

Beta diversity: the phylum stacked barplot and genus heatmaps

Stijn Schreven

4 March 2021

Contents

Load packages	1
Input files	2
1. Prepare data	2
1.1. Plot presets	3
2. Barplot phylum	4
2.1. Prepare data	4
2.2. Confidence intervals	4
2.3. Plot	5
3. Heatmaps	6
3.1. Prepare data	6
3.2. Chicken feed	7
3.3. Chicken manure	8
3.4. Both diets and all genera	9

Load packages

```
library(phyloseq)
library(microbiome)
library(microbiomeutilities)
library(ggplot2)
library(viridis)
library(sciplot)
library(vegan)
library(plyr)
library(magrittr)
```

Input files

```
ps1.work <- readRDS("./phyobjects/ps1.work.rds")
print(ps1.work)
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:           [ 2222 taxa and 93 samples ]
## sample_data() Sample Data:        [ 93 samples by 16 sample variables ]
## tax_table()   Taxonomy Table:      [ 2222 taxa by 6 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree:   [ 2222 tips and 2221 internal nodes ]
```

1. Prepare data

```
ps1.com <- ps1.work

# add OTU column, remove tree
taxic <- as.data.frame(ps1.com@tax_table)
taxic$OTU <- rownames(taxic)
tax_table(ps1.com) <- tax_table(as.matrix(taxic))
ps1.com.bh <- format_to_besthit(ps1.com)

# add best hit, keep tree
taxic$best_hit <- tax_table(ps1.com.bh)[,"best_hit"]
tax_table(ps1.com) <- tax_table(as.matrix(taxic))

# aggregate top 5 phyla, without tree
ps.phylum <- aggregate_taxa(ps1.com.bh, "Phylum", top = 5)
ps.phylum <- microbiome::transform(ps.phylum, "compositional")

# aggregate to genus, with phy_tree
pstot.g <- microbiome::aggregate_taxa(ps1.work, "Genus")
pstot.g <- prune_taxa(taxa_sums(otu_table(pstot.g)) > 0, pstot.g)
pstot.g.r <- microbiome::transform(pstot.g, "compositional")

# format taxa names
# tax table with OTU column and best_hit
tot.tax <- as.data.frame(tax_table(pstot.g.r))
tot.tax$OTU <- rownames(tot.tax)
tax_table(pstot.g.r) <- tax_table(as.matrix(tot.tax))
pstot.g.bh <- format_to_besthit(pstot.g.r)
tot.tax.bh <- as.data.frame(tax_table(pstot.g.bh))
colnames(tot.tax.bh)[7] <- "OTU"
# remove pattern (OTU code) from $best_hit
tot.tax.bh$best_hit <- sub(pattern = "OTU-[0-9]*:", replacement = "",
                           tot.tax.bh$best_hit)
tot.tax.bh$best_hit <- as.factor(tot.tax.bh$best_hit)
# replace the "uncultured" with [Family]:uncultured
tot.tax.bh$best_hit2 <- ifelse(tot.tax.bh$best_hit == "uncultured",
                              yes = paste(tot.tax.bh$Family, sep = " ", "uncultured"),
```

```

    no = paste(tot.tax.bh$best_hit))
tot.tax.bh$best_hit2 <- as.factor(tot.tax.bh$best_hit2)
tot.tax.bh$best_hit2 <- revalue(tot.tax.bh$best_hit2,
                                c("k_NA" = "unassigned taxon"))
tax_table(pstot.g.bh) <- tax_table(as.matrix(tot.tax.bh))

# input for heatmaps

# select top taxa:

# chicken feed
## max(abundance) > .01 (at least 1% in a sample)
## and prevalence > .1 (10% of samples)
hm.CF <- subset_samples(pstot.g.bh, Diet == "CF")
hm.CF <- prune_taxa(taxa_sums(otu_table(hm.CF)) > 0, hm.CF)
hm.CF.otu <- as.data.frame(t(abundances(hm.CF)))
m.CF.otu <- reshape2::melt(hm.CF.otu)
colnames(m.CF.otu) <- c("OTU", "abund")
sum.CF <- ddply(m.CF.otu, .(OTU), summarise,
                max = max(abund),
                prev = sum(abund > 0)/length(abund))
top.CF <- subset(sum.CF, max > .01 & prev > .1) # 19 genera
hm.CF.top <- prune_taxa(taxa_names(hm.CF) %in% droplevels(top.CF$OTU), hm.CF)
hm.CF.top <- prune_samples(sample_sums(otu_table(hm.CF.top)) > 0, hm.CF.top)

# chicken manure
## max(abundance) > .1 (at least 10% in a sample)
## and prevalence > .1 (10% of samples)
hm.CM <- subset_samples(pstot.g.bh, Diet == "CM")
hm.CM <- prune_taxa(taxa_sums(otu_table(hm.CM)) > 0, hm.CM)
hm.CM.otu <- as.data.frame(t(abundances(hm.CM)))
m.CM.otu <- reshape2::melt(hm.CM.otu)
colnames(m.CM.otu) <- c("OTU", "abund")
sum.CM <- ddply(m.CM.otu, .(OTU), summarise,
                max = max(abund),
                prev = sum(abund > 0)/length(abund))
top.CM <- subset(sum.CM, max > .1 & prev > .1) # 27 genera
hm.CM.top <- prune_taxa(taxa_names(hm.CM) %in% droplevels(top.CM$OTU), hm.CM)
hm.CM.top <- prune_samples(sample_sums(otu_table(hm.CM.top)) > 0, hm.CM.top)

# both diets and all genera
ps.hm.all <- pstot.g.bh

