R_microbiome_analysis_final

Vienna

4/25/2020

Summary

The experiment aimed to study the response of honey bee microbiome to three chemicals across host generations and to detect potential microbiome mediated effects on host phenotypes. ## Methods We reared adult bees under controlled lab conditions and inoculated them with a natural bee microbiome (start_pool). Bees were orally stressed with either Tetracycline, Glyphosate, Chloropthalonil or no stressor (control). Three cages per treatment were used. Both, the stress-exposed and control microbiomes were transferred to the next cycle (cage to cage transfer) which was handled as before. The third cycle aims to study effects of the pre-exposed microbiome on phenotypes of naive bee hosts in comparison to control microbiomes. For that we transferred the microbiomes and let them be established in the bee hosts without any stress factor contact. We finally applied high amounts of chemicals to bees with a control microbiome or a pre-exposed microbiome. Bee samples for 16S sequencing has been snap-frozen after cycle 1, cycle 2, cycle 3 BEFORE high stress and cycle 3 AFTER high stress. In addition, the macerated gut pools for transferring microbiome has been saved (start_microbiome as well as pool from each cage to cage transfer). We sequenced the V3-V4 region of the 16S region. Two DNA mock samples from ZymoResearch have been sequenced and named as "positive_control".

Data and metadata

Overview over all experimental variables:

Experimental variables find in metadata file:

sample_type <- microbiome_transfer or single bee *treatment* <- treatment used (control or which toxin) and additional information if microbiome transfer (e.g. Control_transfer), single bee (e.g. Control) *treatment2* <- only treatment, not indicating sample type *cage* <- cage number (three cages per treatment have been used) *treatment_cage* <- combined information of treatment and cage number (e.g. Control_2) *date* <- date of sampling during experiment *cycle* <- experimental cycle (cycle 1, cycle 2, cycle 3 before stress, cycle 3 after stress) *treatment_cycle* <- experimental cycle in combination with treatment information *treatment_cycle2*<- experimental cycle in combination with treatment plus sample type information

Statistics on survival data of bees with pre-exposed microbiomes vs respective controls under high chemical stress for main experiment

```
tetra <- read.table("R_microbiome_data_files/tetra cycle 3 day 5 to 6 survival.txt", header = TRUE)
fisher.test(tetra, alternative = "two.sided")
```

```
##
##  Fisher's Exact Test for Count Data
```

```
##
## data:  tetra
## p-value = 2.573e-05
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##     3.332802 1001.464871
## sample estimates:
## odds ratio
##   23.08462
```

```
chloro <- read.table("R_microbiome_data_files/chloro cycle 3 day 5 to 6 survival.txt", header = TRUE)
fisher.test(chloro, alternative = "two.sided")
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  chloro
## p-value = 0.0259
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.1054762 0.9306751
## sample estimates:
## odds ratio
##  0.3308587
```

```
glypho <- read.table("R_microbiome_data_files/glypho cycle 3 day 5 to 7 survival.txt", header = TRUE)
fisher.test(glypho, alternative = "two.sided")
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  glypho
## p-value = 0.8308
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.4525371 2.9079036
## sample estimates:
## odds ratio
##   1.143036
```

Glyphosate-exposed microbiomes did not significantly affect the survival of bees under high glyphosate stress, chlorothalonil-exposed microbiomes mediated protection and tetracycline-exposed microbiomes lead to higher mortality

Additional chlorothalonil experiments to figure out protective mechanisms of chlorothalonil-exposed microbiomes on bee survival. Statistics on survival data of bees with added filtered pre-exposed gut extract and added chlorothalonil vs respective controls under high chemical stress.

```
survive <- read.table("R_microbiome_data_files/Fisher_test_filtered_Chloro_control_exp.txt", header = TR
fisher.test(survive, alternative = "two.sided")
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  survive
## p-value = 0.03538
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.2701351 0.9670985
## sample estimates:
## odds ratio
##  0.5135861
```

```
survive2 <- read.table("R_microbiome_data_files/Fisher_test_added_Chloro_control_exp.txt", header = TRUE
fisher.test(survive2, alternative = "two.sided")
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  survive2
## p-value = 0.6267
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.2306866 2.0877885
## sample estimates:
## odds ratio
##  0.7177255
```

While filtered pre-exposed gut solution did improve later survival under high chlorothalonil stress, direct addition of chlorothalonil did not.

16S data

```
#read in otu table
otu_table=read.csv("R_microbiome_data_files/Svtab1.csv",sep=",",row.names=1)
otu_table=as.matrix(otu_table)

#read in taxonomy
taxonomy=read.csv("R_microbiome_data_files/taxonomy_modify.csv",sep=",",row.names=1)
taxonomy <- cbind(taxonomy, ASV = paste0("ASV", sprintf("%04d", 1:nrow(taxonomy))))
taxonomy=as.matrix(taxonomy)

metatable <- read.delim("R_microbiome_data_files/metadata_reorder3.txt")
#View(metatable)
row.names(metatable) <- metatable[[1]]
metatable<- metatable[,(-1)]
META <- sample_data(metatable)
```

```
phy_tree <- read_tree("R_microbiome_data_files/rooted_tree.nwk")

#import as phyloseq objects
OTU=otu_table(otu_table,taxa_are_rows=TRUE)
TAX=tax_table(taxonomy)

#create phyloseq object
ps1<- phyloseq(OTU, TAX, META, phy_tree)

# change taxonomy header
colnames(tax_table(ps1)) <- c(D0 = "Kingdom", D1 = "Phylum", D2 = "Class",
                              D3 = "Order", D4 = "Family", D5 = "Genus", D6 = "Species", ASV = "ASV")

#The total number of ASVs in the whole dataset is
length(taxa_names(ps1))
```

```
## [1] 1717
```

```
#get rid of things we do not want
ps1 = subset_taxa(ps1, Kingdom =="Bacteria")
ps1 <- prune_taxa(taxa_sums(ps1) > 0, ps1)
ps1<-subset_taxa(ps1, (Order!="Chloroplast"))
ps1<-subset_taxa(ps1, (Family!="Mitochondria"))
length(taxa_names(ps1))
```

```
## [1] 1167
```

```
#reduces total taxa numbers from 1717 to 1167 (minus 550)

# mean, max and min of sample read counts
smin <- min(sample_sums(ps1))
smean <- mean(sample_sums(ps1))
smax <- max(sample_sums(ps1))
# printing the results
cat("The minimum sample read count is:",smin)
```

```
## The minimum sample read count is: 12351
```

```
cat("The average sample read count is:",smean)
```

```
## The average sample read count is: 29843.03
```

```
cat("The maximum sample read count is:",smax)
```

```
## The maximum sample read count is: 66542
```

Plot rarefaction curves

```
rare<- ps1
rare2 = subset_samples(rare, treatment2 != "positive_control")
smin <- min(sample_sums(rare2))
cat("The minimum sample read count is:",smin)
```

```
## The minimum sample read count is: 12351
```

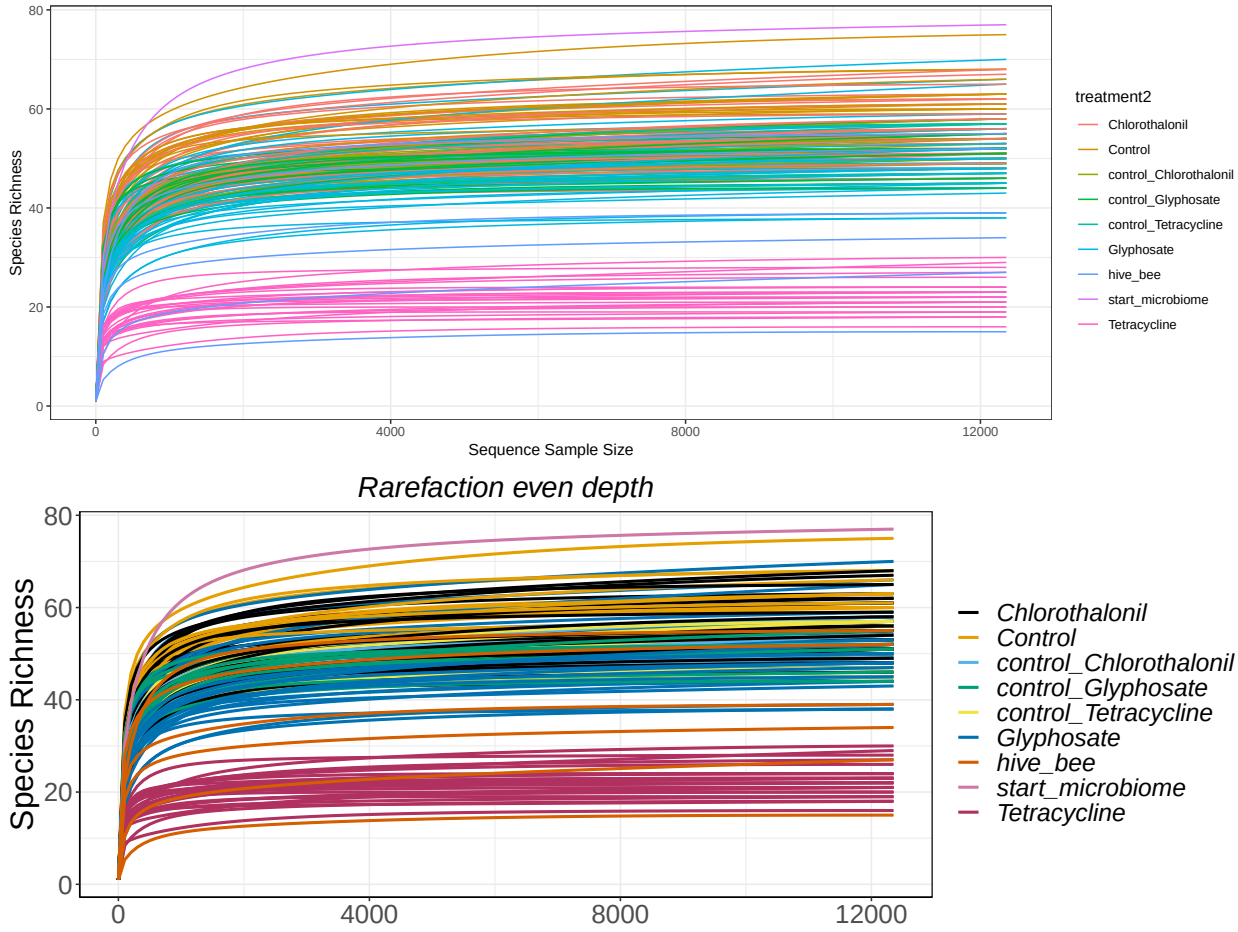
```
# rarefy
set.seed(42)
ps.rarefied = rarefy_even_depth(rare2, rngseed=1, sample.size=12351, replace=F)
library(ranacapa)
cbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7","
ggrare(ps.rarefied, step = 100, se = FALSE, color="treatment2")+ scale_colour_manual(values=cbPalette)+
                legend.background = element_blank(),
                legend.key = element_blank()) + theme(text = element_text(size = 22))+theme(strip.text =
```

```
## rarefying sample X10.3.C1
## rarefying sample X10.3.C2
## rarefying sample X10.3.C3
## rarefying sample X10.3.Ch1
## rarefying sample X10.3.Ch2
## rarefying sample X10.3.Ch3
## rarefying sample X10.3.G1
## rarefying sample X10.3.G2
## rarefying sample X10.3.G3
## rarefying sample X10.3.T1
## rarefying sample X10.3.T2
## rarefying sample X10.3.T3
## rarefying sample X10.C1.1
## rarefying sample X10.C1.2
## rarefying sample X10.C1.6
## rarefying sample X10.C2.1
## rarefying sample X10.C2.4
## rarefying sample X10.C2.5
## rarefying sample X10.C3.2
## rarefying sample X10.C3.3
## rarefying sample X10.C3.4
## rarefying sample X10.Ch1.1
## rarefying sample X10.Ch1.2
## rarefying sample X10.Ch1.3
## rarefying sample X10.Ch1.4
## rarefying sample X10.Ch3.1
## rarefying sample X10.Ch3.2
## rarefying sample X10.Ch3.3
## rarefying sample X10.Ch3.4
## rarefying sample X10.Ch3.5
## rarefying sample X10.G1.1
## rarefying sample X10.G1.2
## rarefying sample X10.G1.5
## rarefying sample X10.G2.2
## rarefying sample X10.G2.4
```

```
## rarefying sample X10.G2.5
## rarefying sample X10.G3.1
## rarefying sample X10.G3.5
## rarefying sample X10.G3.9
## rarefying sample X16.3.C1
## rarefying sample X16.3.C2
## rarefying sample X16.3.C3
## rarefying sample X16.3.Ch1
## rarefying sample X16.3.Ch2
## rarefying sample X16.3.Ch3
## rarefying sample X16.3.G1
## rarefying sample X16.3.G2
## rarefying sample X16.3.G3
## rarefying sample X16.3.T1
## rarefying sample X16.3.T2
## rarefying sample X16.3.T3
## rarefying sample X16.C1.4
## rarefying sample X16.C1.5
## rarefying sample X16.C1.8
## rarefying sample X16.C2.10
## rarefying sample X16.C2.5
## rarefying sample X16.C2.8
## rarefying sample X16.C3.2
## rarefying sample X16.C3.3
## rarefying sample X16.C3.7
## rarefying sample X16.Ch1.4
## rarefying sample X16.Ch1.5
## rarefying sample X16.Ch1.9
## rarefying sample X16.Ch2.2
## rarefying sample X16.Ch2.6
## rarefying sample X16.Ch2.8
## rarefying sample X16.Ch3.2
## rarefying sample X16.Ch3.4
## rarefying sample X16.Ch3.6
## rarefying sample X16.G1.5
## rarefying sample X16.G1.7
## rarefying sample X16.G1.9
## rarefying sample X16.G2.10
## rarefying sample X16.G2.7
## rarefying sample X16.G3.4
## rarefying sample X16.G3.5
## rarefying sample X16.G3.6
## rarefying sample X16.T1.1
## rarefying sample X16.T1.2
## rarefying sample X16.T1.3
## rarefying sample X16.T2.1
## rarefying sample X16.T2.10
## rarefying sample X16.T2.5
## rarefying sample X16.T3.3
## rarefying sample X16.T3.6
## rarefying sample X16.T3.7
## rarefying sample X21.C1.Ch.1
## rarefying sample X21.C1.Ch.2
## rarefying sample X21.C1.Ch.3
```

```
## rarefying sample X21.C1.G.1
## rarefying sample X21.C1.G.3
## rarefying sample X21.C1.G.5
## rarefying sample X21.C1.T.4
## rarefying sample X21.C1.T.5
## rarefying sample X21.C2.Ch.1
## rarefying sample X21.C2.Ch.3
## rarefying sample X21.C2.Ch.6
## rarefying sample X21.C2.G.1
## rarefying sample X21.C2.G.4
## rarefying sample X21.C2.G.6
## rarefying sample X21.C2.T.1
## rarefying sample X21.C2.T.3
## rarefying sample X21.C2.T.6
## rarefying sample X21.C3.Ch.1
## rarefying sample X21.C3.Ch.4
## rarefying sample X21.C3.Ch.5
## rarefying sample X21.C3.G.2
## rarefying sample X21.C3.G.3
## rarefying sample X21.C3.G.5
## rarefying sample X21.C3.T.1
## rarefying sample X21.C3.T.2
## rarefying sample X21.C3.T.5
## rarefying sample X21.C3.T.6
## rarefying sample X21.Ch1.1
## rarefying sample X21.Ch1.3
## rarefying sample X21.Ch1.6
## rarefying sample X21.Ch2.3
## rarefying sample X21.Ch2.4
## rarefying sample X21.Ch2.6
## rarefying sample X21.Ch3.1
## rarefying sample X21.Ch3.4
## rarefying sample X21.Ch3.5
## rarefying sample X21.G1.1
## rarefying sample X21.G1.2
## rarefying sample X21.G1.3
## rarefying sample X21.G2.1
## rarefying sample X21.G2.2
## rarefying sample X21.G2.3
## rarefying sample X21.G3.1
## rarefying sample X21.G3.3
## rarefying sample X21.G3.4
## rarefying sample X21.T1.1
## rarefying sample X21.T1.2
## rarefying sample X21.T1.4
## rarefying sample X21.T2.1
## rarefying sample X21.T2.2
## rarefying sample X21.T2.3
## rarefying sample X21.T3.1
## rarefying sample X21.T3.3
## rarefying sample X21.T3.4
## rarefying sample X22.C1.T.1
## rarefying sample X22.C1.T.2
## rarefying sample X22.C1.T.4
```

```
## rarefying sample X22.C1.T.5
## rarefying sample X22.C2.Ch.1
## rarefying sample X22.C2.Ch.2
## rarefying sample X22.C2.Ch.3
## rarefying sample X22.C3.Ch.1
## rarefying sample X22.C3.Ch.2
## rarefying sample X22.C3.Ch.3
## rarefying sample X22.C3.Ch.4
## rarefying sample X22.C3.T.1
## rarefying sample X22.C3.T.2
## rarefying sample X22.C3.T.4
## rarefying sample X22.C3.T.6
## rarefying sample X22.Ch1.1
## rarefying sample X22.Ch1.2
## rarefying sample X22.Ch1.3
## rarefying sample X22.Ch2.1
## rarefying sample X22.Ch2.2
## rarefying sample X22.Ch2.3
## rarefying sample X22.Ch3.3
## rarefying sample X22.Ch3.5
## rarefying sample X22.Ch3.6
## rarefying sample X22.T2.1
## rarefying sample X23.C1.G.2
## rarefying sample X23.C1.G.4
## rarefying sample X23.C1.G.6
## rarefying sample X23.C2.G.1
## rarefying sample X23.C2.G.3
## rarefying sample X23.C2.G.4
## rarefying sample X23.C3.G.4
## rarefying sample X23.C3.G.6
## rarefying sample X23.C3.G.9
## rarefying sample X23.G1.1
## rarefying sample X23.G1.4
## rarefying sample X23.G1.8
## rarefying sample X23.G2.1
## rarefying sample X23.G2.2
## rarefying sample X23.G2.3
## rarefying sample X23.G3.1
## rarefying sample X23.G3.3
## rarefying sample X23.G3.9
## rarefying sample X28.2.SP
## rarefying sample NB.1
## rarefying sample NB.2
## rarefying sample NB.4
## rarefying sample NB.6
## rarefying sample NB.8
## rarefying sample NB.9
```




```
ggsave("R_microbiome_figures/supp_rarefaction_sub.png", height = 5, width = 12)
```

*** Supplementary figure: Rarefaction curves were used as a qualitative method to estimate the species richness as a function of sequencing depth for all samples. Rarefaction curves reached their asymptotes for all samples, suggesting that saturation in sequencing was achieved.

Positive controls / mock community

The experiment contained two positive control samples with DNA from the ZymoBIOMICS™ Microbial Community Standard (Zymo Research).

Plot taxa in Mock control samples

```
ps_mock <- ps1
ps_mock <- subset_samples(ps_mock, treatment == "positive_control")
ps_mock <- prune_taxa(taxa_sums(ps_mock) > 0, ps_mock)
#transform to proportional data
ps_mock2 <- transform_sample_counts(ps_mock, function(OTU) {OTU / sum(OTU)})
ps_mock2 <- tax_glom(ps_mock2, taxrank = 'Genus')
low1pc_reads <- max(taxa_sums(ps_mock2)) * 1 / 100
low1pc_indices <- which(taxa_sums(ps_mock2) < low1pc_reads)
length(low1pc_indices)
```

```
## [1] 2
```

```
taxa_names(ps_mock2)[low1pc_indices[1]] <- "other"
ps_merge <- merge_taxa(ps_mock2, low1pc_indices, "other")
tax_table(ps_merge)["other", ] <- c("other")

getPalette = colorRampPalette(brewer.pal(10, "Set3"))
speciesList = unique(tax_table(ps_merge)[,"Genus"])
speciesPalette = getPalette(length(speciesList))
names(speciesPalette) = speciesList

ps_df <- psmelt(ps_merge)

ps_df_sum <- ps_df %>%
  group_by(Sample,Genus) %>%
  summarise(Abundance = mean(Abundance)) %>%
  group_by(Sample) %>%
  mutate(Relative_abundance = Abundance / sum(Abundance)) %>%
  ungroup() %>%
  mutate(Genus = fct_reorder(Genus, Abundance, .fun = sum, .desc = TRUE)) %>%
  droplevels()
levels(ps_df_sum$Genus)
```

