c(1,3)]


tmp <- merge(bulk, loosely, by=0); rownames(tmp) <- tmp$Row.names; tmp$Row.names = NULL
tmp3 <- as.data.frame(tax_table(genus_data))
tmp4 <- merge(tmp3, tmp, by=0); rownames(tmp4) <- tmp4$Row.names; tmp4$Row.names = NULL
tmp4 <- tmp4[,-c(1,5:6)]
head(tmp4)


PGroup <- transform_sample_counts(genus_data, function(x)100* x / sum(x))
OTUg <- otu_table(PGroup)
TAXg <- tax_table(PGroup)
AveragedD <- as.data.frame(rowMeans(OTUg))
names(AveragedD) <- c("Mean")
SD <- as.data.frame(rowSds(OTUg),na.rm = T)
```

```r
names(SD) <- c("SD")
tmp <- cbind(AverageD,SD)
GTable <- merge(TAXg, tmp, by=0, all=TRUE); rownames(GTable) <- GTable$Row.names; GTable$Row.names = NU
GTable <- GTable[,-c(1,5:6)]
head(GTable)


tmp5 <- merge(tmp4, GTable, by=0); row.names(tmp5) <- tmp5$Row.names; tmp5$Row.names = NULL
tmp6 <- tmp5[,-c(10:12)]
straw <- tmp6[order(tmp6$Mean, decreasing = TRUE),]
View(straw)
write_csv(straw, file = "Data/strawberry_top_families.csv")
```

Merge the three files into a single file Also, orders with a relative abundance <2 are excluded

```r
# MERGE: ----------------------------------------------------------------
temp <- read.csv("Data/temporal_top_families.csv")
conc <- read.csv("Data/concentration_top_families.csv")
straw <- read.csv("Data/strawberry_top_families.csv")


for (trial in c("temporal", "concentration","strawberry")){

  tax <- "Order"
  ###ANCOM results
  ancom <- read.csv(file = paste0("Data/",trial,"_top_families.csv"))
  ancom$Phylum <- ancom$P.x
  ancom <- merge(ancom, p.colors, by="Phylum", all.y = T, )
  ancom$O.x[ancom$O.x=="uncultured"] <- "Unclassified"
  ancom$O.x[is.na(ancom$O.x)] <- "Unclassified"
  ancom <- add_column(ancom, Name = ancom$O.x, .after = "O.x")
  #select only root section columns
  write_csv(ancom[,-2], file = paste0("Data/",trial,"_top_families_mod.csv"))
}

#merge all and then redo in excel!!
temp <- read.csv("Data/temporal_top_families_mod.csv")[,-c(2,3)]
conc <- read.csv("Data/concentration_top_families_mod.csv")[,-c(2,3)]
straw <- read.csv("Data/strawberry_top_families_mod.csv")[,-c(2,3)]

colnames(temp) <- c("Phylum","Name","B:L_sig","B:T_sig","B:L_W","B:T_W","L:T_sig","L:T_W","Temporal Mea
colnames(conc) <- c("Phylum","Name","B:L_sig","B:T_sig","B:L_W","B:T_W","L:T_sig","L:T_W","Concentratio
colnames(straw) <- c("Phylum","Name","B:L_sig","B:T_sig","B:L_W","B:T_W","L:T_sig","L:T_W","Strawberry

tmp <- merge(temp, conc, by="Name", all.y=T, all.x =T)
tmp <- merge(tmp, straw, by="Name", all.y=T, all.x =T)
tmp$Average <- as.numeric(rowMeans(tmp[c(9,19,29)],na.rm = TRUE))
tmp <- na.omit(tmp[tmp$Average>0.5,])
write_csv(tmp, file = "Data/top_families_mod.csv")
#redo in excel!!!! check for duplicate row names and change if neccessary; also edit rownames
```

```
#change in excel
tmp2 <- read.csv("Data/top_families_mod.csv")
tmp3 <- merge(tmp2, p.colors, by="Phylum"); row.names(tmp3) <- tmp3$Name; tmp3$Name = NULL;
ancom <- tmp3[(rowSums(tmp3[,c(2,3,6,8,9,12,14,15,18)])!=0),]

# Heatmap: ------------------------------------------------------------
mat.df <- ancom
str(mat.df)
cat(paste(shQuote(colnames(mat.df), type="cmd"), collapse=", "))

colnames(mat.df) <- c("Phylum", "B.L_sig.x", "B.T_sig.x", "Temporal: B->L", "Temporal: B->T", "L.T_sig.

#IMPORTANT: specify ONLY the columns with the differentials
col.order <- c("Temporal: B->L", "Concentration: B->L", "Strawberry: B->L",
               "Temporal: L->T", "Concentration: L->T", "Strawberry: L->T",
               "Temporal: B->T", "Concentration: B->T", "Strawberry: B->T")

mat.diff <- as.matrix(mat.df[,c(4,5,7,10,11,13,16,17,19)])
sig_mat <- as.matrix(mat.df[,c(2,3,6,8,9,12,14,15,18)])
sig_mat[is.na(sig_mat)] <- FALSE
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"))

hb1 = rowAnnotation("Mean\n Abundance [%]" = anno_barplot(as.vector(mat.df$Average), gp = gpar(fill = "
                                                       axis_param = list(side = "bottom", labels_rot =
                                                       width = unit(10, "cm")), annotation_name_gp = gpa
                   annotation_name_side = "top", annotation_name_offset = unit(0.3, "cm"), annotation_na

lgd = Legend(at = c(round(min_lc+min_lc*0.05,digits = 1), 0, round(max_lc+max_lc*0.05, digits = 1)),  c
             title = "W-value", title_gp = gpar(fontsize = 32, fontface = "bold"), title_gap = unit(0.5
             labels_gp = gpar(col = "black", fontsize = 28, fontface = "bold"), title_position = "topce
             grid_height = unit(3, "cm"), legend_width = unit(10, "cm"), direction = "horizontal")

ht = Heatmap(mat.diff , name = "logfold change",
             column_gap = unit(5, "mm"),
             row_gap = unit(5, "mm"),
             row_split = mat.df$Phylum,
             row_names_gp = gpar(fontsize = 32, col = mat.df$P.color),
             na_col = "grey",
             column_order = col.order,
             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, #remove 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 = 36, fontface = "bold"),
             row_title_rot = 0,
             column_title = paste0("ANCOM-BC"),
             column_title_gp = gpar(fill = "black", col = "white", fontsize = 28, fontface = "bold", bo
             column_title_side = "top",
             column_names_max_height = unit(6, "cm"),
             column_names_gp = gpar(fontsize = 32, fontface = "bold", col = "black"),
             column_names_rot = 90,
             column_names_centered = TRUE,
             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 = 30, fontface = "bold"))
             },
             right_annotation = c(hb1))
png("Plots/BLT-comparison_order.png", width = 2400, height = 600+45*(nrow(mat.diff)))
draw(ht, padding = unit(c(5, 1, 1, 30), "cm")) # add space for titles
draw(lgd, x = unit(55, "cm"), y = unit(4, "cm"))
dev.off()

tiff("Plots/BLT-comparison_order.png", units="in", width=30, height=5+0.5*(nrow(mat.diff)), pointsize =
draw(ht, padding = unit(c(1.5,1,1,10.5), "in")) # add space for titles
draw(lgd, x = unit(14, "in"), y = unit(1.9, "in"))
dev.off()
```

Create the heatmap