```

1.1. Plot presets

```

theme_hm <- theme_bw() +
  theme(axis.text.y = element_text(colour = 'black', face = 'italic'),
        legend.key = element_blank(),
        text = element_text(size = 20),
        panel.spacing.x = unit(0, "lines"),

```

```

strip.background = element_rect(colour = "black", fill = "white"))

theme_stack <- theme_bw() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = .5),
        text = element_text(size = 20),
        panel.spacing.x = unit(0, "lines"),
        strip.background = element_rect(colour = "black", fill = "white"))

labs_stack <- as_labeller(c(
  CF = "chicken feed", CM = "chicken manure",
  larvae = "larvae", substrate = "substrate",
  "0" = "day 0", "15" = "day 15"))

labs_hm <- as_labeller(c(
  larvae = "L", substrate = "S",
  "S/E" = "S/E", "Si/E" = "Si/E", "Si/Es" = "Si/Es", "Ss/E" = "Ss/E"))

```

2. Barplot phylum

Supplementary Figure S5 in manuscript.

2.1. Prepare data

```

# extract barplot data as dataframe
plot.phyl.df <- plot_composition(ps.phylum)$data
names(plot.phyl.df)[names(plot.phyl.df) == "Sample"] <- "Description" # is sample ID
plot.phyl.df <- merge(meta(ps.phylum), plot.phyl.df, by = "Description")

# summarise to mean and SD per group
phyl <- ddpby(plot.phyl.df, .(Diet, Timepoint, Treatment, Type, OTU),
  summarise, mean = mean(Abundance), se = se(Abundance))

# reorder phyla
phyl$OTU <- revalue(phyl$OTU, c("k__NA" = "Unassigned taxon"))
phyl$OTU <- factor(phyl$OTU, levels(phyl$OTU)[c(6,2,4,5,1,3)])

```

2.2. Confidence intervals

```

# NB: upper limit is mean, so that barplot is only mean - SE
phyl <- phyl %>% mutate(lower = mean - se, upper = mean)

# Calculate stacked CI values
 #(level 2 = mean(level 1) + CI(level 2) ; level 3 = mean(1) + mean(2) + CI(level 3))
phyl[phyl$OTU=='Unassigned taxon',8:9] <- phyl[phyl$OTU=='Other',6] +
  phyl[phyl$OTU=='Unassigned taxon',8:9]
phyl[phyl$OTU=='Proteobacteria',8:9] <- phyl[phyl$OTU=='Other',6] +
  phyl[phyl$OTU=='Unassigned taxon',6] +
  phyl[phyl$OTU=='Proteobacteria',8:9]