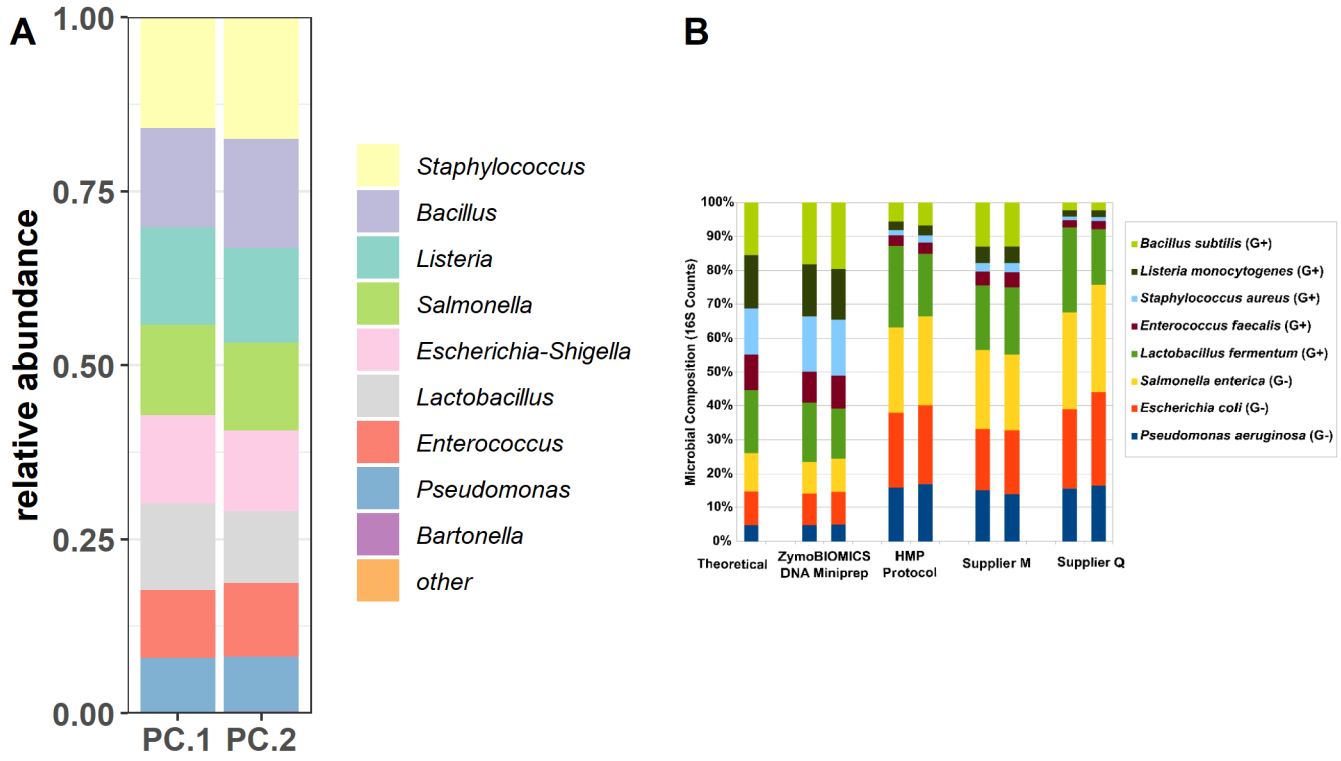
```
##  [1] "Staphylococcus"       "Bacillus"             "Listeria"
##  [4] "Salmonella"           "Escherichia-Shigella" "Lactobacillus"
##  [7] "Enterococcus"         "Pseudomonas"          "Bartonella"
## [10] "other"
```

```
mock_plot<-ggplot(data = ps_df_sum, aes(x = Sample, y = Relative_abundance, fill = Genus)) +
  geom_bar(stat = "identity")+ scale_y_continuous(expand = c(0,0))+ scale_fill_manual(values= speciesPal
#mock_plot<-mock_plot+labs(title="Mock samples")
mock_plot<-mock_plot+ theme(legend.text=element_text(size=9,face="italic"))
mock_plot<-mock_plot+theme(axis.text.x = element_text(size=12, face="bold"))+ theme(axis.title.x=elemen
mock_plot2<-mock_plot+theme(plot.title = element_text(face = "bold", color="grey40",size = (17)))+theme
ps_df_sum
```

```
## # A tibble: 20 x 4
##    Sample Genus                Abundance Relative_abundance
##    <chr>  <fct>                    <dbl>              <dbl>
##  1 PC.1   Bacillus               0.142              0.142
##  2 PC.1   Bartonella             0.00145            0.00145
##  3 PC.1   Enterococcus           0.0970             0.0970
##  4 PC.1   Escherichia-Shigella   0.127              0.127
##  5 PC.1   Lactobacillus          0.124              0.124
##  6 PC.1   Listeria               0.140              0.140
##  7 PC.1   other                  0.000420           0.000420
##  8 PC.1   Pseudomonas            0.0777             0.0777
##  9 PC.1   Salmonella             0.130              0.130
## 10 PC.1   Staphylococcus         0.159              0.159
## 11 PC.2   Bacillus               0.156              0.156
## 12 PC.2   Bartonella             0.00223            0.00223
## 13 PC.2   Enterococcus           0.106              0.106
## 14 PC.2   Escherichia-Shigella   0.116              0.116
```

```
## 15 PC.2   Lactobacillus          0.103              0.103
## 16 PC.2   Listeria               0.136              0.136
## 17 PC.2   other                  0.000488           0.000488
## 18 PC.2   Pseudomonas            0.0787             0.0787
## 19 PC.2   Salmonella             0.127              0.127
## 20 PC.2   Staphylococcus         0.175              0.175
```

```
mock_comp <- "R_microbiome_data_files/mock_composition_ZymoResearch.png"
figure_S1 <- "R_microbiome_figures/Supp_mock_composition_comparison.png"
png(figure_S1, 7 * plot_res, 4 * plot_res, res = plot_res)
ggarrange(mock_plot2, rasterGrob(readPNG(mock_comp)),labels = c("A", "B"))
invisible(dev.off())
knitr::include_graphics(figure_S1, dpi = plot_res)
```



All 8 Mock bacterial taxa correctly detected. **Figure S1: Comparison of the mock community with the theoretical proportions.** The two mock samples sequenced in this study **(A)** show no big difference to the expected theoretical proportions of the reference **(B, bar with label *theoretical*)**. The three other bacterial taxa belong to honey bee core microbiome (Bartonella, Gilliamella and Snodgrassella). The relative abundance of these three taxa across the two Mock samples account for 0.23% of all reads, representing neglectable cross-contamination during sequencing.

comparing difference between methods

We sequenced to types of samples: whole bee abdomen and the mix of three macerated guts for cage to cage transfers after each cycle Therefore, we need to test if these samples are different due to the differences in the methods prior extracting before deciding if we include all or not. Use PERMANOVA on bray-curtis dissimilarities using proportional transformed abundance data

```r
set.seed(42)
methods = ps1
methods <- transform_sample_counts(methods, function(OTU) {OTU / sum(OTU)})

Control = subset_samples(methods, treatment2 == "Control")
Control <- prune_taxa(taxa_sums(Control) > 0, Control)
C_metadata <- as(sample_data(Control), "data.frame")
adonis(distance(Control, method="bray") ~ sample_type,data = C_metadata,perm=999)
```

```
##
## Call:
## adonis(formula = distance(Control, method = "bray") ~ sample_type,      data = C_metadata, permutati
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##             Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## sample_type  1   0.05335 0.053349 0.74896 0.03292  0.516
## Residuals   22   1.56707 0.071231         0.96708
## Total       23   1.62042                  1.00000
```

```r
Tetra = subset_samples(methods, treatment2 == "Tetracycline")
Tetra <- prune_taxa(taxa_sums(Tetra) > 0, Tetra)
T_metadata <- as(sample_data(Tetra), "data.frame")
adonis(distance(Tetra, method="bray") ~ sample_type,data = T_metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(Tetra, method = "bray") ~ sample_type,      data = T_metadata, permutations
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##             Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## sample_type  1   0.05033 0.050334 0.71685 0.03023  0.549
## Residuals   23   1.61498 0.070216         0.96977
## Total       24   1.66531                  1.00000
```

```r
Glypho = subset_samples(methods, treatment2 == "Glyphosate")
Glypho <- prune_taxa(taxa_sums(Glypho) > 0, Glypho)
G_metadata <- as(sample_data(Glypho), "data.frame")
adonis(distance(Glypho, method="bray") ~ sample_type,data = G_metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(Glypho, method = "bray") ~ sample_type,      data = G_metadata, permutatio
##
```

```
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##             Df SumsOfSqs  MeanSqs F.Model   R2 Pr(>F)
## sample_type  1    0.0709 0.070933 0.79591 0.02  0.541
## Residuals   39    3.4757 0.089122         0.98
## Total       40    3.5467                 1.00
```

```
Chloro = subset_samples(methods, treatment2 == "Chlorothalonil")
Chloro <- prune_taxa(taxa_sums(Chloro) > 0, Chloro)
Ch_metadata <- as(sample_data(Chloro), "data.frame")
adonis(distance(Chloro, method="bray") ~ sample_type,data = Ch_metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(Chloro, method = "bray") ~ sample_type,      data = Ch_metadata, permutati
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##             Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## sample_type  1    0.0715 0.071501  1.2362 0.02998  0.253
## Residuals   40    2.3137 0.057842         0.97002
## Total       41    2.3852                 1.00000
```

No significant difference between gut pools or whole bees in any treatment.

Alpha diversity

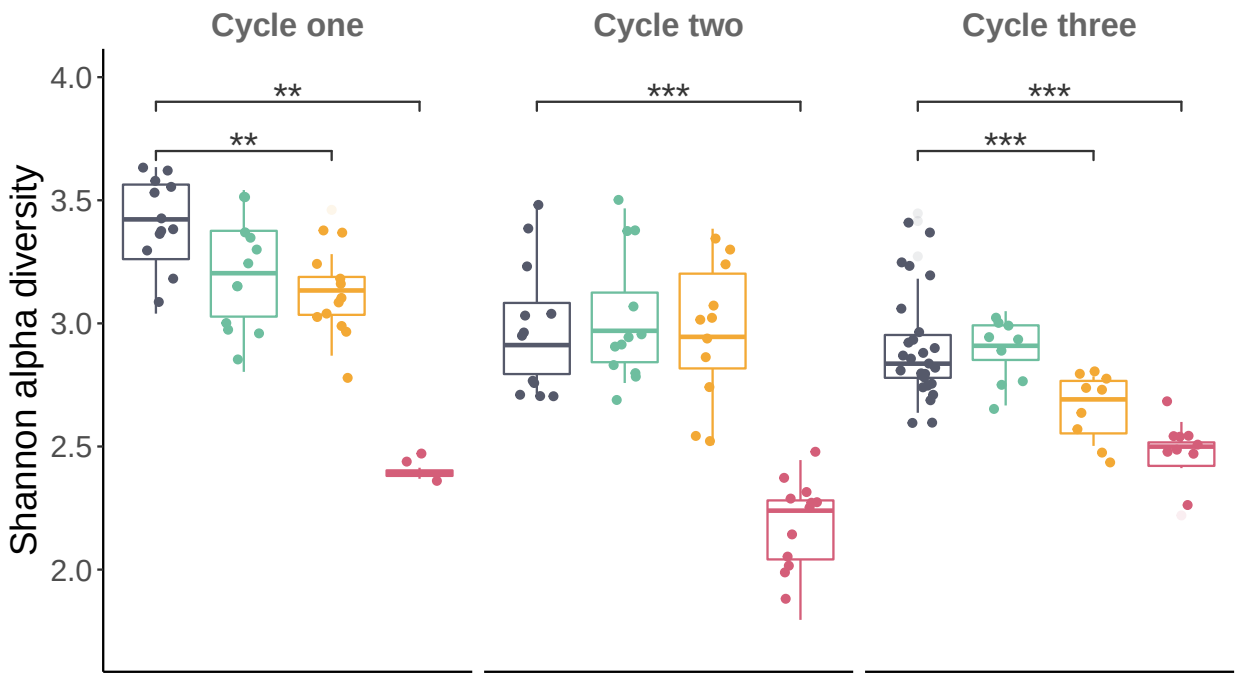
```
alpha<- ps1
#pick cycle 1, 2 and 3 before stress
alpha1 = subset_samples(alpha, cycle=="cycle_one" | cycle=="cycle_two" | cycle == "cycle_three_before_s
#rarefy to even numbers
set.seed(1)
alpha1_ra <- rarefy_even_depth(alpha1,sample.size=12351, replace=FALSE, rngseed = 1)
alpha1_ra<-prune_taxa(taxa_sums(alpha1_ra) > 0, alpha1_ra)

sample_data(alpha1_ra)$cycle<-factor(sample_data(alpha1_ra)$cycle,levels=c("cycle_one","cycle_two","cycl
levels(sample_data(alpha1_ra)$cycle)
```

```
## [1] "Cycle one"   "Cycle two"   "Cycle three"
```

```
sample_data(alpha1_ra)$treatment3<-factor(sample_data(alpha1_ra)$treatment3,levels=c("Control","Chlorot
levels(sample_data(alpha1_ra)$treatment3)
```

```
## [1] "Control"        "Chlorothalonil" "Glyphosate"     "Tetracycline"
```

```
sigFunc = function(x){
  if(x < 0.001){"***"}
  else if(x < 0.01){"**"}
  else if(x < 0.05){"*"}
  else{NA}}

alpha_meas = c("Shannon")
p2 <- plot_richness(alpha1_ra,"treatment3",color="treatment3", measures=alpha_meas)
p2$layers <- p2$layers[-1]
p2<-p2 + geom_boxplot(data=p2$data, aes(x=treatment3, y=value),show_guide=FALSE, alpha=0.1)+geom_point(p
  geom_signif(comparisons=list(c("Control", "Chlorothalonil"), c("Control", "Glyphosate"),c("Control", "
p2 <- p2 + guides(fill=guide_legend(title=element_blank()))+ylim(1.7,4)
p2 <- p2 + theme(plot.title = element_text(hjust = 0.5))+scale_color_manual(values=c("#55596a", "#6ebe9
p2 <- p2+ theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+ theme(panel.ba
p2 <- p2+theme(axis.line = element_line(colour = "black"))
p2<- p2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = element_
p2 <- p2+ theme(panel.border = element_blank())
p2<-p2 + theme(text = element_text(size = 17))+theme(strip.text = element_text(face="bold", size=15, col
Shannon2<-p2+ facet_grid( ~cycle, scales="free_x",space="free")+theme(axis.ticks.x=element_blank())+ther
Shannon2
```




```
ggsave("R_microbiome_figures/alpha_Shannon.png", height = 3.5, width = 6.5)
```

***The Shannon index captures information about both the species richness (number of species) and the relative abundances of the species. Increased Shannon index means increased species diversity.

make statistical comparisons on Shannon alpha diversity index - comparing treatments against respective controls in each cycle

```
alpha<- ps1
alpha1 = subset_samples(alpha, cycle=="cycle_one" | cycle=="cycle_two" | cycle == "cycle_three_before_s
#rarefy to even numbers
set.seed(1)
alpha1_ra <- rarefy_even_depth(alpha1,sample.size=12351, replace=FALSE, rngseed = 1)

results = estimate_richness(alpha1_ra, measures = 'Shannon')
d = sample_data(alpha1_ra)

# calculate wilcox-test
Control_1 = results[d[,'treatment_cycle3'] == 'Control_cycle_1',]
Tetracycline_1 = results[d[,'treatment_cycle3'] == 'Tetracycline_cycle_1',]
Glyphosate_1 = results[d[,'treatment_cycle3'] == 'Glyphosate_cycle_1',]
Chlorothalonil_1 = results[d[,'treatment_cycle3'] == 'Chlorothalonil_cycle_1',]
wilcox.test(Control_1, Chlorothalonil_1)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Control_1 and Chlorothalonil_1
## W = 105, p-value = 0.05966
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(Control_1, Glyphosate_1)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Control_1 and Glyphosate_1
## W = 122, p-value = 0.002914
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(Control_1, Tetracycline_1)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Control_1 and Tetracycline_1
## W = 36, p-value = 0.004396
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.0596,0.002914,0.004396)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.059600 0.006594 0.006594
```

```r
#cycle 2
Control_2 = results[d[,'treatment_cycle3'] == 'Control_cycle_2',]
Tetracycline_2 = results[d[,'treatment_cycle3'] == 'Tetracycline_cycle_2',]
Glyphosate_2 = results[d[,'treatment_cycle3'] == 'Glyphosate_cycle_2',]
Chlorothalonil_2 = results[d[,'treatment_cycle3'] == 'Chlorothalonil_cycle_2',]
wilcox.test(Control_2, Chlorothalonil_2)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Control_2 and Chlorothalonil_2
## W = 64, p-value = 0.6707
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(Control_2, Glyphosate_2)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Control_2 and Glyphosate_2
## W = 64, p-value = 0.9279
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(Control_2, Tetracycline_2)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Control_2 and Tetracycline_2
## W = 144, p-value = 7.396e-07
## alternative hypothesis: true location shift is not equal to 0
```

```r
pvalues<-c(0.671,0.9279,7.396e-07)
p.adjust(pvalues,method="fdr")
```

```
## [1] 9.2790e-01 9.2790e-01 2.2188e-06
```

```r
#cycle 3
control_cycle_3_before_stress = results[d[,'treatment_cycle3'] == 'control_cycle_3_before_stress',]
Tetracycline_cycle_3_before_stress = results[d[,'treatment_cycle3'] == 'Tetracycline_cycle_3_before_str
Glyphosate_cycle_3_before_stress = results[d[,'treatment_cycle3'] == 'Glyphosate_cycle_3_before_stress'
Chlorothalonil_cycle_3_before_stress = results[d[,'treatment_cycle3'] == 'Chlorothalonil_cycle_3_before_
wilcox.test(control_cycle_3_before_stress, Chlorothalonil_cycle_3_before_stress)
```

```
##
##  Wilcoxon rank sum test
##
## data:  control_cycle_3_before_stress and Chlorothalonil_cycle_3_before_stress
## W = 108, p-value = 0.6406
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(control_cycle_3_before_stress, Glyphosate_cycle_3_before_stress)
```

```
##
##  Wilcoxon rank sum test
##
## data:  control_cycle_3_before_stress and Glyphosate_cycle_3_before_stress
## W = 213, p-value = 0.0003847
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(control_cycle_3_before_stress, Tetracycline_cycle_3_before_stress)
```

```
##
##  Wilcoxon rank sum test
##
## data:  control_cycle_3_before_stress and Tetracycline_cycle_3_before_stress
## W = 243, p-value = 2.124e-08
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.641,0.000385,2.124e-08)
p.adjust(pvalues,method="fdr")
```

```
## [1] 6.410e-01 5.775e-04 6.372e-08
```

Under tetracycline the Shannon alpha diversity was significantly affected in all three cycles. Glyphosate significantly affected the alpha diversity in cycle 1 and 3 and chlorothalonil did not had any significant effect.

observed species numbers - alpha diversity

```
alpha<- ps1
alpha1 = subset_samples(alpha, treatment != "hive_bee" & treatment != "positive_control"& treatment != "
#rarefy to even numbers
smin <- min(sample_sums(alpha1))
cat("The minimum sample read count is:",smin)
```

```
## The minimum sample read count is: 12351
```

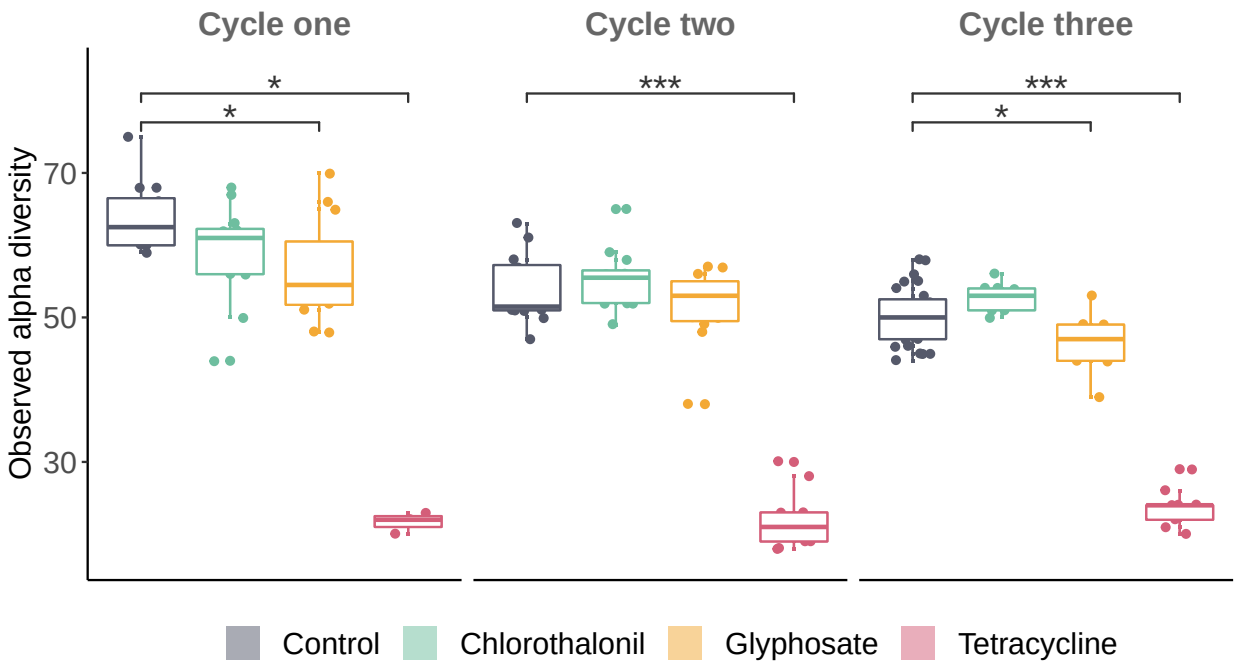
```
set.seed(1)
alpha1_ra <- rarefy_even_depth(alpha1,sample.size=12351, replace=FALSE, rngseed = 1)

sample_data(alpha1_ra)$cycle<-factor(sample_data(alpha1_ra)$cycle,levels=c("cycle_one","cycle_two","cycl
levels(sample_data(alpha1_ra)$cycle)
```

```
## [1] "Cycle one"   "Cycle two"   "Cycle three"
```

```
sample_data(alpha1_ra)$treatment3<-factor(sample_data(alpha1_ra)$treatment3,levels=c("Control","Chlorotl
levels(sample_data(alpha1_ra)$treatment3)
```

```
## [1] "Control"        "Chlorothalonil" "Glyphosate"     "Tetracycline"
```

```
sigFunc = function(x){
  if(x < 0.001){"***"}
  else if(x < 0.01){"**"}
  else if(x < 0.05){"*"}
  else{NA}}

alpha_meas = c("Observed")
p2 <- plot_richness(alpha1_ra,"treatment3",color="treatment3", measures=alpha_meas)
p2$layers <- p2$layers[-1]
p2<-p2 + geom_boxplot(data=p2$data, aes(x=treatment3, y=value),show_guide=FALSE, alpha=0.1)+geom_point(
  geom_signif(comparisons=list(c("Control", "Chlorothalonil"), c("Control", "Glyphosate"),c("Control",

p2 <- p2 + guides(fill=guide_legend(title=element_blank()))+ylim(17,84)
p2 <- p2 + theme(plot.title = element_text(hjust = 0.5))+scale_color_manual(values=c("#55596a", "#6ebe9
p2 <- p2+ theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+ theme(panel.ba
p2 <- p2+theme(axis.line = element_line(colour = "black"))
p2<- p2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = element_
p2 <- p2+ theme(panel.border = element_blank())
p2<-p2 + theme(text = element_text(size = 17))+theme(strip.text = element_text(face="bold", size=15, col
p2<-p2+ facet_grid( ~cycle,scales="free_x",space="free")+theme(axis.ticks.x=element_blank())+theme(leger
Observed<-p2+ geom_point(size = -1, shape = 15,fill="treatment3") +geom_boxplot(show.legend = FALSE)+gu
Observed
```




```
ggsave("R_microbiome_figures/Supp_alpha_Observed.png", height = 3.5, width = 6.5)
```