```

```

phyl[phyl$OTU=='Firmicutes',8:9] <- phyl[phyl$OTU=='Other',6] +
  phyl[phyl$OTU=='Unassigned taxon',6] +
  phyl[phyl$OTU=='Proteobacteria',6] +
  phyl[phyl$OTU=='Firmicutes',8:9]
phyl[phyl$OTU=='Bacteroidetes',8:9] <- phyl[phyl$OTU=='Other',6] +
  phyl[phyl$OTU=='Unassigned taxon',6] +
  phyl[phyl$OTU=='Proteobacteria',6] +
  phyl[phyl$OTU=='Firmicutes',6] +
  phyl[phyl$OTU=='Bacteroidetes',8:9]
phyl[phyl$OTU=='Actinobacteria',8:9] <- phyl[phyl$OTU=='Other',6] +
  phyl[phyl$OTU=='Unassigned taxon',6] +
  phyl[phyl$OTU=='Proteobacteria',6] +
  phyl[phyl$OTU=='Firmicutes',6] +
  phyl[phyl$OTU=='Bacteroidetes',6] +
  phyl[phyl$OTU=='Actinobacteria',8:9]

```

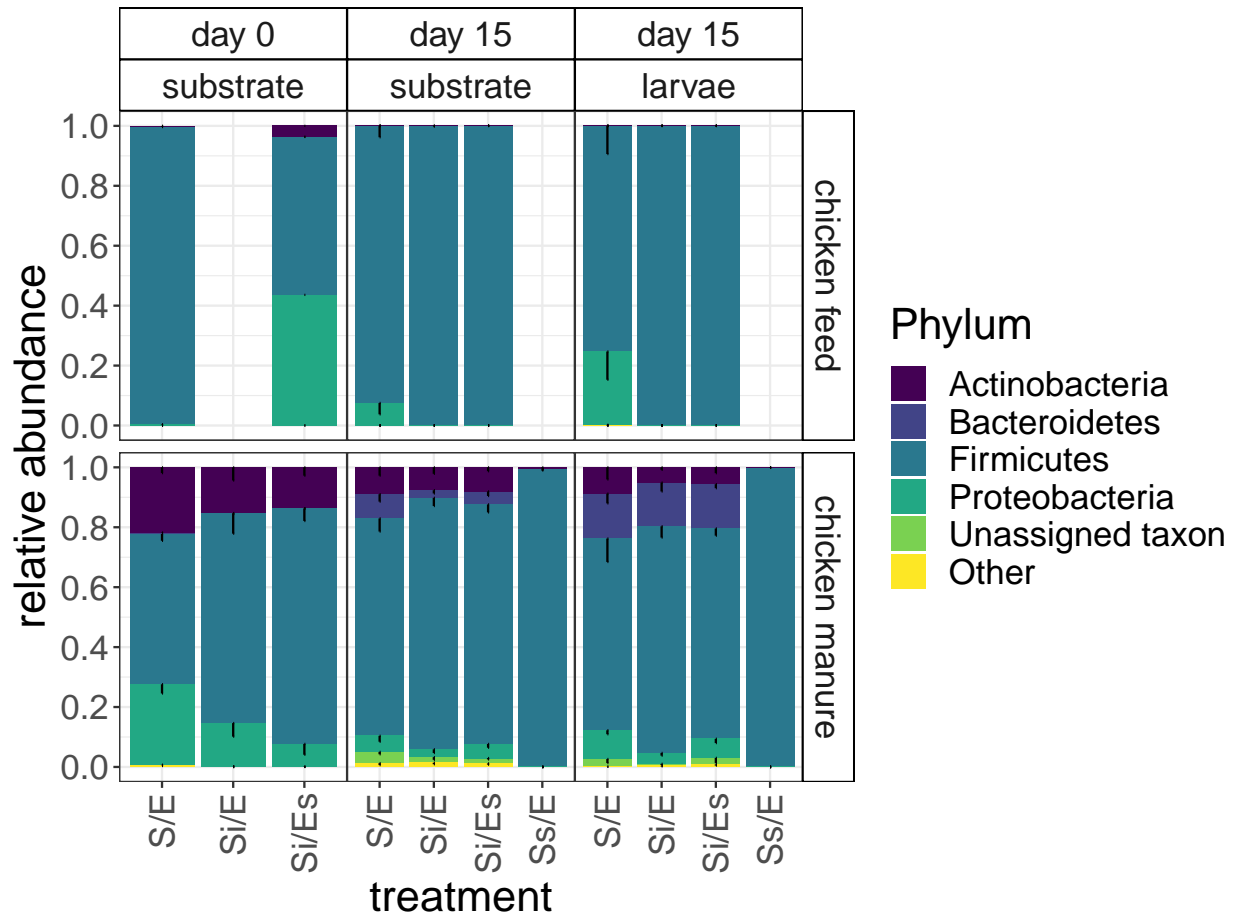
2.3. Plot

```

pbar3 <- ggplot(phyl, aes(x = Treatment, y = mean, fill = OTU)) +
  geom_bar(stat = "identity", position = "stack", width = .9) +
  scale_fill_viridis(discrete = T, option = "D") +
  scale_y_continuous(breaks = seq(0, 1, .2)) +
  labs(y = "relative abundance", x = "treatment", fill = "Phylum") +
  facet_grid(Diet ~ Timepoint + Type, scales = "free_x",
             labeller = labs_stack) +
  geom_errorbar(aes(ymax = upper, ymin = lower), width = 0) +
  theme_stack

pbar3