*** Supplementary figure Observed alpha diversity The observed plot shows the difference in detected OTUs.

make statistical comparisons on Observed species numbers - comparing treatments against respective controls in each cycle

```
alpha<- ps1
alpha1 = subset_samples(alpha, cycle=="cycle_one" | cycle=="cycle_two" | cycle == "cycle_three_before_s
#rarefy to even numbers
set.seed(1)
alpha1_ra <- rarefy_even_depth(alpha1,sample.size=12351, replace=FALSE, rngseed = 1)

results = estimate_richness(alpha1_ra, measures = 'Observed')
d = sample_data(alpha1_ra)

# calculate wilcox-test
Control_1 = results[d,'treatment_cycle3'] == 'Control_cycle_1',]
Tetracycline_1 = results[d,'treatment_cycle3'] == 'Tetracycline_cycle_1',]
Glyphosate_1 = results[d,'treatment_cycle3'] == 'Glyphosate_cycle_1',]
Chlorothalonil_1 = results[d,'treatment_cycle3'] == 'Chlorothalonil_cycle_1',]
wilcox.test(Control_1, Chlorothalonil_1)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Control_1 and Chlorothalonil_1
## W = 98, p-value = 0.1391
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(Control_1, Glyphosate_1)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Control_1 and Glyphosate_1
## W = 116, p-value = 0.01184
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(Control_1, Tetracycline_1)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Control_1 and Tetracycline_1
## W = 36, p-value = 0.0111
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.139,0.0184,0.0111)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.1390 0.0276 0.0276
```

```r
#cycle 2
Control_2 = results[d[,'treatment_cycle3'] == 'Control_cycle_2',]
Tetracycline_2 = results[d[,'treatment_cycle3'] == 'Tetracycline_cycle_2',]
Glyphosate_2 = results[d[,'treatment_cycle3'] == 'Glyphosate_cycle_2',]
Chlorothalonil_2 = results[d[,'treatment_cycle3'] == 'Chlorothalonil_cycle_2',]
wilcox.test(Control_2, Chlorothalonil_2)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Control_2 and Chlorothalonil_2
## W = 50.5, p-value = 0.2218
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(Control_2, Glyphosate_2)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Control_2 and Glyphosate_2
## W = 78, p-value = 0.4769
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(Control_2, Tetracycline_2)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Control_2 and Tetracycline_2
## W = 144, p-value = 3.449e-05
## alternative hypothesis: true location shift is not equal to 0
```

```r
pvalues<-c(0.222,0.477,3.449e-05)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.33300000 0.47700000 0.00010347
```

```r
#cycle 3
control_cycle_3_before_stress = results[d[,'treatment_cycle3'] == 'control_cycle_3_before_stress',]
Tetracycline_cycle_3_before_stress = results[d[,'treatment_cycle3'] == 'Tetracycline_cycle_3_before_str
Glyphosate_cycle_3_before_stress = results[d[,'treatment_cycle3'] == 'Glyphosate_cycle_3_before_stress'
Chlorothalonil_cycle_3_before_stress = results[d[,'treatment_cycle3'] == 'Chlorothalonil_cycle_3_before_
wilcox.test(control_cycle_3_before_stress, Chlorothalonil_cycle_3_before_stress)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  control_cycle_3_before_stress and Chlorothalonil_cycle_3_before_stress
## W = 72, p-value = 0.07223
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(control_cycle_3_before_stress, Glyphosate_cycle_3_before_stress)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  control_cycle_3_before_stress and Glyphosate_cycle_3_before_stress
## W = 178.5, p-value = 0.03822
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(control_cycle_3_before_stress, Tetracycline_cycle_3_before_stress)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  control_cycle_3_before_stress and Tetracycline_cycle_3_before_stress
## W = 243, p-value = 9.364e-06
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.072,0.038,9.364e-06)
p.adjust(pvalues,method="fdr")
```

```
## [1] 7.2000e-02 5.7000e-02 2.8092e-05
```

Numbers of observed species is significantly different under tetracycline to the control in all three cycles, while glyphosate affected the observed species number significantly in cycle 1 and chlorothalonil had no significant effect at any time point.

Beta diversity

ordination plots

```
ord2<- ps1
ord2 = subset_samples(ord2, cycle=="cycle_one" | cycle=="cycle_two" | cycle == "cycle_three_before_stres

ord2 <- transform_sample_counts(ord2, function(OTU) {OTU / sum(OTU)})
sample_data(ord2)$cycle<-factor(sample_data(ord2)$cycle,levels=c("cycle_one","cycle_two", "cycle_three_
levels(sample_data(ord2)$cycle)
```

```
## [1] "Cycle one"   "Cycle two"   "Cycle three"
```

```
sample_data(ord2)$treatment3<-factor(sample_data(ord2)$treatment3,levels=c("Control","Chlorothalonil","
levels(sample_data(ord2)$treatment3)
```

```
## [1] "Control"        "Chlorothalonil" "Glyphosate"     "Tetracycline"
```

```r
ord2 <- prune_taxa(taxa_sums(ord2) > 0, ord2)
set.seed(42)
dist<-phyloseq::distance(ord2, method="bray")
ordination = ordinate(ord2, method="NMDS", distance=dist)
```

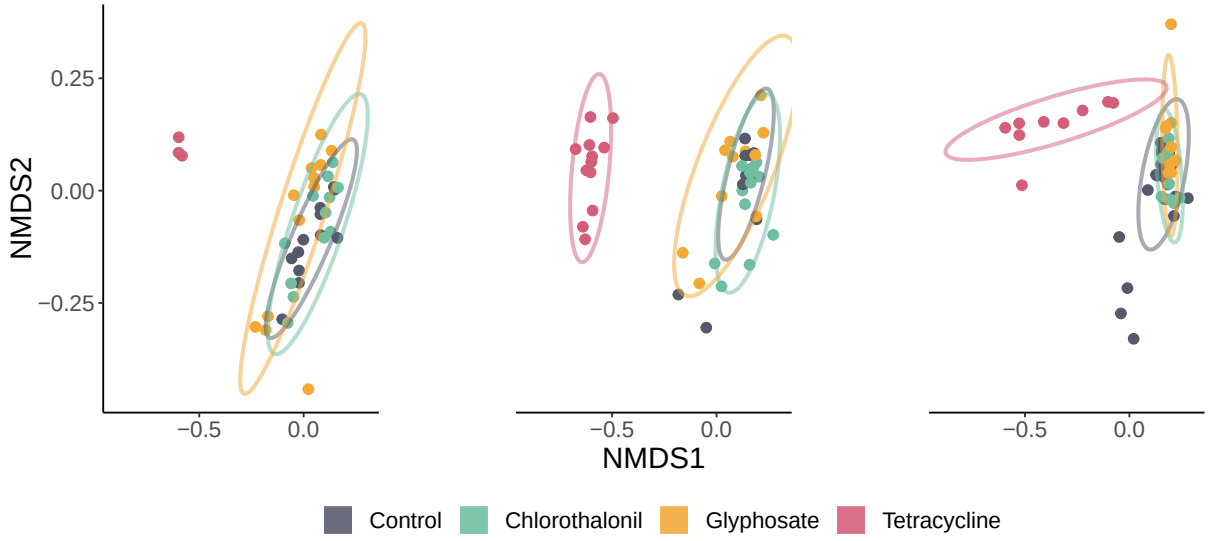
```
## Run 0 stress 0.09856789
## Run 1 stress 0.100735
## Run 2 stress 0.09968012
## Run 3 stress 0.1006036
## Run 4 stress 0.09813058
## ... New best solution
## ... Procrustes: rmse 0.01377179  max resid 0.110288
## Run 5 stress 0.09856256
## ... Procrustes: rmse 0.01394676  max resid 0.1104272
## Run 6 stress 0.09908369
## Run 7 stress 0.09999148
## Run 8 stress 0.1108652
## Run 9 stress 0.1103568
## Run 10 stress 0.1003239
## Run 11 stress 0.09813093
## ... Procrustes: rmse 7.874533e-05  max resid 0.0008520147
## ... Similar to previous best
## Run 12 stress 0.1104334
## Run 13 stress 0.0996829
## Run 14 stress 0.1007221
## Run 15 stress 0.09817203
## ... Procrustes: rmse 0.01620558  max resid 0.1119934
## Run 16 stress 0.09813176
## ... Procrustes: rmse 0.0001382161  max resid 0.001490884
## ... Similar to previous best
## Run 17 stress 0.1125378
## Run 18 stress 0.09813098
## ... Procrustes: rmse 9.547599e-05  max resid 0.0003116982
## ... Similar to previous best
## Run 19 stress 0.09967974
## Run 20 stress 0.09813072
## ... Procrustes: rmse 3.142913e-05  max resid 9.177655e-05
## ... Similar to previous best
## *** Solution reached
```

```r
cat("stress is:", ordination$stress)
```

```
## stress is: 0.09813058
```

```r
p2 <- plot_ordination(ord2, ordination, color="treatment3") +geom_point(size=2.5)
p2 <- p2 + guides(fill=guide_legend(title=element_blank()))
p2 <- p2 + theme(plot.title = element_text(hjust = 0.5))+scale_color_manual(values=c("#55596a", "#6ebe9
p2 <- p2+ theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+ theme(panel.ba
p2 <- p2+theme(axis.line = element_line(colour = "black"))
p2<- p2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = element
p2 <- p2+ theme(panel.border = element_blank())
p2<-p2 + theme(text = element_text(size = 16.5))+theme(strip.text = element_blank())
```

```
#+theme(strip.text = element_text(face="bold", size=14, color="gray40"))+ theme(axis.title.x=element_bl
p2<-p2+ facet_wrap( ~cycle,scales="free_x")+theme(legend.position="bottom")+theme(strip.background =eler
NMDS_main<-p2+stat_ellipse(aes(group = treatment3),size=1.1, alpha=0.5)+ theme(panel.spacing.x=unit(5.5
NMDS_main
```




```
ggsave("R_microbiome_figures/NMDS_man.png", height = 5, width = 10)
```

```
ord2<- ps1
ord2 = subset_samples(ord2, cycle=="cycle_one" | cycle=="cycle_two" | cycle == "cycle_three_before_stres
ord2 <- transform_sample_counts(ord2, function(OTU) {OTU / sum(OTU)})
sample_data(ord2)$cycle<-factor(sample_data(ord2)$cycle,levels=c("cycle_one","cycle_two", "cycle_three_
levels(sample_data(ord2)$cycle)
```

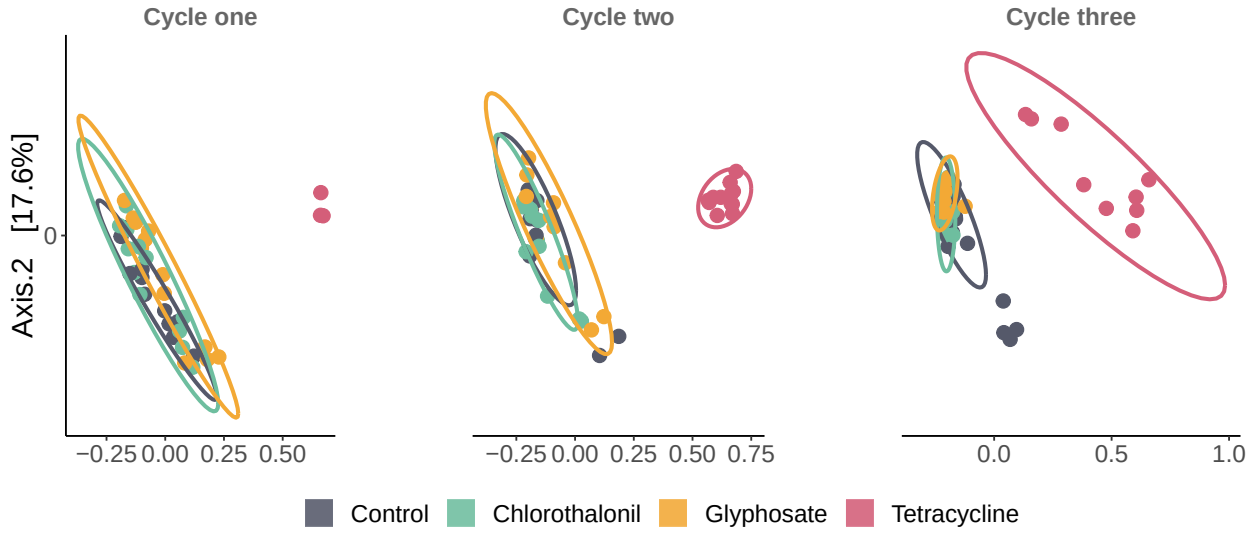
```
## [1] "Cycle one"   "Cycle two"   "Cycle three"
```

```
sample_data(ord2)$treatment3<-factor(sample_data(ord2)$treatment3,levels=c("Control","Chlorothalonil","G
levels(sample_data(ord2)$treatment3)
```

```
## [1] "Control"        "Chlorothalonil" "Glyphosate"     "Tetracycline"
```

```
set.seed(42)
dist<-phyloseq::distance(ord2, method="bray")
ordination = ordinate(ord2, method="PCoA", distance=dist)
p2 <- plot_ordination(ord2, ordination, color="treatment3") +geom_point(size=3.5)+scale_colour_manual(va

p2 <- p2 + guides(fill=guide_legend(title=element_blank()))
p2 <- p2 + theme(plot.title = element_text(hjust = 0.5))+scale_color_manual(values=c("#55596a", "#6ebe9f
p2 <- p2+ theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+ theme(panel.bac
p2 <- p2+theme(axis.line = element_line(colour = "black"))
p2<- p2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = element_
p2 <- p2+ theme(panel.border = element_blank())
p2<-p2 + theme(text = element_text(size = 17))+theme(strip.text = element_text(face="bold", size=14, col
```

```
p2<-p2+ facet_grid( ~cycle,scales="free_x",space="free")+theme(legend.position="bottom")+theme(strip.ba
p2<-p2+stat_ellipse(aes(group = treatment3),size=1.1)+ theme(panel.spacing.x=unit(5.5, "lines"))
p2+scale_y_continuous(breaks = seq(0.0, 0.5))
```




```
ggsave("R_microbiome_figures/PCoA_man.png", height = 5, width = 9)
```

Beta diversity stats

test difference between treatments and respective controls in the three cycles to statistically verify microbial community compositional difference seen in ordination plots

```
set.seed(42)
compare=ps1
compare1 <- transform_sample_counts(compare, function(OTU) {OTU / sum(OTU)})

cycle1 = subset_samples(compare1, cycle == "cycle_one")
cycle1 <- prune_taxa(taxa_sums(cycle1) > 0, cycle1)
cycle1.1 <- as(sample_data(cycle1), "data.frame")

cycle2 = subset_samples(compare1, cycle == "cycle_two")
cycle2 <- prune_taxa(taxa_sums(cycle2) > 0, cycle2)
cycle2.1 <- as(sample_data(cycle2), "data.frame")

cycle3b = subset_samples(compare1, cycle == "cycle_three_before_stress")
cycle3b <- prune_taxa(taxa_sums(cycle3b) > 0, cycle3b)
cycle3b.1 <- as(sample_data(cycle3b), "data.frame")

#cycle 1

subs <- subset_samples(cycle1, treatment2 %in% c("Control", "Chlorothalonil"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment2,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment2,      data = metadata, permutations = 9
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment2  1   0.07746 0.077456  1.3139 0.05636  0.211
## Residuals  22   1.29693 0.058951         0.94364
## Total      23   1.37439                  1.00000
```

```
subs <- subset_samples(cycle1, treatment2 %in% c("Control", "Glyphosate"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment2, data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment2,      data = metadata, permutations = 9
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment2  1   0.23951 0.239510  3.1008 0.12353  0.022 *
## Residuals  22   1.69932 0.077242         0.87647
## Total      23   1.93883                  1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
subs <- subset_samples(cycle1, treatment2 %in% c("Control", "Tetracycline"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment2,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment2,      data = metadata, permutations = 9
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## treatment2  1   1.39916 1.39916  31.186 0.70579  0.004 **
## Residuals  13   0.58325 0.04487         0.29421
## Total      14   1.98241                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
pvalues <- c(0.21,0.022,0.004)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.210 0.033 0.012
```

```r
#cycle 2
subs <- subset_samples(cycle2, treatment2 %in% c("Control", "Chlorothalonil"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment2,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment2,      data = metadata, permutations = 9
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment2  1   0.04865 0.048654  0.7824 0.03434  0.492
## Residuals  22   1.36809 0.062186         0.96566
## Total      23   1.41675                  1.00000
```

```r
subs <- subset_samples(cycle2, treatment2 %in% c("Control", "Glyphosate"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment2,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment2,      data = metadata, permutations = 9
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment2  1   0.14948 0.149477  1.9993 0.08693  0.079 .
## Residuals  21   1.57005 0.074764         0.91307
## Total      22   1.71952                  1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
subs <- subset_samples(cycle2, treatment2 %in% c("Control", "Tetracycline"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment2,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment2,      data = metadata, permutations = 9
```

```
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## treatment2  1    3.6437  3.6437  62.591 0.73992  0.001 ***
## Residuals  22    1.2807  0.0582         0.26008
## Total      23    4.9244                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
pvalues <- c(0.5,0.08,0.001)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.500 0.120 0.003
```

```
#cycle 3 before stress

subs <- subset_samples(cycle3b, treatment3 %in% c("Control", "Chlorothalonil"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment3,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment3,      data = metadata, permutations = 9
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment3  1   0.07295 0.072949  1.5559 0.04376  0.158
## Residuals  34   1.59406 0.046884         0.95624
## Total      35   1.66700                 1.00000
```

```
subs <- subset_samples(cycle3b, treatment3 %in% c("Control", "Glyphosate"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment3,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment3,      data = metadata, permutations = 9
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
```

```
## treatment3  1   0.25456 0.254564  4.9932 0.12805  0.002 **
## Residuals  34   1.73338 0.050982         0.87195
## Total      35   1.98795                  1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
subs <- subset_samples(cycle3b, treatment3 %in% c("Control", "Tetracycline"))
metadata <- as(sample_data(subs), "data.frame")
adonis(distance(subs, method="bray") ~ treatment3,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs, method = "bray") ~ treatment3,      data = metadata, permutations = 9
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## treatment3  1    2.5924 2.59238  46.491 0.57759  0.001 ***
## Residuals  34    1.8959 0.05576         0.42241
## Total      35    4.4882                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
pvalues <- c(0.158,0.002,0.001)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.158 0.003 0.003
```

Community composition was significantly affected under tetracycline in all cycles. Glyphosate affected it in cycle 1 and 3 while chlorothalonil did not show to cause significant changes

Test general effects of treatment across cycles

```
set.seed(42)
compare=ps1
compare <- prune_taxa(taxa_sums(compare) > 0, compare)
compare <- transform_sample_counts(compare, function(OTU) {OTU / sum(OTU)})
compare.1 <- as(sample_data(compare), "data.frame")

cycle1 = subset_samples(compare, cycle == "cycle_one")
cycle1 <- prune_taxa(taxa_sums(cycle1) > 0, cycle1)
cycle1.2 <- as(sample_data(cycle1), "data.frame")

cycle2 = subset_samples(compare, cycle == "cycle_two")
cycle2 <- prune_taxa(taxa_sums(cycle2) > 0, cycle2)
cycle2.2 <- as(sample_data(cycle2), "data.frame")

cycle3_before_stress = subset_samples(compare, cycle == "cycle_three_before_stress")
```

```
cycle3_before_stress <- prune_taxa(taxa_sums(cycle3_before_stress) > 0, cycle3_before_stress)
cycle3_before_stress.2 <- as(sample_data(cycle3_before_stress), "data.frame")

d = distance(compare, "bray")
adonis(d ~ treatment3, compare.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ treatment3, data = compare.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##             Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## treatment3   9    15.696 1.74397   24.73 0.55014  0.001 ***
## Residuals  182    12.835 0.07052         0.44986
## Total      191    28.531                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
d1 = distance(cycle1, "bray")
adonis(d1 ~ treatment3, cycle1.2, perm=999)
```

```
##
## Call:
## adonis(formula = d1 ~ treatment3, data = cycle1.2, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs MeanSqs F.Model     R2 Pr(>F)
## treatment3  3    1.8842 0.62807  9.0261 0.4362  0.001 ***
## Residuals  35    2.4354 0.06958         0.5638
## Total      38    4.3196                 1.0000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
d2 = distance(cycle2, "bray")
adonis(d2 ~ treatment3, cycle2.2, perm=999)
```

```
##
## Call:
## adonis(formula = d2 ~ treatment3, data = cycle2.2, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
```

```
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs MeanSqs F.Model     R2 Pr(>F)
## treatment3  3    5.6894 1.89647  30.557 0.6807  0.001 ***
## Residuals  43    2.6687 0.06206         0.3193
## Total      46    8.3581                 1.0000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
d3 = distance(cycle3_before_stress, "bray")
adonis(d3 ~ treatment3, cycle3_before_stress.2, perm=999)
```

```
##
## Call:
## adonis(formula = d3 ~ treatment3, data = cycle3_before_stress.2,      permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##            Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## treatment3  3    3.3069 1.10229  22.015 0.56913  0.001 ***
## Residuals  50    2.5035 0.05007         0.43087
## Total      53    5.8104                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Treatment significantly explains differences in microbiome in all cycles (between 43 and 65 % of variation).

Test for cage effects

```
set.seed(42)
cycle1_chloro = subset_samples(cycle1, treatment3 == "Chlorothalonil")
cycle1_chloro.1 <- as(sample_data(cycle1_chloro), "data.frame")
d = distance(cycle1_chloro, "bray")
adonis(d ~ cage, cycle1_chloro.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle1_chloro.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model     R2 Pr(>F)
## cage       2   0.14049 0.070243  1.0818 0.1938  0.369
## Residuals  9   0.58440 0.064934         0.8062
## Total     11   0.72489                 1.0000
```

```
cycle1_gl = subset_samples(cycle1, treatment3 == "Glyphosate")
cycle1_gl.1 <- as(sample_data(cycle1_gl), "data.frame")
d = distance(cycle1_gl, "bray")
adonis(d ~ cage, cycle1_gl.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle1_gl.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.13139 0.065696 0.59371 0.11656   0.75
## Residuals  9   0.99589 0.110655         0.88344
## Total     11   1.12728                 1.00000
```

```
cycle1_co = subset_samples(cycle1, treatment3 == "Control")
cycle1_co.1 <- as(sample_data(cycle1_co), "data.frame")
d = distance(cycle1_co, "bray")
adonis(d ~ cage, cycle1_co.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle1_co.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.09317 0.046587 0.87558 0.16288  0.523
## Residuals  9   0.47887 0.053207         0.83712
## Total     11   0.57204                 1.00000
```

```
#cycle2

cycle2_control = subset_samples(cycle2, treatment3 == "Control")
cycle2_control.1 <- as(sample_data(cycle2_control), "data.frame")
d = distance(cycle2_control, "bray")
adonis(d ~ cage, cycle2_control.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle2_control.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
```

```
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.36945 0.184726  4.0989 0.47668  0.001 ***
## Residuals  9   0.40561 0.045067         0.52332
## Total     11   0.77506                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
cycle2_chloro = subset_samples(cycle2, treatment3 == "Chlorothalonil")
cycle2_chloro.1 <- as(sample_data(cycle2_chloro), "data.frame")
d = distance(cycle2_chloro, "bray")
adonis(d ~ cage, cycle2_chloro.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle2_chloro.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.11535 0.057676  1.0867 0.19451  0.338
## Residuals  9   0.47768 0.053076         0.80549
## Total     11   0.59304                 1.00000
```

```
cycle2_gl = subset_samples(cycle2, treatment3 == "Glyphosate")
cycle2_gl.1 <- as(sample_data(cycle2_gl), "data.frame")
d = distance(cycle2_gl, "bray")
adonis(d ~ cage, cycle2_gl.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle2_gl.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.27728 0.138641  2.1424 0.34879  0.058 .
## Residuals  8   0.51771 0.064713         0.65121
## Total     10   0.79499                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
cycle2_tet = subset_samples(cycle2, treatment3 == "Tetracycline")
cycle2_tet.1 <- as(sample_data(cycle2_tet), "data.frame")
d = distance(cycle2_tet, "bray")
adonis(d ~ cage, cycle2_tet.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle2_tet.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.07478 0.037391 0.78103 0.14789  0.511
## Residuals  9   0.43086 0.047874         0.85211
## Total     11   0.50564                  1.00000
```

```
#cycle 3 before stress

cycle3_control = subset_samples(cycle3_before_stress, treatment3 == "Control")
cycle3_control.1 <- as(sample_data(cycle3_control), "data.frame")
d = distance(cycle3_control, "bray")
adonis(d ~ cage, cycle3_control.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle3_control.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.15617 0.078084  1.5568 0.11484  0.168
## Residuals 24   1.20372 0.050155         0.88516
## Total     26   1.35989                  1.00000
```

```
cycle3_chloro = subset_samples(cycle3_before_stress, treatment3 == "Chlorothalonil")
cycle3_chloro.1 <- as(sample_data(cycle3_chloro), "data.frame")
d = distance(cycle3_chloro, "bray")
adonis(d ~ cage, cycle3_chloro.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle3_chloro.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.06693 0.033465  1.2006 0.28582  0.286
## Residuals  6   0.16723 0.027872         0.71418
## Total      8   0.23417                  1.00000
```

```
cycle3_gl = subset_samples(cycle3_before_stress, treatment3 == "Glyphosate")
cycle3_gl.1 <- as(sample_data(cycle3_gl), "data.frame")
d = distance(cycle3_gl, "bray")
adonis(d ~ cage, cycle3_gl.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle3_gl.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.22242 0.111209  4.4167 0.59551  0.008 **
## Residuals  6   0.15108 0.025179         0.40449
## Total      8   0.37349                  1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
cycle3_tet = subset_samples(cycle3_before_stress, treatment3 == "Tetracycline")
cycle3_tet.1 <- as(sample_data(cycle3_tet), "data.frame")
d = distance(cycle3_tet, "bray")
adonis(d ~ cage, cycle3_tet.1, perm=999)
```

```
##
## Call:
## adonis(formula = d ~ cage, data = cycle3_tet.1, permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cage       2   0.11974 0.059872 0.86309 0.22342  0.569
## Residuals  6   0.41622 0.069370         0.77658
## Total      8   0.53596                  1.00000
```

Testing if the three cages within a treatment show significant variations in microbiome composition (cage effects) reveals that a cage effect is not common in our data. Only in cycle 2 control and cycle 3 glyphosate significant cage variations are seen.

Time effect

Test if microbial composition of treatments differ across the three cycles

```
set.seed(42)
compare = ps1
compare2 <- prune_taxa(taxa_sums(compare) > 0, compare)
```

```r
compare2 <- transform_sample_counts(compare2, function(OTU) {OTU / sum(OTU)})

subs1 <- subset_samples(compare2, treatment_cycle3 == "Control_cycle_1"|treatment_cycle3=="Control_cycl
metadata <- as(sample_data(subs1), "data.frame")
adonis(distance(subs1, method="bray") ~ cycle,
       data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs1, method = "bray") ~ cycle, data = metadata,      permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cycle      2    0.6091 0.304525  5.3998 0.18367  0.002 **
## Residuals 48    2.7070 0.056396         0.81633
## Total     50    3.3160                  1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
subs2 <- subset_samples(compare2, treatment_cycle =="Chlorothalonil_cycle_1"|treatment_cycle== "Chlorot
metadata <- as(sample_data(subs2), "data.frame")
adonis(distance(subs2, method="bray") ~ cycle,
       data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs2, method = "bray") ~ cycle, data = metadata,      permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cycle      2   0.46568 0.232842  4.5006 0.23079  0.002 **
## Residuals 30   1.55209 0.051736         0.76921
## Total     32   2.01778                  1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
subs3 <- subset_samples(compare2, treatment_cycle3 %in% c("Glyphosate_cycle_1", "Glyphosate_cycle_2","G
metadata <- as(sample_data(subs3), "data.frame")
adonis(distance(subs3, method="bray") ~ cycle,
       data = metadata, perm=999)
```

```
##
## Call:
```

```
## adonis(formula = distance(subs3, method = "bray") ~ cycle, data = metadata,      permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## cycle      2   0.75172 0.37586  4.7478 0.24667  0.001 ***
## Residuals 29   2.29577 0.07916         0.75333
## Total     31   3.04748                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
subs4 <- subset_samples(compare2, treatment_cycle %in% c("Tetracycline_cycle_1", "Tetracycline_cycle_2"
metadata <- as(sample_data(subs4), "data.frame")
adonis(distance(subs4, method="bray") ~ cycle,
       data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs4, method = "bray") ~ cycle, data = metadata,      permutations = 999)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## cycle      2   0.51429 0.257144  5.1291 0.32818  0.003 **
## Residuals 21   1.05282 0.050134         0.67182
## Total     23   1.56711                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

All treatments differ significantly across cycles

compare treatments before and after high stress

```
set.seed(42)
compare = ps1
compare2 <- transform_sample_counts(compare2, function(OTU) {OTU / sum(OTU)})
subs1 <- subset_samples(compare2, treatment_cycle %in% c("control_cycle_3_before_stress","control_Chlor
metadata <- as(sample_data(subs1), "data.frame")
adonis(distance(subs1, method="bray") ~ treatment_cycle,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs1, method = "bray") ~ treatment_cycle,      data = metadata, permutati
##
## Permutation: free
## Number of permutations: 999
```

```
##
## Terms added sequentially (first to last)
##
##                 Df SumsOfSqs  MeanSqs F.Model     R2 Pr(>F)
## treatment_cycle  1   0.09785 0.097850  2.1806 0.0638  0.056 .
## Residuals       32   1.43591 0.044872         0.9362
## Total           33   1.53376                  1.0000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
subs1 <- subset_samples(compare2, treatment_cycle %in% c("control_cycle_3_before_stress", "control_Glyp
metadata <- as(sample_data(subs1), "data.frame")
adonis(distance(subs1, method="bray") ~ treatment_cycle,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs1, method = "bray") ~ treatment_cycle,      data = metadata, permutati
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##                 Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment_cycle  1   0.03649 0.036488 0.81354 0.02337  0.533
## Residuals       34   1.52493 0.044851         0.97663
## Total           35   1.56142                  1.00000
```

```
subs1 <- subset_samples(compare2, treatment_cycle %in% c("control_cycle_3_before_stress", "control_Tetr
metadata <- as(sample_data(subs1), "data.frame")
adonis(distance(subs1, method="bray") ~ treatment_cycle,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs1, method = "bray") ~ treatment_cycle,      data = metadata, permutati
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##                 Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment_cycle  1   0.03703 0.037034 0.77715 0.02301  0.569
## Residuals       33   1.57256 0.047653         0.97699
## Total           34   1.60959                  1.00000
```

```
subs1 <- subset_samples(compare2, treatment_cycle %in% c("Chlorothalonil_cycle_3_before_stress", "Chlor
metadata <- as(sample_data(subs1), "data.frame")
adonis(distance(subs1, method="bray") ~ treatment_cycle,data = metadata, perm=999)
```

```
##
## Call:
```

```
## adonis(formula = distance(subs1, method = "bray") ~ treatment_cycle,      data = metadata, permutati
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##                 Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment_cycle  1   0.03599 0.035989  1.2943 0.07484  0.224
## Residuals       16   0.44489 0.027805         0.92516
## Total           17   0.48088                  1.00000
```

```
subs1 <- subset_samples(compare2, treatment_cycle %in% c("Glyphosate_cycle_3_before_stress", "Glyphosat
metadata <- as(sample_data(subs1), "data.frame")
adonis(distance(subs1, method="bray") ~ treatment_cycle,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs1, method = "bray") ~ treatment_cycle,      data = metadata, permutati
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##                 Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment_cycle  1   0.07618 0.076178   1.937 0.10799  0.105
## Residuals       16   0.62925 0.039328         0.89201
## Total           17   0.70543                  1.00000
```

```
subs1 <- subset_samples(compare2, treatment_cycle %in% c("Tetracycline_cycle_3_before_stress", "Tetracy
metadata <- as(sample_data(subs1), "data.frame")
adonis(distance(subs1, method="bray") ~ treatment_cycle,data = metadata, perm=999)
```

```
##
## Call:
## adonis(formula = distance(subs1, method = "bray") ~ treatment_cycle,      data = metadata, permutati
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##                 Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## treatment_cycle  1   0.19271 0.192707  2.8764 0.26446  0.099 .
## Residuals        8   0.53596 0.066996         0.73554
## Total            9   0.72867                  1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

No significant difference for any treatment before and after high stress application -> likely the time was not enough in between the two time points and we sequenced a lot of dead bacteria

test spread of variance of groups

```
set.seed(53)
test<-ps1
test2 <- transform_sample_counts(test, function(OTU) {OTU / sum(OTU)})

cycle1 = subset_samples(test2, cycle == "cycle_one")
cycle1 <- prune_taxa(taxa_sums(cycle1) > 0, cycle1)

cycle2 = subset_samples(test2, cycle == "cycle_two")
cycle2 <- prune_taxa(taxa_sums(cycle2) > 0, cycle2)

cycle3_b = subset_samples(test2, cycle == "cycle_three_before_stress")
cycle3_b <- prune_taxa(taxa_sums(cycle3_b) > 0, cycle3_b)

#testing treatment
set.seed(53)
d_cycle1 = distance(cycle1, "bray")
df_cycle1 = as(sample_data(cycle1), "data.frame")
df_cycle1$treatment3 <- factor(df_cycle1$treatment3 , levels=c("Control", "Chlorothalonil", "Glyphosate
groups <- df_cycle1[["treatment3"]]
beta <- betadisper(d_cycle1, df_cycle1$treatment3)
permutest(beta,pairwise = TRUE, permutations = 999)
```

```
##
## Permutation test for homogeneity of multivariate dispersions
## Permutation: free
## Number of permutations: 999
##
## Response: Distances
##           Df  Sum Sq  Mean Sq     F N.Perm Pr(>F)
## Groups     3 0.12856 0.042852 4.266    999  0.016 *
## Residuals 35 0.35157 0.010045
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Pairwise comparisons:
## (Observed p-value below diagonal, permuted p-value above diagonal)
##                  Control Chlorothalonil Glyphosate Tetracycline
## Control                        0.4190000  0.1080000        0.007
## Chlorothalonil 0.4157655                  0.3450000        0.008
## Glyphosate     0.1177930      0.3295631                   0.016
## Tetracycline   0.0044018      0.0040197  0.0173897
```

```
anova(betadisper(d_cycle1, groups))
```

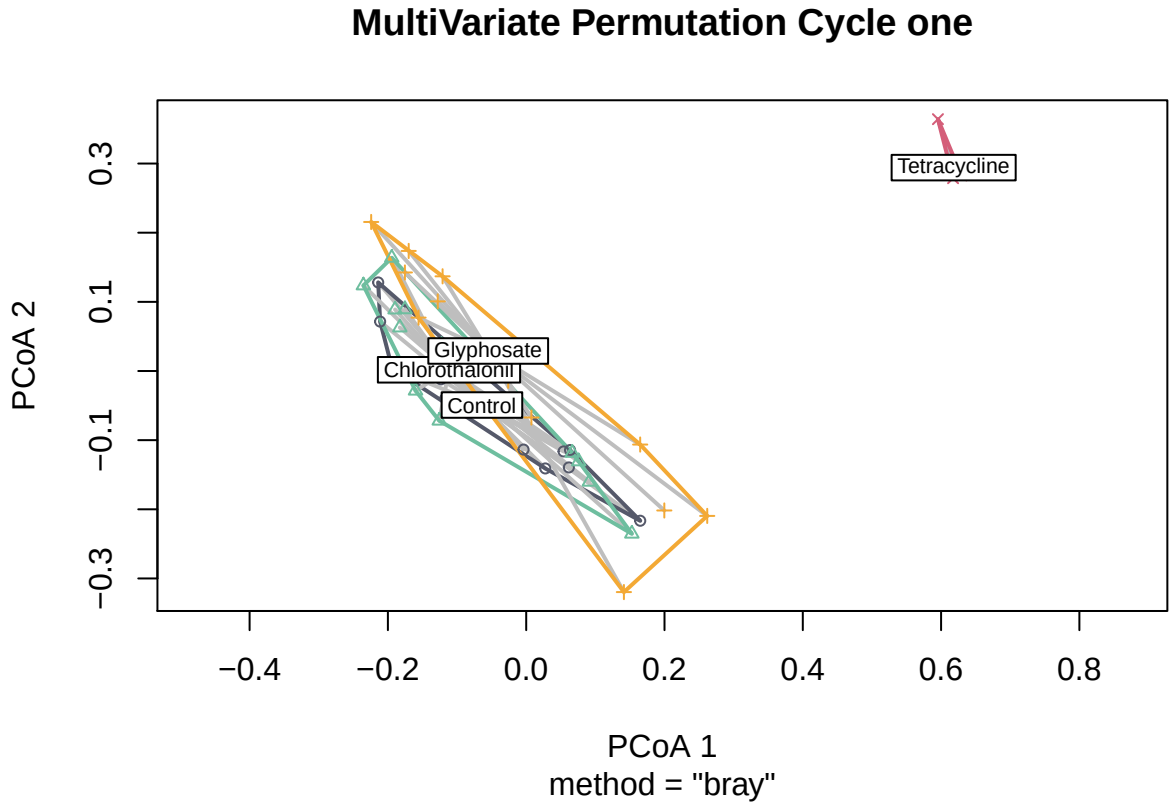
```
## Analysis of Variance Table
##
## Response: Distances
##           Df  Sum Sq  Mean Sq F value  Pr(>F)
## Groups     3 0.12856 0.042852   4.266 0.01142 *
## Residuals 35 0.35157 0.010045
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

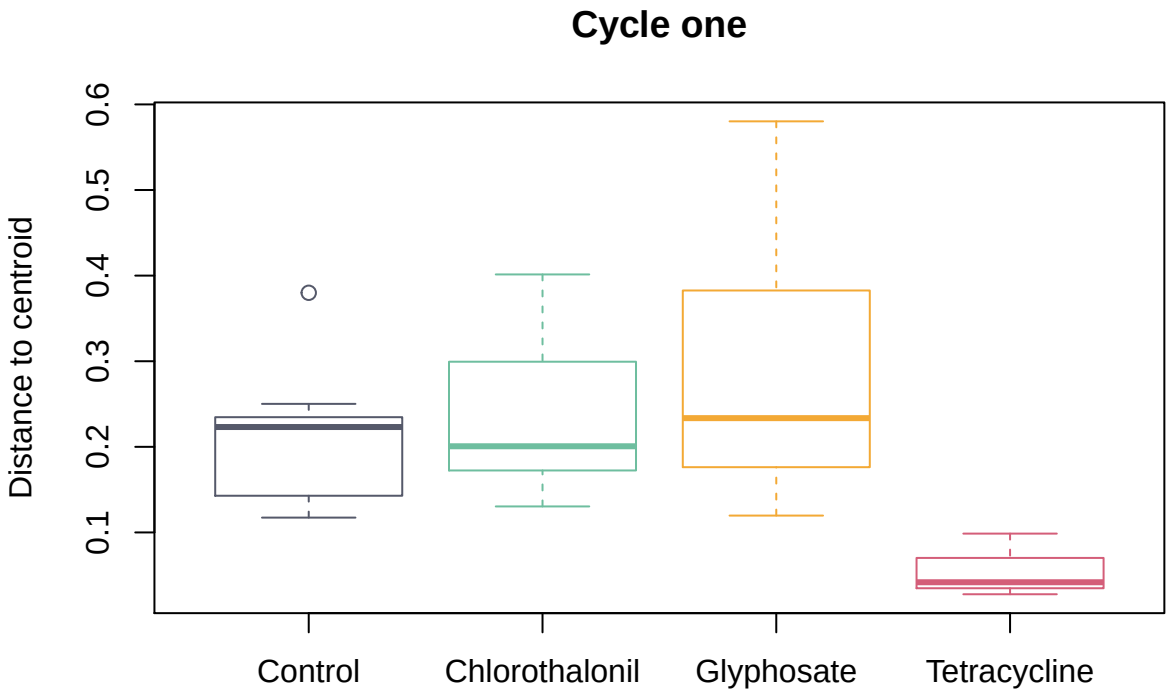
```
mod.HSD <- TukeyHSD(beta,conf.level = 0.95)
mod.HSD
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = distances ~ group, data = df)
##
## $group
##                                     diff         lwr          upr     p adj
## Chlorothalonil-Control       0.02676925 -0.08357872  0.137117211 0.9133487
## Glyphosate-Control           0.07354989 -0.03679808  0.183897850 0.2915645
## Tetracycline-Control        -0.15206757 -0.32654302  0.022407884 0.1060221
## Glyphosate-Chlorothalonil    0.04678064 -0.06356733  0.157128605 0.6656851
## Tetracycline-Chlorothalonil -0.17883681 -0.35331227 -0.004361362 0.0428096
## Tetracycline-Glyphosate     -0.22561745 -0.40009291 -0.051142001 0.0069775
```

```
plot(betadisper(d_cycle1, groups),main="MultiVariate Permutation Cycle one",lwd=2, label = TRUE,label.c
```




```
boxplot(betadisper(d_cycle1, groups),main="Cycle one",border=c("#55596a", "#6ebe9f","#f3a935", "#D45E79
```




```
set.seed(53)
d_cycle2 = distance(cycle2, "bray")
df_cycle2 = as(sample_data(cycle2), "data.frame")
df_cycle2$treatment3 <- factor(df_cycle2$treatment3 , levels=c("Control", "Chlorothalonil", "Glyphosate
groups <- df_cycle2[["treatment3"]]
beta <- betadisper(d_cycle2, df_cycle2$treatment3)
permutest(beta,pairwise = TRUE, permutations = 99)
```

```
##
## Permutation test for homogeneity of multivariate dispersions
## Permutation: free
## Number of permutations: 99
##
## Response: Distances
##           Df  Sum Sq   Mean Sq      F N.Perm Pr(>F)
## Groups     3 0.02853 0.0095114 0.6322     99   0.64
## Residuals 43 0.64693 0.0150448
##
## Pairwise comparisons:
## (Observed p-value below diagonal, permuted p-value above diagonal)
##                Control Chlorothalonil Glyphosate Tetracycline
## Control                        0.96000    0.48000         0.66
## Chlorothalonil 0.94337                    0.29000         0.68
## Glyphosate     0.46170        0.27741                    0.11
## Tetracycline   0.66557        0.62562    0.11957
```

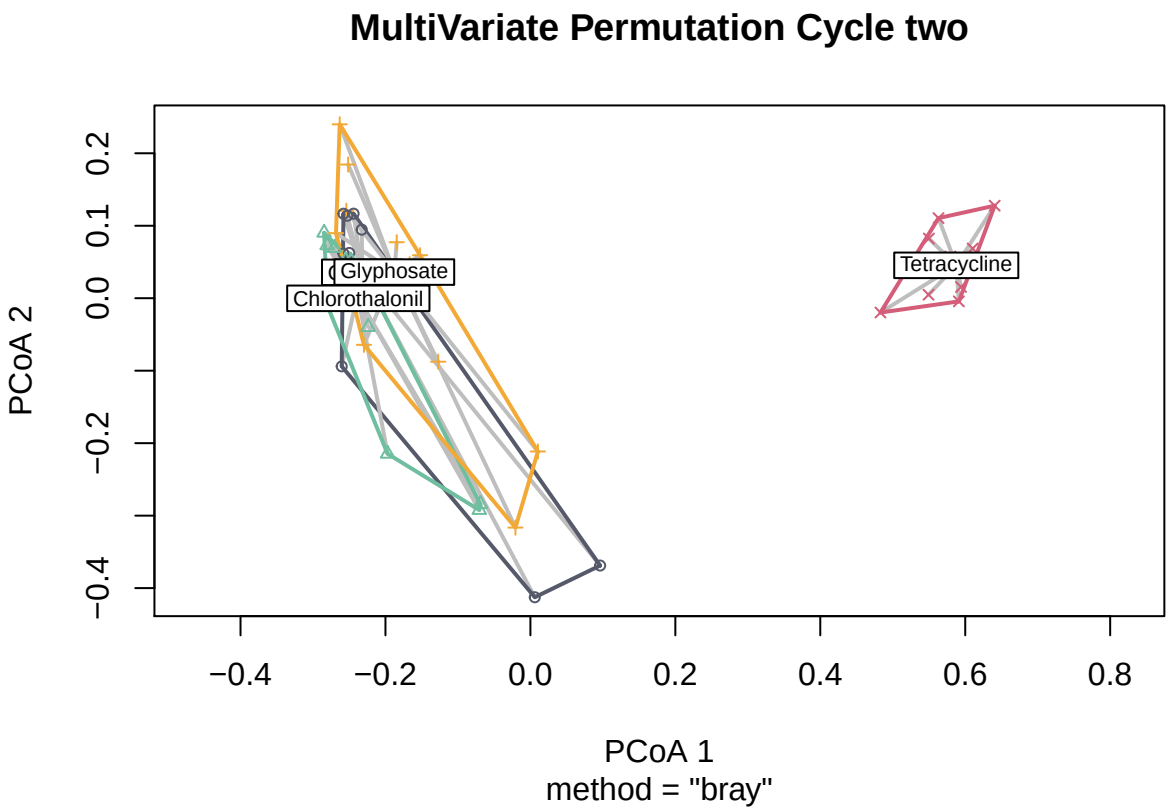
```r
anova(betadisper(d_cycle2, groups))
```

```
## Analysis of Variance Table
##
## Response: Distances
##           Df  Sum Sq   Mean Sq F value Pr(>F)
## Groups     3 0.02853 0.0095114  0.6322 0.5983
## Residuals 43 0.64693 0.0150448
```