```



```
ggsave(plot = pbar3, "./figures/Fig_S5_phylum.png", h = 6, w = 8)
ggsave(plot = pbar3, "./figures/Fig_S5_phylum.pdf", w = 320, h = 240, u = "mm")
```

3. Heatmaps

3.1. Prepare data

Chicken feed:

```
# extract heatmap data
hm.CF.df <- plot_heatmap(hm.CF.top, method = "CAP", distance = "bray",
                        formula = ~ Treatment * Timepoint * Type)$data
# summarise per treatment (Treatment * Timepoint * Type)
hm.CF.sum <- ddply(hm.CF.df, .(Timepoint, Type, Treatment, OTU, best_hit2),
                  summarise, median = median(Abundance), mean = mean(Abundance))

# taxa clustering by Bray-Curtis
cf.hm.cast <- reshape2::dcast(hm.CF.sum,
                             best_hit2 ~ Treatment + Type + Timepoint, value.var = "mean")
rownames(cf.hm.cast) <- cf.hm.cast[,1]
cf.hm.mat <- as.matrix(cf.hm.cast[,c(2:9)])
```

```
cf.hm.dm <- vegdist(cf.hm.mat, method = "bray")

# reorder levels by clustering
cf.order <- hclust(cf.hm.dm)$order
hm.CF.sum$best_hit2 <- factor(hm.CF.sum$best_hit2, levels(hm.CF.sum$best_hit2)[cf.order])
```

Chicken manure:

```
hm.CM.df <- plot_heatmap(hm.CM.top, method = "CAP", distance = "bray",
                        formula = ~ Treatment * Timepoint * Type)$data

# add higher tax ranks to "--"
hm.CM.df$best_hit2 <- ifelse(hm.CM.df$best_hit2 == "--",
                             yes = paste(hm.CM.df$Class, "order", hm.CM.df$Order,
                                           "uncultured", sep = " "),
                             no = paste(hm.CM.df$best_hit2))
hm.CM.df$best_hit2 <- as.factor(hm.CM.df$best_hit2)

# summarise
hm.CM.sum <- ddply(hm.CM.df, .(Timepoint, Type, Treatment, OTU, best_hit2),
                  summarise, median = median(Abundance), mean = mean(Abundance))

# taxa clustering by Bray-Curtis
cm.hm.cast <- reshape2::dcast(hm.CM.sum,
                             best_hit2 ~ Treatment + Type + Timepoint, value.var = "mean")
rownames(cm.hm.cast) <- cm.hm.cast[,1]
cm.hm.mat <- as.matrix(cm.hm.cast[,c(2:12)])
cm.hm.dm <- vegdist(cm.hm.mat, method = "bray")

# reorder levels by clustering
cm.order <- hclust(cm.hm.dm)$order
hm.CM.sum$best_hit2 <- factor(hm.CM.sum$best_hit2, levels(hm.CM.sum$best_hit2)[cm.order])
```