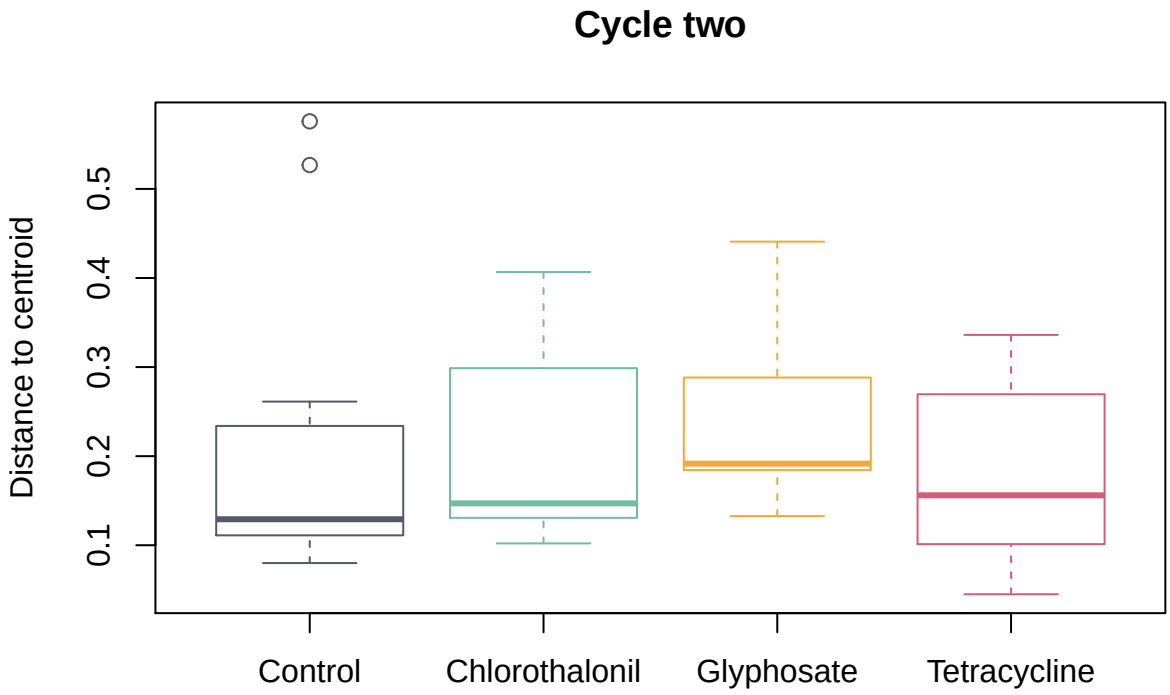
```r
mod.HSD <- TukeyHSD(beta)
mod.HSD
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = distances ~ group, data = df)
##
## $group
##                                    diff         lwr        upr     p adj
## Chlorothalonil-Control      -0.004100517 -0.13792105 0.12972001 0.9997999
## Glyphosate-Control           0.044085522 -0.09274259 0.18091363 0.8247329
## Tetracycline-Control        -0.024768446 -0.15858897 0.10905208 0.9598636
## Glyphosate-Chlorothalonil    0.048186039 -0.08864207 0.18501415 0.7829567
## Tetracycline-Chlorothalonil -0.020667929 -0.15448846 0.11315260 0.9759900
## Tetracycline-Glyphosate     -0.068853968 -0.20568207 0.06797414 0.5401612
```

```r
plot(betadisper(d_cycle2, groups),main="MultiVariate Permutation Cycle two",lwd=2, label = TRUE,label.c
```




```
boxplot(betadisper(d_cycle2, groups),main="Cycle two",border=c("#55596a", "#6ebe9f","#f3a935", "#D45E79
```




```
set.seed(53)
d_cycle3_b = distance(cycle3_b, "bray")
df_cycle3_b = as(sample_data(cycle3_b), "data.frame")
df_cycle3_b$treatment3 <- factor(df_cycle3_b$treatment3 , levels=c("Control", "Chlorothalonil", "Glypho
groups <- df_cycle3_b[["treatment3"]]
beta <- betadisper(d_cycle3_b, df_cycle3_b$treatment3)
permutest(beta,pairwise = TRUE, permutations = 99)
```

```
##
## Permutation test for homogeneity of multivariate dispersions
## Permutation: free
## Number of permutations: 99
##
## Response: Distances
##           Df  Sum Sq  Mean Sq      F N.Perm Pr(>F)
## Groups     3 0.02830 0.009432 0.8083     99   0.51
## Residuals 50 0.58346 0.011669
##
## Pairwise comparisons:
## (Observed p-value below diagonal, permuted p-value above diagonal)
##                 Control Chlorothalonil Glyphosate Tetracycline
## Control                        0.510000   0.780000         0.40
## Chlorothalonil 0.416346                   0.610000         0.02
## Glyphosate     0.732130       0.641863                    0.27
## Tetracycline   0.375692       0.019867   0.246354
```

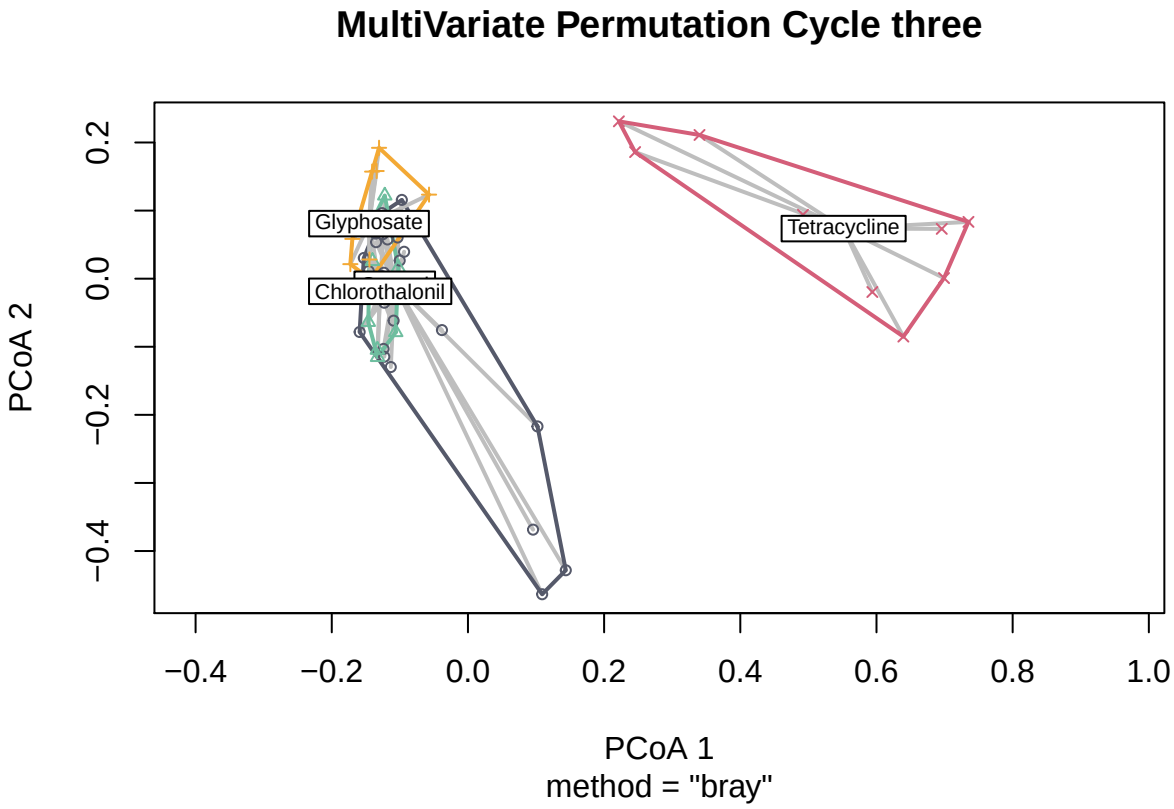
```r
anova(betadisper(d_cycle3_b, groups))
```

```
## Analysis of Variance Table
##
## Response: Distances
##           Df  Sum Sq  Mean Sq F value Pr(>F)
## Groups     3 0.02830 0.009432  0.8083 0.4953
## Residuals 50 0.58346 0.011669
```

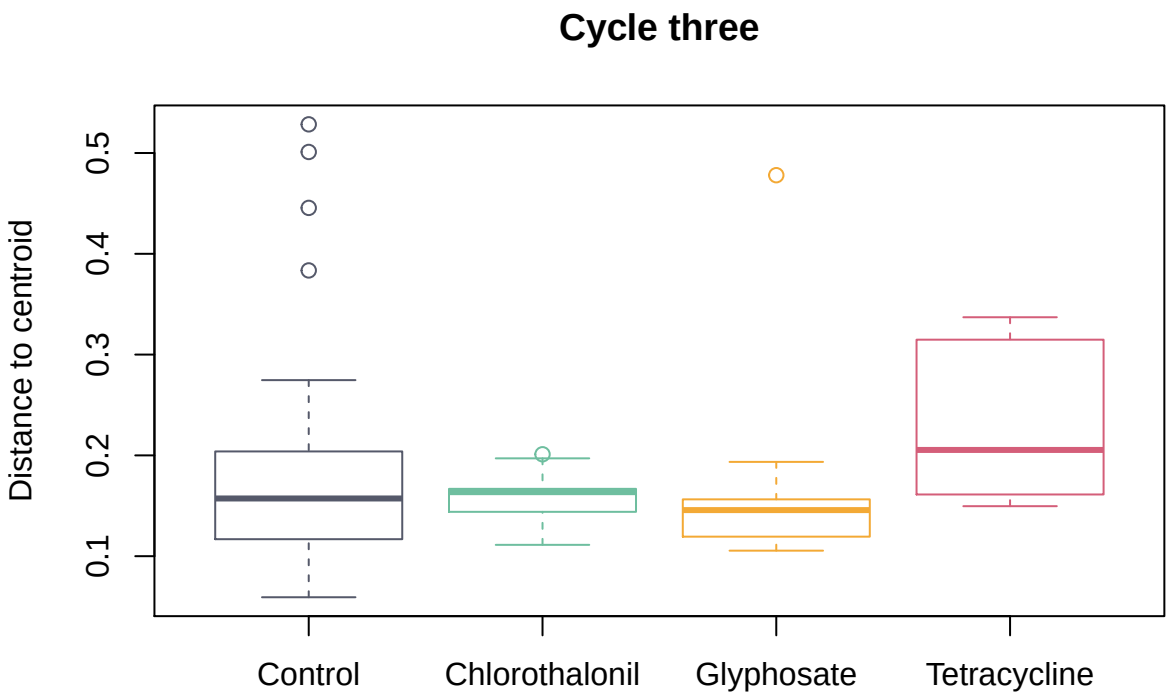
```r
mod.HSD <- TukeyHSD(beta)
mod.HSD
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = distances ~ group, data = df)
##
## $group
##                                    diff         lwr        upr     p adj
## Chlorothalonil-Control      -0.03526071 -0.14575879 0.07523736 0.8311939
## Glyphosate-Control          -0.01645136 -0.12694944 0.09404672 0.9787710
## Tetracycline-Control         0.04062639 -0.06987169 0.15112446 0.7630330
## Glyphosate-Chlorothalonil    0.01880935 -0.11652260 0.15414131 0.9825928
## Tetracycline-Chlorothalonil  0.07588710 -0.05944485 0.21121905 0.4509069
## Tetracycline-Glyphosate      0.05707775 -0.07825420 0.19240970 0.6785135
```

```r
plot(betadisper(d_cycle3_b, groups),main="MultiVariate Permutation Cycle three",lwd=2, label = TRUE,lab
```




```
boxplot(betadisper(d_cycle3_b, groups),main="Cycle three",border=c("#55596a", "#6ebe9f","#f3a935", "#D45
```



Taxonomy plots and stats

```
tax <-ps1
ps_bdiv2 <- transform_sample_counts(tax, function(OTU) {OTU / sum(OTU)})
tax_all2<- ps_bdiv2
ps_pd2 <- tax_glom(tax_all2, taxrank = 'Genus')

low1pc_reads <- max(taxa_sums(ps_pd2)) * 1 / 100
low1pc_indices <- which(taxa_sums(ps_pd2) < low1pc_reads)
length(low1pc_indices)
```

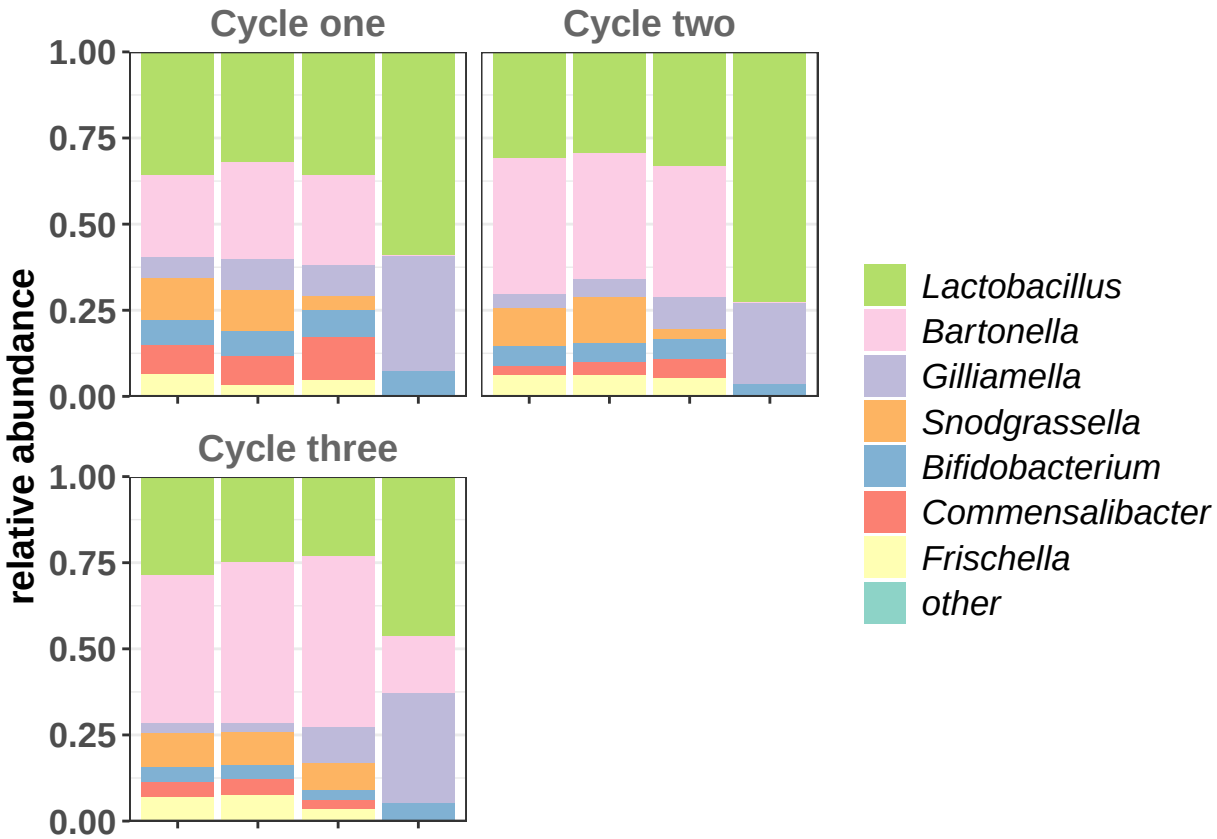
```
## [1] 26
```

```
taxa_names(ps_pd2)[low1pc_indices[1]] <- "other"
taxa3 <- merge_taxa(ps_pd2, low1pc_indices, "other")
tax_table(taxa3)["other", ] <- c("other")

#build a custom color palette for phyloseq object
getPalette = colorRampPalette(brewer.pal(8, "Set3"))
speciesList = unique(tax_table(taxa3)[,"Genus"])
speciesPalette = getPalette(length(speciesList))
names(speciesPalette) = speciesList
```

```
ps2 <-taxa3
ps3 = subset_samples(ps2, cycle=="cycle_one" | cycle=="cycle_two" | cycle == "cycle_three_before_stress
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)

ps_df <- psmelt(ps3)
ps_df_sum <- ps_df %>%
  group_by(treatment3,Genus,cycle) %>%
  summarise(Abundance = mean(Abundance)) %>%
  group_by(treatment3,cycle) %>%
  mutate(relative_abundance = Abundance / sum(Abundance)) %>%
  ungroup() %>%
  mutate(Genus = fct_reorder(Genus, Abundance, .fun = sum, .desc = TRUE))
levels(ps_df_sum$Genus)
```

```
## [1] "Lactobacillus"    "Bartonella"       "Gilliamella"      "Snodgrassella"
## [5] "Bifidobacterium"  "Commensalibacter" "Frischella"       "other"
```

```
neworder <- c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
ps_df_sum2 <- arrange(transform(ps_df_sum, cycle=factor(cycle,levels=neworder,labels=change)),cycle)

neworder2 <- c("Control","Chlorothalonil","Glyphosate", "Tetracycline")
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2)),treatment3)

p<-ggplot(data = ps_df_sum3, aes(x = treatment3, y = relative_abundance, fill = Genus)) +
  geom_bar(stat = "identity")+ scale_fill_manual(values= speciesPalette)+ scale_y_continuous(expand = c
#p<-p+ theme(legend.position="bottom")
p<-p+ theme(legend.text=element_text(size=13, face = "italic"))+theme(legend.key = element_rect(color =
                                                    legend.key.size = unit(0.6, "cm"))+theme(legend.ti
#p<-p + facet_grid( ~cycle, scales="free_x",space="free")
p<-p + facet_wrap( ~cycle, scales="free_x",nrow=2)
p<-p + theme(strip.text = element_text(size=14,face="bold",color="grey40"))
taxa2<-p+theme(axis.title.y = element_text(size=14, face="bold"))+theme(axis.text.y = element_text(size
taxa2
```



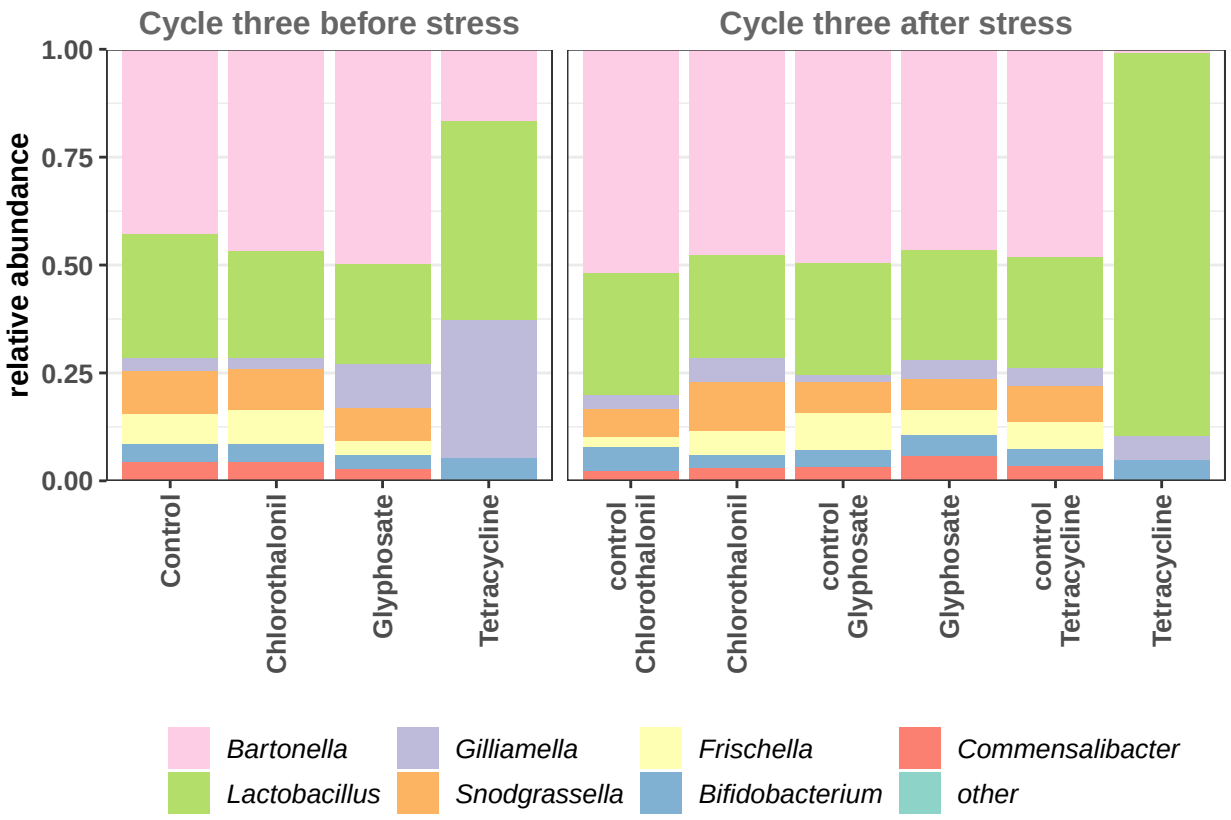

```
ggsave("R_microbiome_figures/taxa_man.png", height = 4.5, width = 6.5)
```

```
ps2 <-taxa3
ps3 = subset_samples(ps2, cycle !="positive_control" & cycle !="hive_bee" & cycle !="cycle_two"& cycle
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)

ps_df <- psmelt(ps3)
ps_df_sum <- ps_df %>%
  group_by(treatment3,Genus,cycle) %>%
  summarise(Abundance = mean(Abundance)) %>%
  group_by(treatment3,cycle) %>%
  mutate(relative_abundance = Abundance / sum(Abundance)) %>%
  ungroup() %>%
  mutate(Genus = fct_reorder(Genus, Abundance, .fun = sum, .desc = TRUE))
levels(ps_df_sum$Genus)
```

```
## [1] "Bartonella"       "Lactobacillus"    "Gilliamella"      "Snodgrassella"
## [5] "Frischella"       "Bifidobacterium"  "Commensalibacter" "other"
```

```
neworder <- c("cycle_three_before_stress","cycle_three_after_stress")
change <- c("Cycle three before stress","Cycle three after stress")

ps_df_sum2 <- arrange(transform(ps_df_sum, cycle=factor(cycle,levels=neworder,labels=change)),cycle)

neworder2 <- c("Control","control_Chlorothalonil", "Chlorothalonil","control_Glyphosate","Glyphosate","
```

```
change2 <- c("Control","control \n Chlorothalonil", "Chlorothalonil","control \n Glyphosate","Glyphosat
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2,labels=change

p<-ggplot(data = ps_df_sum3, aes(x = treatment3, y = relative_abundance, fill = Genus)) +
  geom_bar(stat = "identity")+ scale_fill_manual(values= speciesPalette)+ scale_y_continuous(expand = c
p<-p+ theme(legend.position="bottom")
p<-p+ theme(legend.text=element_text(size=10, face = "italic"))+theme(legend.key = element_rect(color =
                                                    legend.key.size = unit(0.6, "cm"))+theme(legend.ti
taxa3<-p +theme(strip.text = element_text(size=12,face="bold",color="grey40"))
taxa3<-taxa3+theme(axis.title.y = element_text(size=11, face="bold"))+theme(axis.text.y = element_text(
taxa3
```




```
ggsave("R_microbiome_figures/Supp_taxonomy.png", height = 4.5, width = 5.9)
```

Core taxa abundance

```
ps2 <-ps1
ps2 <- rarefy_even_depth(ps2,sample.size=12351, replace=FALSE, rngseed = 1)
ps2 = subset_samples(ps2, cycle == "cycle_one"| cycle == "cycle_two"| cycle == "cycle_three_before_stres
ps3 <- prune_taxa(taxa_sums(ps2) > 0, ps2)
ps3 <- tax_glom(ps3, taxrank = 'Genus')
psOrd3 = subset_taxa(ps3, Genus=="Lactobacillus" | Genus=="Bartonella" | Genus=="Gilliamella" | Genus==
```

```r
#Melt and plot
melt<-psmelt(psOrd3)
levels2=c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
melt2 <- arrange(transform(melt, cycle=factor(cycle,levels=levels2,labels=change)),cycle)
neworder2 <- c("Bartonella","Lactobacillus","Gilliamella","Snodgrassella","Bifidobacterium","Commensali
melt3 <- arrange(transform(melt2, Genus=factor(Genus,levels=neworder2)),Genus)
neworder3 <- c("Control","Chlorothalonil","Glyphosate","Tetracycline")
melt4 <- arrange(transform(melt3, treatment3=factor(treatment3,levels=neworder3)),treatment3)

a_mean <- melt4 %>%
  group_by(treatment3,Genus,cycle) %>%
  summarize(mean_val = mean(Abundance))
a_mean2<-subset(a_mean, treatment3 =="Control")
print(a_mean2)
```

```
## # A tibble: 21 x 4
## # Groups:   treatment3, Genus [7]
##    treatment3 Genus         cycle       mean_val
##    <fct>      <fct>         <fct>          <dbl>
##  1 Control    Bartonella    Cycle one      2959.
##  2 Control    Bartonella    Cycle two      4882.
##  3 Control    Bartonella    Cycle three    5283.
##  4 Control    Lactobacillus Cycle one      4373.
##  5 Control    Lactobacillus Cycle two      3780.
##  6 Control    Lactobacillus Cycle three    3538.
##  7 Control    Gilliamella   Cycle one       743
##  8 Control    Gilliamella   Cycle two       512.
##  9 Control    Gilliamella   Cycle three     385.
## 10 Control    Snodgrassella Cycle one      1525.
## # ... with 11 more rows
```

```r
sigFunc = function(x){
  if(x < 0.001){"***"}
  else if(x < 0.01){"**"}
  else if(x < 0.05){"*"}
  else{NA}}

p<-ggplot(data = melt4, aes(x = treatment3, y = Abundance)) +
  geom_boxplot(aes(fill=Genus),alpha=0.5,lwd=0.7, position = position_dodge(width = 0.3), width=0.45,ou
  labs(x = "", y = "Abundance\n")+
  facet_grid(Genus~cycle, scales = "free")+theme_bw()+
  geom_signif(comparisons=list(c("Control", "Chlorothalonil"), c("Control", "Glyphosate"),c("Control",
p<-p+ theme(legend.position="right")+ylab("Total abundance")
p<-p+ theme(legend.text=element_text(size=29, face = "italic"))+theme(legend.key = element_rect(color =
abu<-p +theme(strip.text.x = element_text(size=25,face="bold",color = "grey 35"))+theme(strip.backgroun
cycle<-abu+theme(axis.title.y = element_text(size=25, face="bold"))+theme(axis.text.y = element_text(si
ggsave("R_microbiome_figures/core_total_abundance.png", height = 18, width = 16)
```

stats on abundances

compare if taxa abundances are significantly different between controls and treatments in cycle 1, 2 and cycle 3, use Wilcoxon tests with following fdr correction. In addition test if control samples differ in the abundances of the core taxa between cycle 1 and 3 (time or lab-adaptation effect)

```
set.seed(42)
ps2 <-ps1
ps2 <- rarefy_even_depth(ps2,sample.size=12351, replace=FALSE, rngseed = 1)
ps2 = subset_samples(ps2, cycle == "cycle_one"| cycle == "cycle_two"| cycle == "cycle_three_before_stres
ps3 <- prune_taxa(taxa_sums(ps2) > 0, ps2)
abundance <- tax_glom(ps3, taxrank = 'Genus')
Lac = subset_taxa(abundance, Genus=="Lactobacillus")

#cycle 1
Gly <- subset_samples(Lac, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Glyphosate_cycle_1"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 73, p-value = 0.9774
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Lac, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Chlorothalonil_cycl
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 58, p-value = 0.4428
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Lac, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 0, p-value = 0.004396
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.9774,0.4428,0.0044)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.9774 0.6642 0.0132
```

```
#cycle 2
Gly <- subset_samples(Lac, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Glyphosate_cycle_2"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 53, p-value = 0.4491
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Lac, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Chlorothalonil_cyc
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 80, p-value = 0.6707
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Lac, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 6, p-value = 2.219e-05
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.45,0.67,0.000022)
p.adjust(pvalues,method="fdr")
```

```
## [1] 6.7e-01 6.7e-01 6.6e-05
```

```
#cycle 3
Gly <- subset_samples(Lac, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Glyphosat
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 163, p-value = 0.1361
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Lac, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Chloroth
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 112, p-value = 0.7467
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Lac, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Tetracycl
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 33, p-value = 0.0006537
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.136,0.75,0.0007)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.2040 0.7500 0.0021
```

```
# test control cycle 1 in comparison to cycle 3

Lac1 <- subset_samples(Lac,treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Control_c
Lac1 <- prune_taxa(taxa_sums(Lac1) > 0, Lac1)
melt<-psmelt(Lac1)
wilcox.test(Abundance~treatment_cycle, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment_cycle
## W = 234, p-value = 0.02813
## alternative hypothesis: true location shift is not equal to 0
```

```
Bart = subset_taxa(abundance, Genus=="Bartonella")

Gly <- subset_samples(Bart, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Glyphosate_cycle_1
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 66, p-value = 0.7553
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Bart, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Chlorothalonil_cy
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 83, p-value = 0.5512
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Bart, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Tetracycline_cycle
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 36, p-value = 0.004396
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.76,0.55,0.0044)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.7600 0.7600 0.0132
```

```
Gly <- subset_samples(Bart, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Glyphosate_cycle_2
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
```

```
##
## data:  Abundance by treatment2
## W = 74, p-value = 0.6505
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Bart, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Chlorothalonil_cy
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 59, p-value = 0.4776
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Bart, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 142.5, p-value = 5.253e-05
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.65, 0.48, 5.253e-05)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.65000000 0.65000000 0.00015759
```

```
#cycle 3
Gly <- subset_samples(Bart, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Glyphosa
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 92, p-value = 0.2951
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Bart, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Chlorot
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 124, p-value = 0.9428
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Bart, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Tetracycl
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 213, p-value = 0.0003847
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.3,0.94,0.0004)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.4500 0.9400 0.0012
```

```
# test control cycle 1 in comparison to cycle 3
Bart1 <- subset_samples(Bart,treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Control_
Bart1 <- prune_taxa(taxa_sums(Bart1) > 0, Bart1)
melt<-psmelt(Bart1)
wilcox.test(Abundance~treatment_cycle, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment_cycle
## W = 55, p-value = 0.0007082
## alternative hypothesis: true location shift is not equal to 0
```

Gilliamella

```
Gil = subset_taxa(abundance, Genus=="Gilliamella")

Gly <- subset_samples(Gil, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Glyphosate_cycle_1"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 40, p-value = 0.06836
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Gil, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Chlorothalonil_cycl
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 105, p-value = 0.05966
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Gil, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Tetracycline_cycle_1
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 0, p-value = 0.004396
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.07,0.06,0.0044)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.0700 0.0700 0.0132
```

```
Gly <- subset_samples(Gil, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Glyphosate_cycle_2")
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 31, p-value = 0.03174
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Gil, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Chlorothalonil_cycl
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 83, p-value = 0.5512
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Gil, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Tetracycline_cycle_2
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 22, p-value = 0.002914
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.03,0.55,0.003)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.045 0.550 0.009
```

```
#cycle 3
Gly <- subset_samples(Gil, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Glyphosate
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment3
## W = 96.5, p-value = 0.3707
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Gil, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Chlorotha
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 108, p-value = 0.6406
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Gil, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Tetracycli
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 0, p-value = 2.124e-08
## alternative hypothesis: true location shift is not equal to 0
```

```r
pvalues<-c(0.3707,0.6406,2.124e-08)
p.adjust(pvalues,method="fdr")
```

```
## [1] 5.5605e-01 6.4060e-01 6.3720e-08
```

```r
#cycle 3 in comp cycle 1 control
Gil1 <- subset_samples(Gil,treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Control_cy
Gil1 <- prune_taxa(taxa_sums(Gil1) > 0, Gil1)
melt<-psmelt(Gil1)
wilcox.test(Abundance~treatment_cycle, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment_cycle
## W = 262, p-value = 0.001714
## alternative hypothesis: true location shift is not equal to 0
```

Snodgrassella

```r
set.seed(42)
ps2 <-ps1
ps2 <- rarefy_even_depth(ps2,sample.size=12351, replace=FALSE, rngseed = 1)
ps2 = subset_samples(ps2, cycle == "cycle_one"| cycle == "cycle_two"| cycle == "cycle_three_before_stres
ps3 <- prune_taxa(taxa_sums(ps2) > 0, ps2)
abundance <- tax_glom(ps3, taxrank = 'Genus')
Snod = subset_taxa(abundance, Genus=="Snodgrassella")

Gly <- subset_samples(Snod, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Glyphosate_cycle_1
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 132, p-value = 0.0002012
## alternative hypothesis: true location shift is not equal to 0
```

```r
Chlo <- subset_samples(Snod, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Chlorothalonil_cy
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 67, p-value = 0.7987
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Snod, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 36, p-value = 0.01125
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.0002,0.8,0.0113)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.00060 0.80000 0.01695
```

```
Gly <- subset_samples(Snod, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Glyphosate_cycle_2
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 132, p-value = 1.479e-06
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Snod, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Chlorothalonil_cy
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 97, p-value = 0.16
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Snod, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 144, p-value = 3.588e-05
## alternative hypothesis: true location shift is not equal to 0
```

```r
pvalues<-c(1.479e-06,0.16,3.588e-05)
p.adjust(pvalues,method="fdr")
```

```
## [1] 4.437e-06 1.600e-01 5.382e-05
```

```r
#cycle 3
Gly <- subset_samples(Snod, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Glyphosa
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment3
## W = 144, p-value = 0.4215
## alternative hypothesis: true location shift is not equal to 0
```

```r
Chlo <- subset_samples(Snod, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Chlorot
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment3
## W = 113.5, p-value = 0.7841
## alternative hypothesis: true location shift is not equal to 0
```

```r
Tet <- subset_samples(Snod, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Tetracycl
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment3
## W = 243, p-value = 9.578e-06
## alternative hypothesis: true location shift is not equal to 0
```

```r
pvalues<-c(0.42,0.78,9.578e-06)
p.adjust(pvalues,method="fdr")
```

```
## [1] 6.3000e-01 7.8000e-01 2.8734e-05
```

```r
#cycle 3 in comp cycle 1 control
Snod1 <- subset_samples(Snod,treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Control
Snod1 <- prune_taxa(taxa_sums(Snod1) > 0, Snod1)
melt<-psmelt(Snod1)
wilcox.test(Abundance~treatment_cycle, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment_cycle
## W = 238, p-value = 0.02159
## alternative hypothesis: true location shift is not equal to 0
```

Frischella

```
Fri = subset_taxa(abundance, Genus=="Frischella")

Gly <- subset_samples(Fri, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Glyphosate_cycle_1"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 104, p-value = 0.06872
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Fri, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Chlorothalonil_cyc
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 35, p-value = 0.03324
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Fri, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 36, p-value = 0.004396
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.06872,0.033,0.004396)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.068720 0.049500 0.013188
```

```
Gly <- subset_samples(Fri, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Glyphosate_cycle_2"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 71, p-value = 0.7818
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Fri, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Chlorothalonil_cyc
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 79, p-value = 0.7125
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Fri, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 144, p-value = 1.831e-05
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.7818,0.713,0.00001831)
p.adjust(pvalues,method="fdr")
```

```
## [1] 7.818e-01 7.818e-01 5.493e-05
```

```
#cycle 3
Gly <- subset_samples(Fri, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Glyphosat
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 184, p-value = 0.02154
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Fri, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Chloroth
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 135, p-value = 0.6406
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Fri, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Tetracycl
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment3
## W = 243, p-value = 8.806e-06
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.02,0.64,8.806e-06)
p.adjust(pvalues,method="fdr")
```

```
## [1] 3.0000e-02 6.4000e-01 2.6418e-05
```

```
#cycle 3 in comp cycle 1 control
Fri1 <- subset_samples(Fri,treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Control_cy
Fri1 <- prune_taxa(taxa_sums(Fri1) > 0, Fri1)
melt<-psmelt(Fri1)
wilcox.test(Abundance~treatment_cycle, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment_cycle
## W = 164, p-value = 0.9641
## alternative hypothesis: true location shift is not equal to 0
```

Commensalibacter

```
Com = subset_taxa(abundance, Genus=="Commensalibacter")

Gly <- subset_samples(Com, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Glyphosate_cycle_1"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 40, p-value = 0.0689
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Com, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Chlorothalonil_cycl
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 77, p-value = 0.795
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Com, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 36, p-value = 0.01117
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.069,0.795,0.0111)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.1035 0.7950 0.0333
```

```
Gly <- subset_samples(Com, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Glyphosate_cycle_2"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 48, p-value = 0.2875
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Com, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Chlorothalonil_cycl
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 97.5, p-value = 0.1488
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Com, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Tetracycline_cycle_1
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 139, p-value = 0.0001015
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.288,0.149,0.0001015)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.2880000 0.2235000 0.0003045
```

```
#cycle 3
Gly <- subset_samples(Com, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Glyphosat
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 145, p-value = 0.4075
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Com, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Chlorotha
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 121, p-value = 1
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Com, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Tetracycli
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment3
## W = 241.5, p-value = 1.21e-05
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.41,1,0.000012)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.615000 1.000000 0.000036
```

```
#cycle 3 in comp cycle 1 control
Com1 <- subset_samples(Com,treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Control_cy
Com1 <- prune_taxa(taxa_sums(Com1) > 0, Com1)
melt<-psmelt(Com1)
wilcox.test(Abundance~treatment_cycle, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment_cycle
## W = 280, p-value = 0.0003494
## alternative hypothesis: true location shift is not equal to 0
```

Bifidobacterium

```
Bif = subset_taxa(abundance, Genus=="Bifidobacterium")

Gly <- subset_samples(Bif, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Glyphosate_cycle_1"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 66, p-value = 0.7553
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Bif, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Chlorothalonil_cycl
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 69, p-value = 0.8874
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Bif, treatment_cycle =="Control_cycle_1" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 17, p-value = 0.9451
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.755,0.887,0.9451)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.9451 0.9451 0.9451
```

```
Gly <- subset_samples(Bif, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Glyphosate_cycle_2"
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment2
## W = 77, p-value = 0.5254
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Bif, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Chlorothalonil_cyc
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 62.5, p-value = 0.6033
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Bif, treatment_cycle =="Control_cycle_2" | treatment_cycle =="Tetracycline_cycle_
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment2, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment2
## W = 104.5, p-value = 0.06461
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.5254,0.603,0.065)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.603 0.603 0.195
```

```
#cycle 3
Gly <- subset_samples(Bif, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Glyphosat
Gly <- prune_taxa(taxa_sums(Gly) > 0, Gly)
melt<-psmelt(Gly)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 168, p-value = 0.09324
## alternative hypothesis: true location shift is not equal to 0
```

```
Chlo <- subset_samples(Bif, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Chloroth
Chlo <- prune_taxa(taxa_sums(Chlo) > 0, Chlo)
melt<-psmelt(Chlo)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment3
## W = 129, p-value = 0.8017
## alternative hypothesis: true location shift is not equal to 0
```

```
Tet <- subset_samples(Bif, treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Tetracycl
Tet <- prune_taxa(taxa_sums(Tet) > 0, Tet)
melt<-psmelt(Tet)
wilcox.test(Abundance~treatment3, data=melt)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Abundance by treatment3
## W = 92.5, p-value = 0.2978
## alternative hypothesis: true location shift is not equal to 0
```

```
pvalues<-c(0.093,0.802,0.298)
p.adjust(pvalues,method="fdr")
```

```
## [1] 0.279 0.802 0.447
```

```
#cycle 3 in comp cycle 1 control
Bif1 <- subset_samples(Bif,treatment_cycle=="control_cycle_3_before_stress"|treatment_cycle=="Control_cy
Bif1 <- prune_taxa(taxa_sums(Bif1) > 0, Bif1)
melt<-psmelt(Bif1)
wilcox.test(Abundance~treatment_cycle, data=melt)
```

```
##
##  Wilcoxon rank sum test
##
## data:  Abundance by treatment_cycle
## W = 264, p-value = 0.001343
## alternative hypothesis: true location shift is not equal to 0
```

core taxa abundances: Gardner-Altman estimation plots

```
#devtools::install_github("mikheyev/dabestr")
library(dabestr)
microbeColors <- function (n) {
  mcols <- c("#55596a", "#6ebe9f","#f3a935", "#D45E79")
  return(mcols[4 %/% n])
}
```

Bartonella

```
ps2 <-ps1
set.seed(42)
ps2.rare = rarefy_even_depth(ps2, rngseed=1, sample.size=12351, replace=F)

sample_data(ps2.rare)$treatment3<-factor(sample_data(ps2.rare)$treatment3,levels=c("Control","Chlorotha
levels(sample_data(ps2.rare)$treatment3)
```

```
## [1] "Control" "Chloro"  "Glypho"  "Tetra"
```

```
cycle1 = subset_samples(ps2.rare, cycle == "cycle_one")
ps3 <- tax_glom(cycle1, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Bartonella")

#Melt and plot
melt<-psmelt(psord)

unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
```

```r
# Display the results in a user-friendly format.
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  537 [95CI  -978; 1980]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=12)
##  281 [95CI  -1320; 1740]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=3)
##  -2940 [95CI  -3750; -1980]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```r
Bartonella_cycle1<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",tick.fontsize=17,axes.title.fo

#cycle2
cycle2 = subset_samples(ps2.rare, cycle == "cycle_two")
ps3 <- tax_glom(cycle2, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Bartonella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  -343 [95CI  -1860; 1320]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=11)
##  -159 [95CI  -1820; 1470]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=12)
##  -4850 [95CI  -5900; -3390]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Bartonella_cycle2<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff.",t


#cycle3
cycle3 = subset_samples(ps2.rare, cycle == "cycle_three_before_stress")
ps3 <- tax_glom(cycle3, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Bartonella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=27) minus Control (n=9)
##  462 [95CI  -372; 1530]
##
## Unpaired mean difference of Glypho (n=27) minus Control (n=9)
##  853 [95CI  -156; 1880]
##
## Unpaired mean difference of Tetra (n=27) minus Control (n=9)
##  -3250 [95CI  -4640; -1470]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Bartonella_cycle3<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff.",t
```

Lactobacillus

```
ps2 <-ps1
set.seed(42)
ps2.rare = rarefy_even_depth(ps2, rngseed=1, sample.size=12351, replace=F)
sample_data(ps2.rare)$treatment3<-factor(sample_data(ps2.rare)$treatment3,levels=c("Control","Chlorotha
levels(sample_data(ps2.rare)$treatment3)
```

```
## [1] "Control" "Chloro"  "Glypho"  "Tetra"
```

```
cycle1 = subset_samples(ps2.rare, cycle == "cycle_one")
ps3 <- tax_glom(cycle1, taxrank = 'Genus')
psord = subset_taxa(ps3, Genus=="Lactobacillus")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  -475 [95CI  -1410; 478]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=12)
##  22 [95CI  -898; 977]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=3)
##  2970 [95CI  2230; 3740]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Lactobacillus_cycle1<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",tick.fontsize=17,axes.title
```

```
#cycle2
cycle2 = subset_samples(ps2.rare, cycle == "cycle_two")
ps3 <- tax_glom(cycle2, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Lactobacillus")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  -176 [95CI  -1510; 834]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=11)
##  284 [95CI  -1190; 1410]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=12)
##  5180 [95CI  3560; 6590]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Lactobacillus_cycle2<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff
```

```
#cycle3
cycle3 = subset_samples(ps2.rare, cycle == "cycle_three_before_stress")
```

```
ps3 <- tax_glom(cycle3, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Lactobacillus")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=27) minus Control (n=9)
##  -475 [95CI  -1100; 58.8]
##
## Unpaired mean difference of Glypho (n=27) minus Control (n=9)
##  -699 [95CI  -1280; -242]
##
## Unpaired mean difference of Tetra (n=27) minus Control (n=9)
##  2160 [95CI  1050; 3400]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Lactobacillus_cycle3<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff
```

Snodgrassella

```
ps2 <-ps1
set.seed(42)
ps2.rare = rarefy_even_depth(ps2, rngseed=1, sample.size=12351, replace=F)
sample_data(ps2.rare)$treatment3<-factor(sample_data(ps2.rare)$treatment3,levels=c("Control","Chlorotha
levels(sample_data(ps2.rare)$treatment3)
```

```
## [1] "Control" "Chloro"  "Glypho"  "Tetra"
```

```
cycle1 = subset_samples(ps2.rare, cycle == "cycle_one")
ps3 <- tax_glom(cycle1, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Snodgrassella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
```

```
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  -30.1 [95CI  -362; 312]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=12)
##  -1020 [95CI  -1400; -412]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=3)
##  -1520 [95CI  -1690; -1360]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Snodgrassella_cycle1<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",tick.fontsize=17,axes.title

#cycle2
cycle2 = subset_samples(ps2.rare, cycle == "cycle_two")
ps3 <- tax_glom(cycle2, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Snodgrassella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  314 [95CI  -115; 714]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=11)
##  -954 [95CI  -1280; -664]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=12)
##  -1330 [95CI  -1640; -1060]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Snodgrassella_cycle2<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff

#cycle3
cycle3 = subset_samples(ps2.rare, cycle == "cycle_three_before_stress")
ps3 <- tax_glom(cycle3, taxrank = 'Genus')
```

```
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Snodgrassella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=27) minus Control (n=9)
##  -33.4 [95CI  -488; 473]
##
## Unpaired mean difference of Glypho (n=27) minus Control (n=9)
##  -273 [95CI  -627; 21.2]
##
## Unpaired mean difference of Tetra (n=27) minus Control (n=9)
##  -1210 [95CI  -1460; -995]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Snodgrassella_cycle3<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff
```

Gilliamella

```
ps2 <-ps1
sample_data(ps2)$treatment3<-factor(sample_data(ps2)$treatment3,levels=c("Control","Chlorothalonil","Gly
levels(sample_data(ps2)$treatment3)
```

```
## [1] "Control" "Chloro"  "Glypho"  "Tetra"
```

```
cycle1 = subset_samples(ps2, cycle == "cycle_one")
ps3 <- tax_glom(cycle1, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Gilliamella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
```

```
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  936 [95CI  178; 1670]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=12)
##  887 [95CI  61.6; 1580]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=3)
##  9670 [95CI  6870; 14600]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Gilliamella_cycle1<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",tick.fontsize=15.5,axes.title

#cycle2
cycle2 = subset_samples(ps2, cycle == "cycle_two")
ps3 <- tax_glom(cycle2, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Gilliamella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  9.75 [95CI  -722; 892]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=11)
##  1080 [95CI  49.1; 2080]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=12)
##  7010 [95CI  3780; 10200]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Gilliamella_cycle2<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff."

#cycle3
cycle3 = subset_samples(ps2, cycle == "cycle_three_before_stress")
ps3 <- tax_glom(cycle3, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Gilliamella")
melt<-psmelt(psord)
```

```
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=27) minus Control (n=9)
##  -352 [95CI  -890; 177]
##
## Unpaired mean difference of Glypho (n=27) minus Control (n=9)
##  2530 [95CI  384; 5550]
##
## Unpaired mean difference of Tetra (n=27) minus Control (n=9)
##  9680 [95CI  6630; 12900]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Gilliamella_cycle3<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff."
```

Bifidobacterium

```
ps2 <-ps1
set.seed(42)
ps2.rare = rarefy_even_depth(ps2, rngseed=1, sample.size=12351, replace=F)
sample_data(ps2.rare)$treatment3<-factor(sample_data(ps2.rare)$treatment3,levels=c("Control","Chlorotha
levels(sample_data(ps2.rare)$treatment3)
```

```
## [1] "Control" "Chloro"  "Glypho"  "Tetra"
```

```
cycle1 = subset_samples(ps2.rare, cycle == "cycle_one")
ps3 <- tax_glom(cycle1, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Bifidobacterium")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
```

```
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  -2.17 [95CI  -266; 287]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=12)
##  82.2 [95CI  -226; 391]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=3)
##  22.3 [95CI  -407; 350]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Bifidobacterium_cycle1<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",tick.fontsize=17,axes.titl

#cycle2
cycle2 = subset_samples(ps2.rare, cycle == "cycle_two")
ps3 <- tax_glom(cycle2, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Bifidobacterium")

#Melt and plot
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  -52 [95CI  -306; 171]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=11)
##  -17.6 [95CI  -343; 287]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=12)
##  -302 [95CI  -567; -80.8]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Bifidobacterium_cycle2<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean di

#cycle3
cycle3 = subset_samples(ps2.rare, cycle == "cycle_three_before_stress")
ps3 <- tax_glom(cycle3, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Bifidobacterium")
```

```
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=27) minus Control (n=9)
##  -12.7 [95CI  -142; 116]
##
## Unpaired mean difference of Glypho (n=27) minus Control (n=9)
##  -160 [95CI  -294; -36.8]
##
## Unpaired mean difference of Tetra (n=27) minus Control (n=9)
##  139 [95CI  -78.9; 411]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Bifidobacterium_cycle3<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean di
```

Frischella

```
ps2 <-ps1
set.seed(42)
ps2.rare = rarefy_even_depth(ps2, rngseed=1, sample.size=12351, replace=F)

sample_data(ps2.rare)$treatment3<-factor(sample_data(ps2.rare)$treatment3,levels=c("Control","Chlorotha
levels(sample_data(ps2.rare)$treatment3)
```

```
## [1] "Control" "Chloro"  "Glypho"  "Tetra"
```

```
cycle1 = subset_samples(ps2.rare, cycle == "cycle_one")
ps3 <- tax_glom(cycle1, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Frischella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
```

```
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  -369 [95CI  -672; -65.2]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=12)
##  -212 [95CI  -659; 299]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=3)
##  -798 [95CI  -1000; -562]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Frischella_cycle1<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",tick.fontsize=17,axes.title.fo
```

```
#cycle2
cycle2 = subset_samples(ps2.rare, cycle == "cycle_two")
ps3 <- tax_glom(cycle2, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Frischella")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  17.4 [95CI  -410; 464]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=11)
##  -57.4 [95CI  -547; 388]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=12)
##  -718 [95CI  -1060; -426]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Frischella_cycle2<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff.",
```

```
#cycle3
cycle3 = subset_samples(ps2.rare, cycle == "cycle_three_before_stress")
ps3 <- tax_glom(cycle3, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Frischella")
```

```
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=27) minus Control (n=9)
##  119 [95CI  -306; 552]
##
## Unpaired mean difference of Glypho (n=27) minus Control (n=9)
##  -437 [95CI  -792; -66.1]
##
## Unpaired mean difference of Tetra (n=27) minus Control (n=9)
##  -858 [95CI  -1090; -679]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Frischella_cycle3<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean diff.",
```

Commensalibacter

```
ps2 <-ps1
set.seed(42)
ps2.rare = rarefy_even_depth(ps2, rngseed=1, sample.size=12351, replace=F)

sample_data(ps2.rare)$treatment3<-factor(sample_data(ps2.rare)$treatment3,levels=c("Control","Chlorotha
levels(sample_data(ps2.rare)$treatment3)
```

```
## [1] "Control" "Chloro"  "Glypho"  "Tetra"
```

```
cycle1 = subset_samples(ps2.rare, cycle == "cycle_one")
ps3 <- tax_glom(cycle1, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Commensalibacter")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
```

```
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  2 [95CI  -320; 287]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=12)
##  489 [95CI  16.3; 941]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=3)
##  -1060 [95CI  -1270; -837]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Commensalibacter_cycle1<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",tick.fontsize=17,axes.ti

#cycle2
cycle2 = subset_samples(ps2.rare, cycle == "cycle_two")
ps3 <- tax_glom(cycle2, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Commensalibacter")
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=12) minus Control (n=12)
##  134 [95CI  -153; 378]
##
## Unpaired mean difference of Glypho (n=12) minus Control (n=11)
##  337 [95CI  -77.5; 743]
##
## Unpaired mean difference of Tetra (n=12) minus Control (n=12)
##  -334 [95CI  -565; -162]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Commensalibacter_cycle2<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean di

#cycle3
cycle3 = subset_samples(ps2.rare, cycle == "cycle_three_before_stress")
ps3 <- tax_glom(cycle3, taxrank = 'Genus')
ps3 <- prune_taxa(taxa_sums(ps3) > 0, ps3)
psord = subset_taxa(ps3, Genus=="Commensalibacter")
```

```
melt<-psmelt(psord)
unpaired_mean_diff <- dabest(melt, treatment3,Abundance,
                             idx = c("Control", "Chloro", "Glypho","Tetra"),
                             paired = FALSE)
unpaired_mean_diff
```

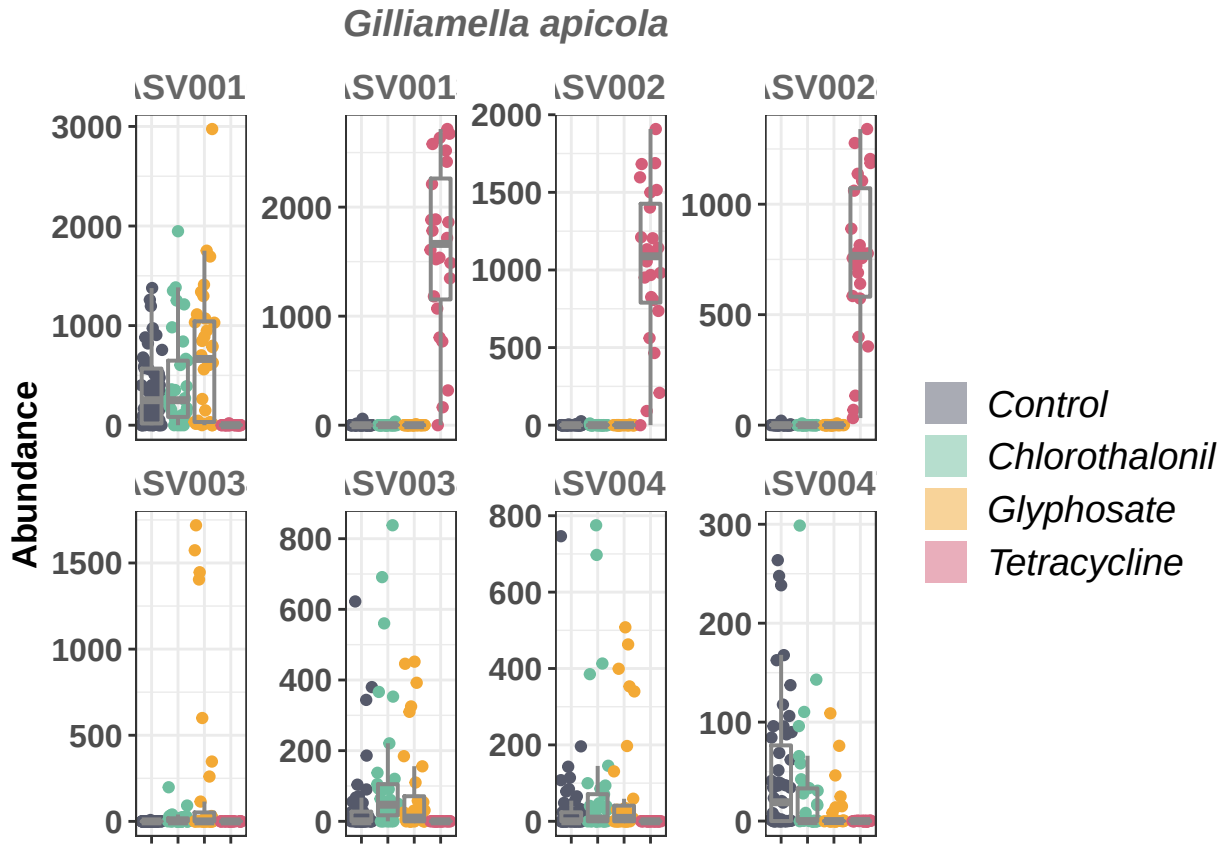
```
## DABEST (Data Analysis with Bootstrap Estimation) v0.2.1.9000
## ============================================================
##
## Variable: Abundance
##
## Unpaired mean difference of Chloro (n=27) minus Control (n=9)
##  2.81 [95CI  -282; 317]
##
## Unpaired mean difference of Glypho (n=27) minus Control (n=9)
##  -188 [95CI  -439; 28.2]
##
## Unpaired mean difference of Tetra (n=27) minus Control (n=9)
##  -534 [95CI  -741; -381]
##
##
## 5000 bootstrap resamples.
## All confidence intervals are bias-corrected and accelerated.
```

```
Commensalibacter_cycle3<-plot(unpaired_mean_diff, rawplot.ylabel = "Abundance",effsize.ylabel = "Mean di
```

have a look on most abundant ASVs in the core taxa

```
tax<-ps1
tax = rarefy_even_depth(tax, rngseed=1, sample.size=12351, replace=F)
ps2 <- subset_samples(tax, cycle =="cycle_one"|cycle =="cycle_two"|cycle =="cycle_three_before_stress")
ps2 <- subset_samples(ps2, treatment3 =="Control"|treatment3 =="Chlorothalonil"|treatment3 =="Glyphosate
ps2 <- subset_taxa(ps2, Genus =="Gilliamella")

ps2.3 <- prune_taxa(taxa_sums(ps2) > 1000, ps2)
ps2.4 <- subset_taxa(ps2.3, ASV !="ASV0068" & ASV !="ASV0070")

ps2m <- psmelt(ps2.4)

neworder <- c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
ps_df_sum2 <- arrange(transform(ps2m, cycle=factor(cycle,levels=neworder,labels=change)),cycle)

neworder2 <- c("Control","Chlorothalonil","Glyphosate", "Tetracycline")
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2)),treatment3)

p<-ggplot(data = ps_df_sum3, aes(x = treatment3, y = Abundance, colour = treatment3)) +
  geom_point(position = position_jitter())+geom_boxplot(show.legend=FALSE, alpha = 0,colour="grey53",siz
p<-p + facet_wrap(~ASV, scales="free_y",nrow=2)
p<-p + theme(strip.text = element_text(size=13,face="bold",color="grey40"))
taxa2<-p+theme(axis.title.y = element_text(size=13, face="bold"))+theme(axis.text.y = element_text(size=
```

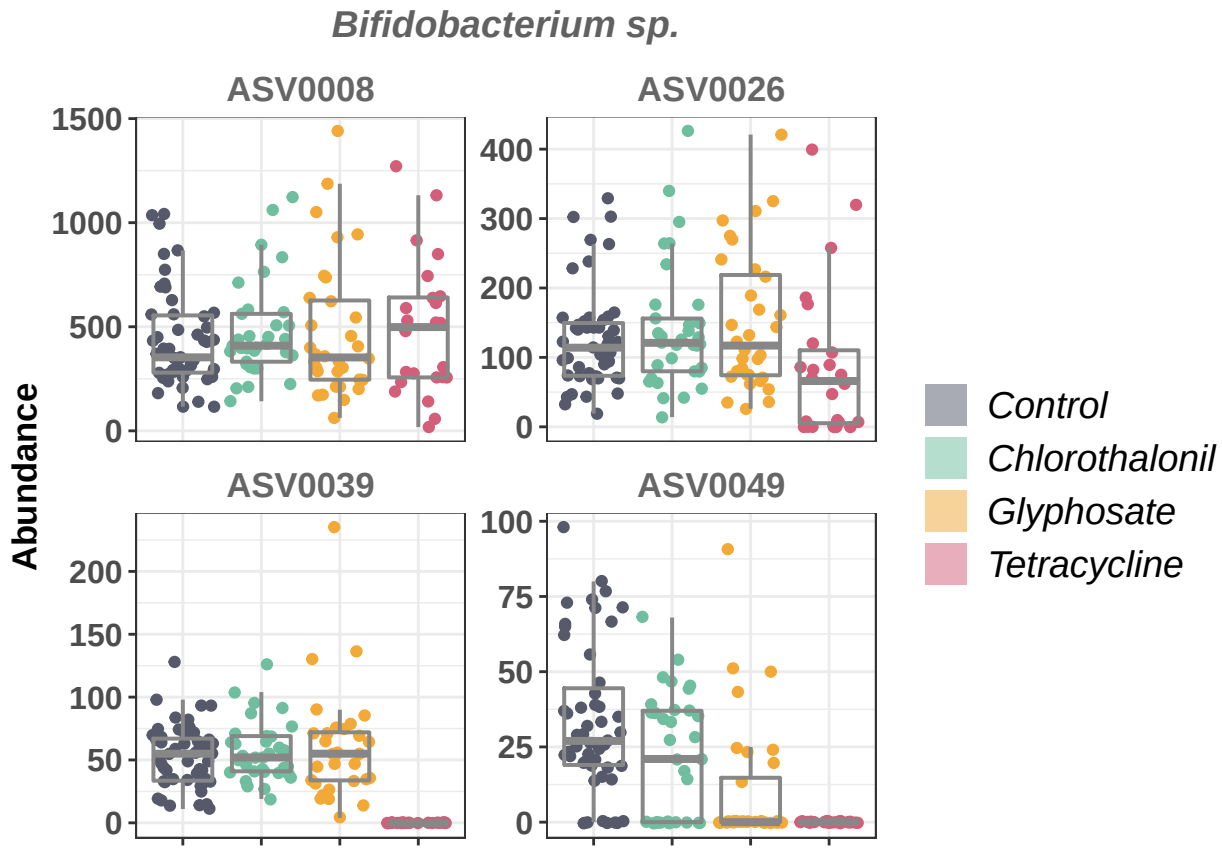
```
taxa2<- taxa2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = el
taxa2+ggtitle("Gilliamella apicola")+theme(plot.title = element_text(hjust = 0.5, size=14, face="bold.it
```




```
ggsave("R_microbiome_figures/ASV_Gilliamella.png", height = 4.5, width = 10)
```

```
tax<-ps1
tax = rarefy_even_depth(tax, rngseed=1, sample.size=12351, replace=F)
ps2 <- subset_samples(tax, cycle =="cycle_one"|cycle =="cycle_two"|cycle =="cycle_three_before_stress")
ps2 <- subset_samples(ps2, treatment3 =="Control"|treatment3 =="Chlorothalonil"|treatment3 =="Glyphosat
ps2 <- subset_taxa(ps2, Genus =="Bifidobacterium")

ps2.3 <- prune_taxa(taxa_sums(ps2) > 1000, ps2)
ps2m <- psmelt(ps2.3)

neworder <- c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
ps_df_sum2 <- arrange(transform(ps2m, cycle=factor(cycle,levels=neworder,labels=change)),cycle)

neworder2 <- c("Control","Chlorothalonil","Glyphosate", "Tetracycline")
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2)),treatment3)

p<-ggplot(data = ps_df_sum3, aes(x = treatment3, y = Abundance, colour = treatment3)) +
  geom_point(position = position_jitter())+geom_boxplot(show.legend=FALSE, alpha = 0,colour="grey53",si
p<-p + facet_wrap(~ASV, scales="free_y",nrow=2)
p<-p + theme(strip.text = element_text(size=13,face="bold",color="grey40"))
```

```
taxa2<-p+theme(axis.title.y = element_text(size=13, face="bold"))+theme(axis.text.y = element_text(size=
taxa2<- taxa2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = el
taxa2+ggtitle("Bifidobacterium sp.")+theme(plot.title = element_text(hjust = 0.5, size=14, face="bold.it
```

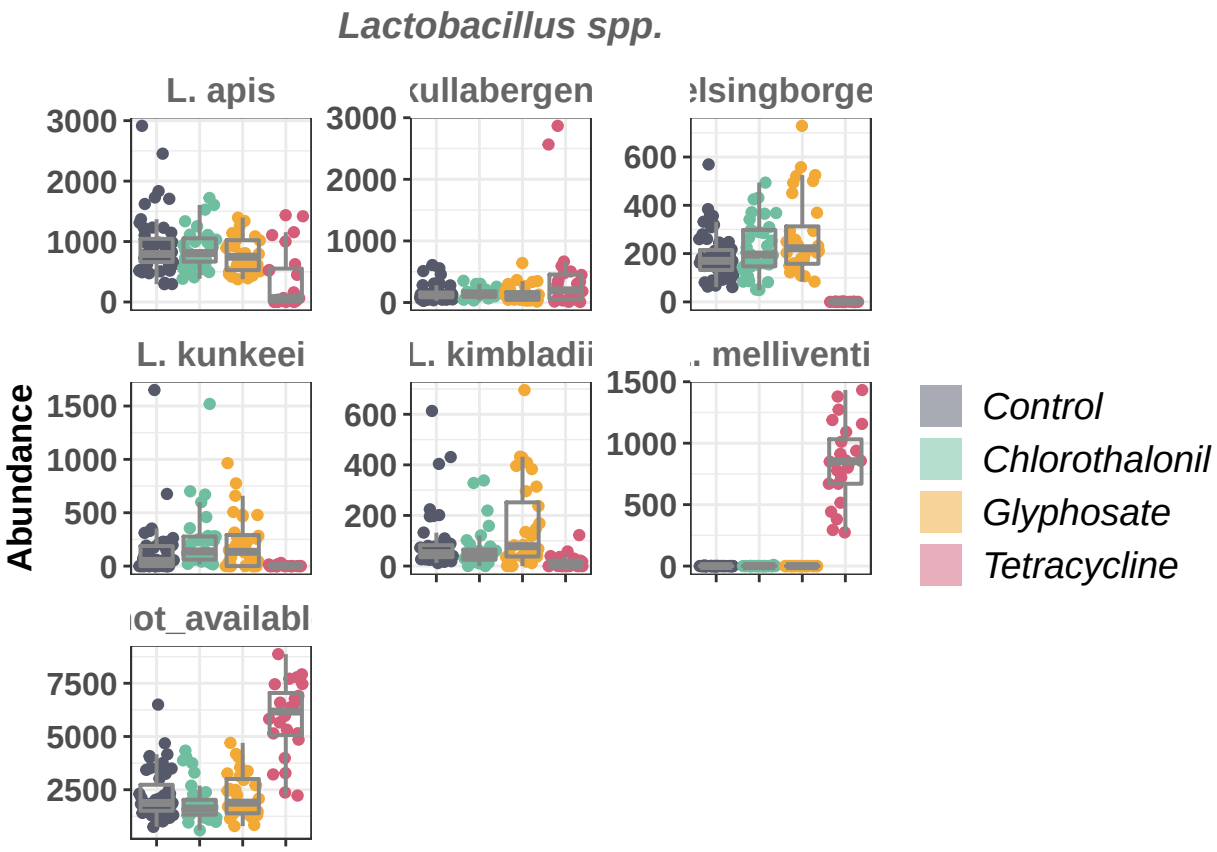



```
ggsave("R_microbiome_figures/ASV_Bifidobacterium.png", height = 4.5, width = 7)
```

species Lactobacillus

```
tax<-ps1
tax = rarefy_even_depth(tax, rngseed=1, sample.size=12351, replace=F)
ps2 <- subset_samples(tax, cycle =="cycle_one"|cycle =="cycle_two"|cycle =="cycle_three_before_stress")
ps2 <- subset_samples(ps2, treatment3 =="Control"|treatment3 =="Chlorothalonil"|treatment3 =="Glyphosate
ps2 <- subset_taxa(ps2, Genus =="Lactobacillus")
ps2 <- prune_taxa(taxa_sums(ps2) > 1000, ps2)
ps2 <- tax_glom(ps2, taxrank = 'Species')
ps2m <- psmelt(ps2)

neworder <- c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
ps_df_sum2 <- arrange(transform(ps2m, cycle=factor(cycle,levels=neworder,labels=change)),cycle)

neworder2 <- c("Control","Chlorothalonil","Glyphosate", "Tetracycline")
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2)),treatment3)
```

```
neworder3 <- c("Lactobacillus apis","Lactobacillus kullabergensis","Lactobacillus helsingborgensis","La
change3 <- c("L. apis","L. kullabergensis","L. helsingborgensis","L. kunkeei","L. kimbladii","L. melliv
ps_df_sum4 <- arrange(transform(ps_df_sum3, Species=factor(Species,levels=neworder3,labels=change3)),Sp

p<-ggplot(data = ps_df_sum4, aes(x = treatment3, y = Abundance, colour = treatment3)) +
  geom_point(position = position_jitter())+geom_boxplot(show.legend=FALSE, alpha = 0,colour="grey53",si
p<-p + facet_wrap(~Species, scales="free_y",nrow=3)
p<-p + theme(strip.text = element_text(size=13,face="bold",color="grey40"))
taxa2<-p+theme(axis.title.y = element_text(size=13, face="bold"))+theme(axis.text.y = element_text(size
taxa2<- taxa2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = el
taxa2+ggtitle("Lactobacillus spp.")+theme(plot.title = element_text(hjust = 0.5, size=14, face="bold.ita
```

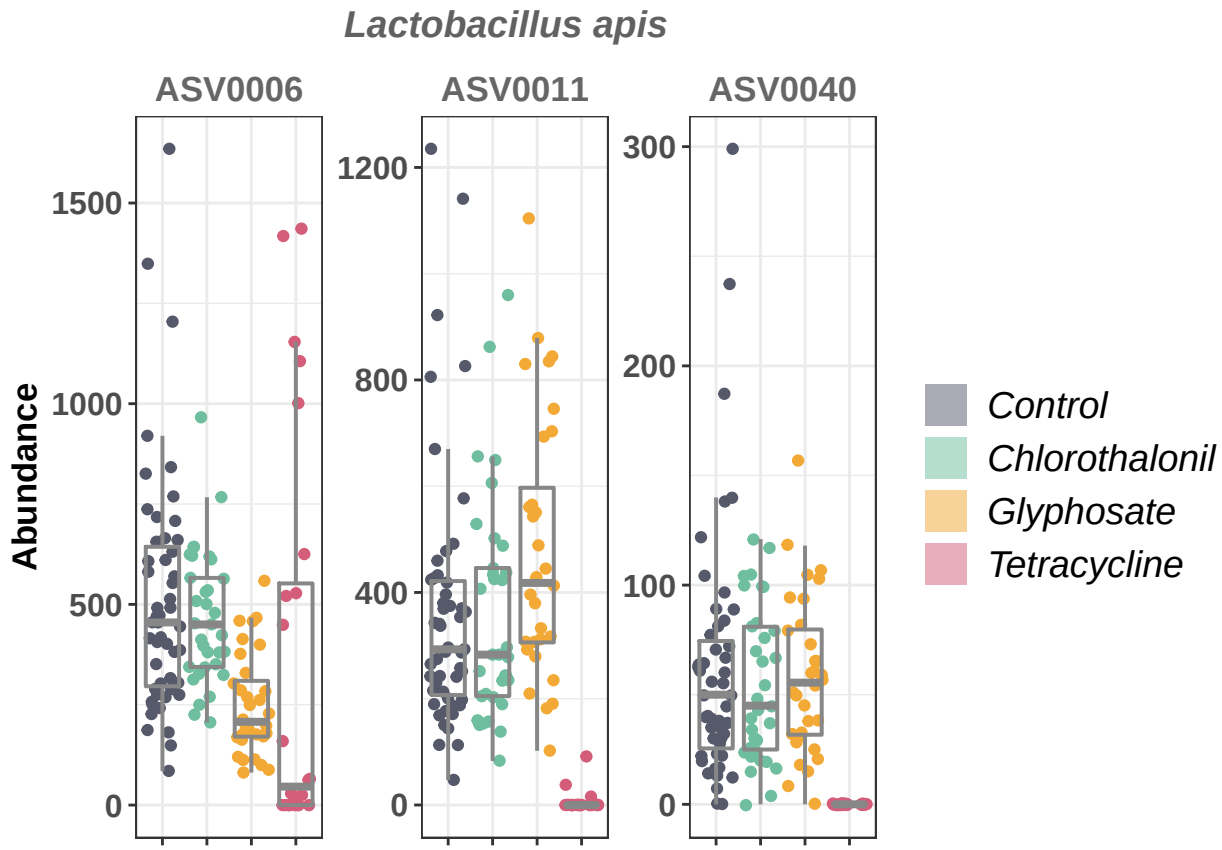



```
ggsave("R_microbiome_figures/Lactobacillus spp.png", height = 7, width = 11)
```

ASV Lactobacillus

```
tax<-ps1
tax = rarefy_even_depth(tax, rngseed=1, sample.size=12351, replace=F)
ps2 <- subset_samples(tax, cycle =="cycle_one"|cycle =="cycle_two"|cycle =="cycle_three_before_stress")
ps2 <- subset_samples(ps2, treatment3 =="Control"|treatment3 =="Chlorothalonil"|treatment3 =="Glyphosat
ps2 <- subset_taxa(ps2, Genus =="Lactobacillus")
ps2 <- subset_taxa(ps2, Species =="Lactobacillus apis")
```

```
ps2.3 <- prune_taxa(taxa_sums(ps2) > 1000, ps2)
ps2.3 <- prune_taxa(taxa_sums(ps2.3) > 0, ps2.3)
ps2m <- psmelt(ps2.3)
neworder <- c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
ps_df_sum2 <- arrange(transform(ps2m, cycle=factor(cycle,levels=neworder,labels=change)),cycle)
neworder2 <- c("Control","Chlorothalonil","Glyphosate", "Tetracycline")
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2)),treatment3)
p<-ggplot(data = ps_df_sum3, aes(x = treatment3, y = Abundance, colour = treatment3)) +
  geom_point(position = position_jitter())+geom_boxplot(show.legend=FALSE, alpha = 0,colour="grey53",si
p<-p + facet_wrap(~ASV, scales="free_y",nrow=1)
p<-p + theme(strip.text = element_text(size=13,face="bold",color="grey40"))
taxa2<-p+theme(axis.title.y = element_text(size=13, face="bold"))+theme(axis.text.y = element_text(size
taxa2<- taxa2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = el
taxa2+ggtitle("Lactobacillus apis")+theme(plot.title = element_text(hjust = 0.5, size=14, face="bold.ita
```

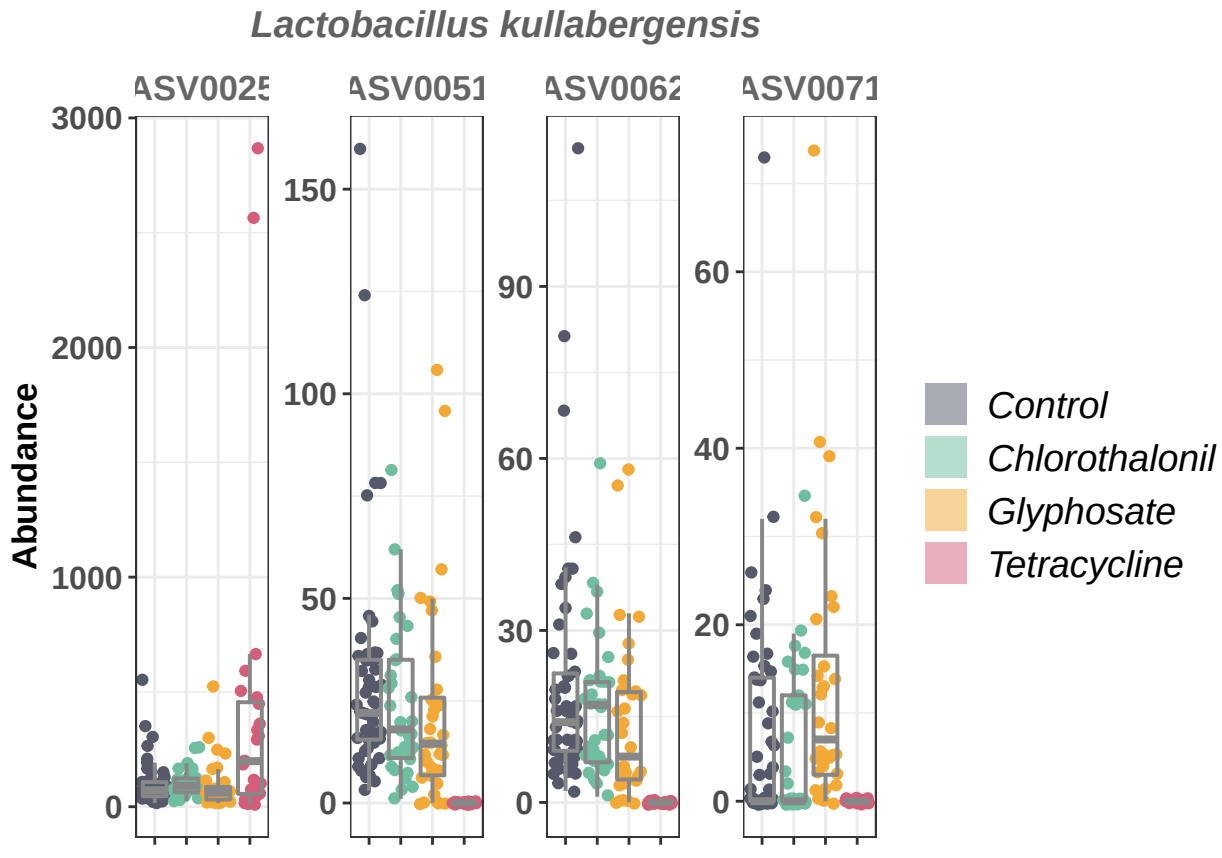



```
ggsave("R_microbiome_figures/ASV_Lactobacillus apis.png", height = 3, width = 8)
```

Lactobacillus kullabergensis

```
tax<-ps1
tax = rarefy_even_depth(tax, rngseed=1, sample.size=12351, replace=F)
```

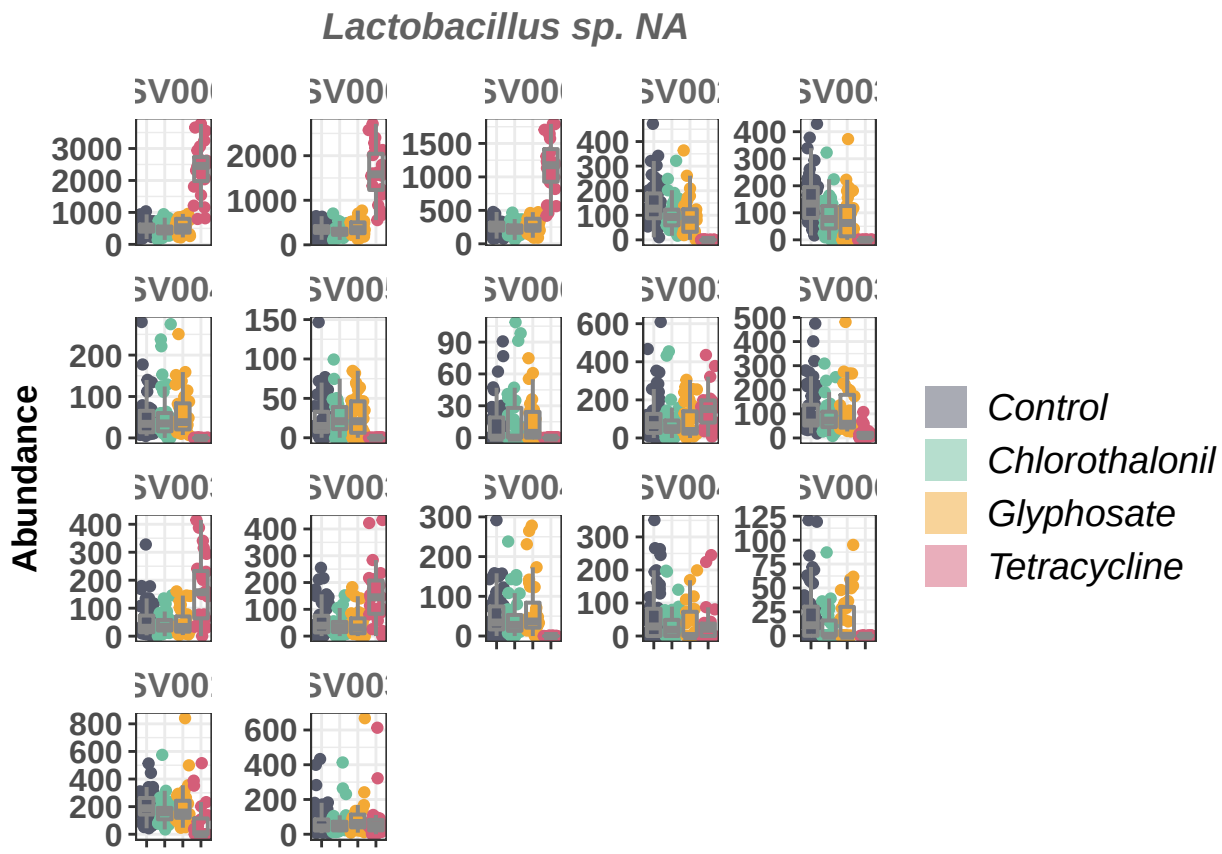
```
ps2 <- subset_samples(tax, cycle =="cycle_one"|cycle =="cycle_two"|cycle =="cycle_three_before_stress")
ps2 <- subset_samples(ps2, treatment3 =="Control"|treatment3 =="Chlorothalonil"|treatment3 =="Glyphosat
ps2 <- subset_taxa(ps2, Genus =="Lactobacillus")
ps2 <- subset_taxa(ps2, Species =="Lactobacillus kullabergensis")
ps2.3 <- prune_taxa(taxa_sums(ps2) > 1000, ps2)
ps2.3 <- prune_taxa(taxa_sums(ps2.3) > 0, ps2.3)
ps2m <- psmelt(ps2.3)
neworder <- c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
ps_df_sum2 <- arrange(transform(ps2m, cycle=factor(cycle,levels=neworder,labels=change)),cycle)

neworder2 <- c("Control","Chlorothalonil","Glyphosate", "Tetracycline")
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2)),treatment3)

p<-ggplot(data = ps_df_sum3, aes(x = treatment3, y = Abundance, colour = treatment3)) +
  geom_point(position = position_jitter())+geom_boxplot(show.legend=FALSE, alpha = 0,colour="grey53",si
p<-p + facet_wrap(~ASV, scales="free_y",nrow=1)
p<-p + theme(strip.text = element_text(size=13,face="bold",color="grey40"))
taxa2<-p+theme(axis.title.y = element_text(size=13, face="bold"))+theme(axis.text.y = element_text(size
taxa2<- taxa2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = el
taxa2+ggtitle("Lactobacillus kullabergensis")+theme(plot.title = element_text(hjust = 0.5, size=14, face
```




```
ggsave("R_microbiome_figures/ASV_Lactobacillus kullabergensis.png", height = 3, width = 8)
```

Lactobacillus NA

```
tax<-ps1
tax = rarefy_even_depth(tax, rngseed=1, sample.size=12351, replace=F)
ps2 <- subset_samples(tax, cycle =="cycle_one"|cycle =="cycle_two"|cycle =="cycle_three_before_stress")
ps2 <- subset_samples(ps2, treatment3 =="Control"|treatment3 =="Chlorothalonil"|treatment3 =="Glyphosat
ps2 <- subset_taxa(ps2, Genus =="Lactobacillus")
ps2 <- subset_taxa(ps2, Species =="not_available")
ps2 <- subset_taxa(ps2, ASV !="ASV0069")
ps2.3 <- prune_taxa(taxa_sums(ps2) > 1000, ps2)
ps2m <- psmelt(ps2.3)
neworder <- c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
ps_df_sum2 <- arrange(transform(ps2m, cycle=factor(cycle,levels=neworder,labels=change)),cycle)
neworder2 <- c("Control","Chlorothalonil","Glyphosate", "Tetracycline")
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2)),treatment3)
neworder3 <- c("ASV0004","ASV0005","ASV0009", "ASV0027","ASV0031","ASV0045","ASV0052","ASV0067","ASV003
ps_df_sum4 <- arrange(transform(ps_df_sum3, ASV=factor(ASV,levels=neworder3)),ASV)

p<-ggplot(data = ps_df_sum4, aes(x = treatment3, y = Abundance, colour = treatment3)) +
  geom_point(position = position_jitter())+geom_boxplot(show.legend=FALSE, alpha = 0,colour="grey53",si
p<-p + facet_wrap(~ASV, scales="free_y",nrow=4)
p<-p + theme(strip.text = element_text(size=13,face="bold",color="grey40"))
taxa2<-p+theme(axis.title.y = element_text(size=13, face="bold"))+theme(axis.text.y = element_text(size
taxa2<- taxa2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = el
taxa2+ggtitle("Lactobacillus sp. NA")+theme(plot.title = element_text(hjust = 0.5, size=14, face="bold.
```

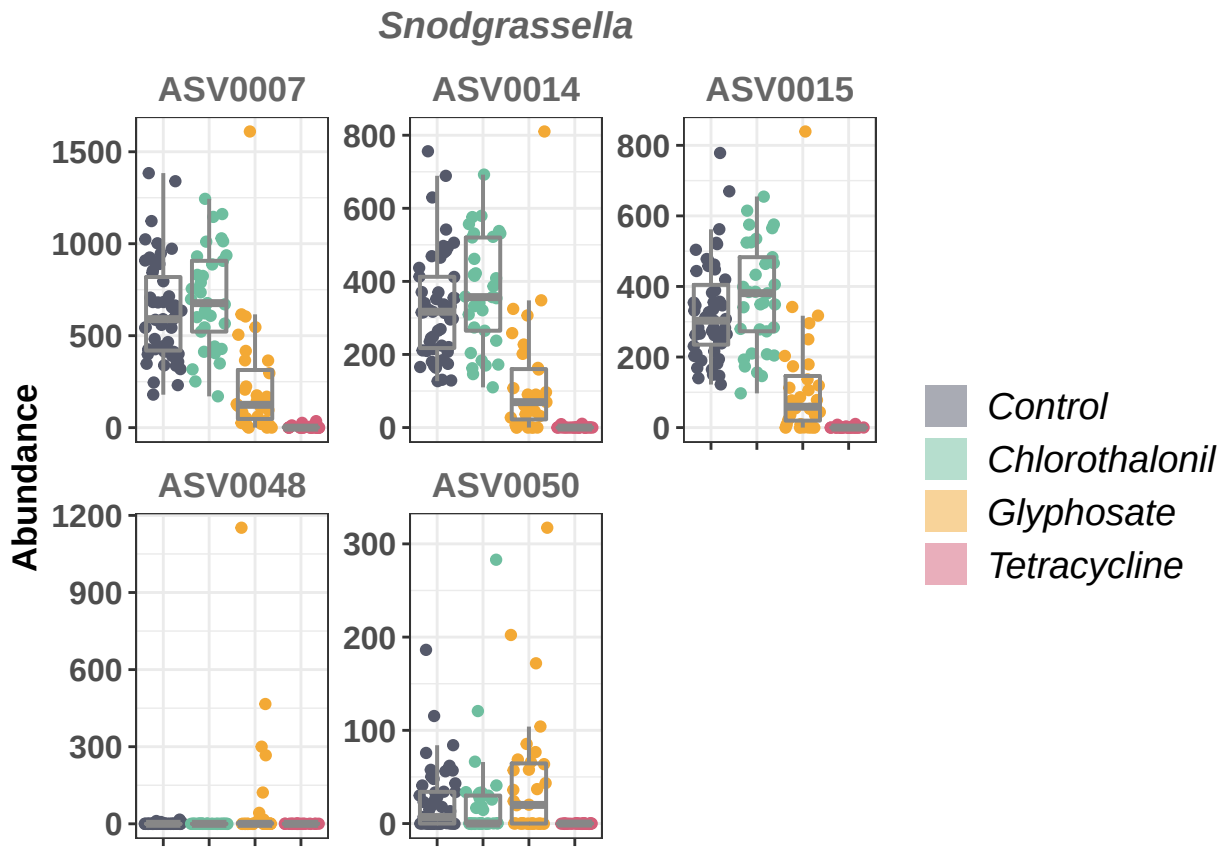



```
ggsave("R_microbiome_figures/ASV_Lactobacillus sp. NA.png", height = 8, width = 13)
```

Snodgrassella

```
tax<-ps1
tax = rarefy_even_depth(tax, rngseed=1, sample.size=12351, replace=F)
ps2 <- subset_samples(tax, cycle =="cycle_one"|cycle =="cycle_two"|cycle =="cycle_three_before_stress")
ps2 <- subset_samples(ps2, treatment3 =="Control"|treatment3 =="Chlorothalonil"|treatment3 =="Glyphosat
ps2 <- subset_taxa(ps2, Genus =="Snodgrassella")
ps2.3 <- prune_taxa(taxa_sums(ps2) > 1000, ps2)
ps2m <- psmelt(ps2.3)
neworder <- c("cycle_one","cycle_two","cycle_three_before_stress")
change <- c("Cycle one","Cycle two","Cycle three")
ps_df_sum2 <- arrange(transform(ps2m, cycle=factor(cycle,levels=neworder,labels=change)),cycle)

neworder2 <- c("Control","Chlorothalonil","Glyphosate", "Tetracycline")
ps_df_sum3 <- arrange(transform(ps_df_sum2, treatment3=factor(treatment3,levels=neworder2)),treatment3)

p<-ggplot(data = ps_df_sum3, aes(x = treatment3, y = Abundance, colour = treatment3)) +
  geom_point(position = position_jitter())+geom_boxplot(show.legend=FALSE, alpha = 0,colour="grey53",si
p<-p + facet_wrap(~ASV, scales="free_y",nrow=2)
p<-p + theme(strip.text = element_text(size=13,face="bold",color="grey40"))
taxa2<-p+theme(axis.title.y = element_text(size=13, face="bold"))+theme(axis.text.y = element_text(size
```

```
taxa2<- taxa2 + theme(legend.title = element_blank(),legend.background = element_blank(),legend.key = el
taxa2+ggtitle("Snodgrassella")+theme(plot.title = element_text(hjust = 0.5, size=14, face="bold.italic"
```




```
ggsave("R_microbiome_figures/ASV_Snodgrassella.png", height = 6, width = 10.5)
```