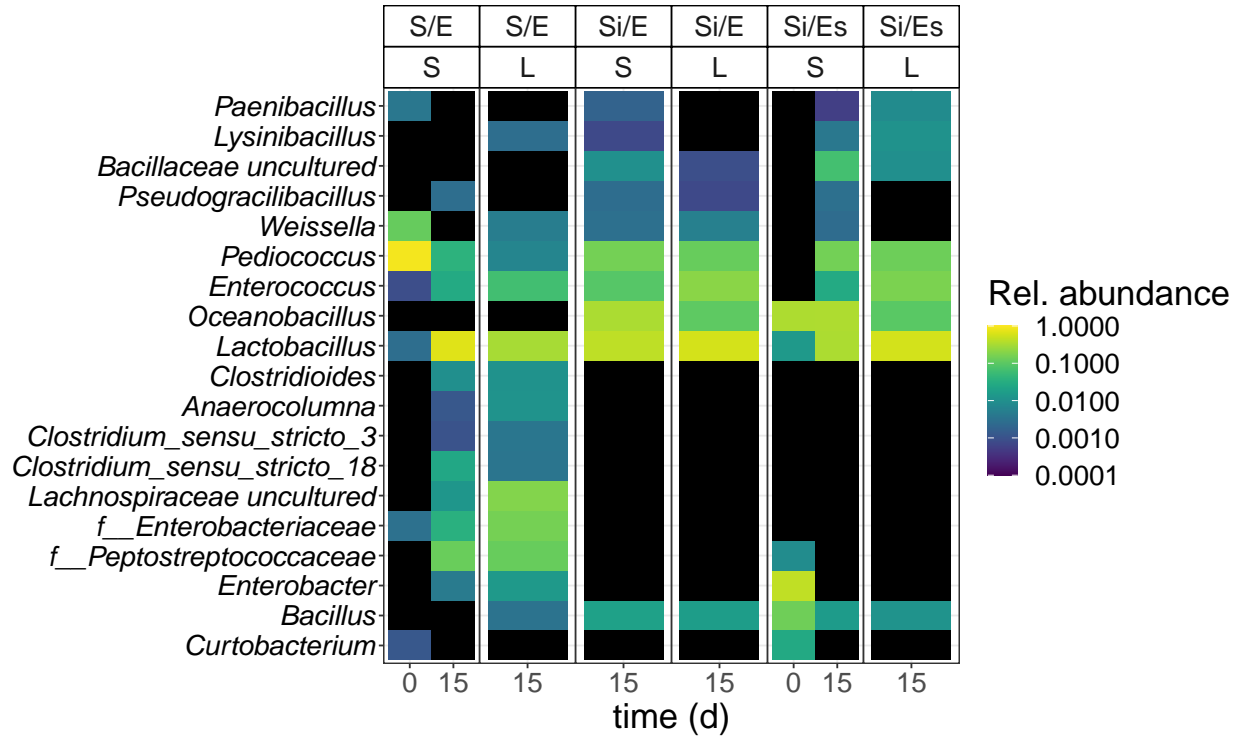
Both diets, all genera:

```
ps.hm.all.df <- plot_heatmap(ps.hm.all, method = "CAP", distance = "bray",
                           formula = ~ Diet * Treatment * Timepoint * Type)$data
ps.hm.all.sum <- ddply(ps.hm.all.df, .(Timepoint, Diet, Type, Treatment, OTU),
                      summarise, median = median(Abundance), mean = mean(Abundance))
```

3.2. Chicken feed

Figure 6A in manuscript.

```
p.hm.CF <- ggplot(hm.CF.sum, aes(x = Timepoint, y = best_hit2)) +
  geom_tile(aes(fill = mean)) +
  scale_fill_viridis("Rel. abundance", option = "D", na.value = "black",
                    trans = "log10", limits = c(.0001, 1),
                    labels = function(n){format(n, scientific = F)}) +
  facet_grid(~ Treatment + Type, scales = "free", labeller = labs_hm) +
  labs(x = "time (d)", y = NULL) + theme_hm
p.hm.CF
```

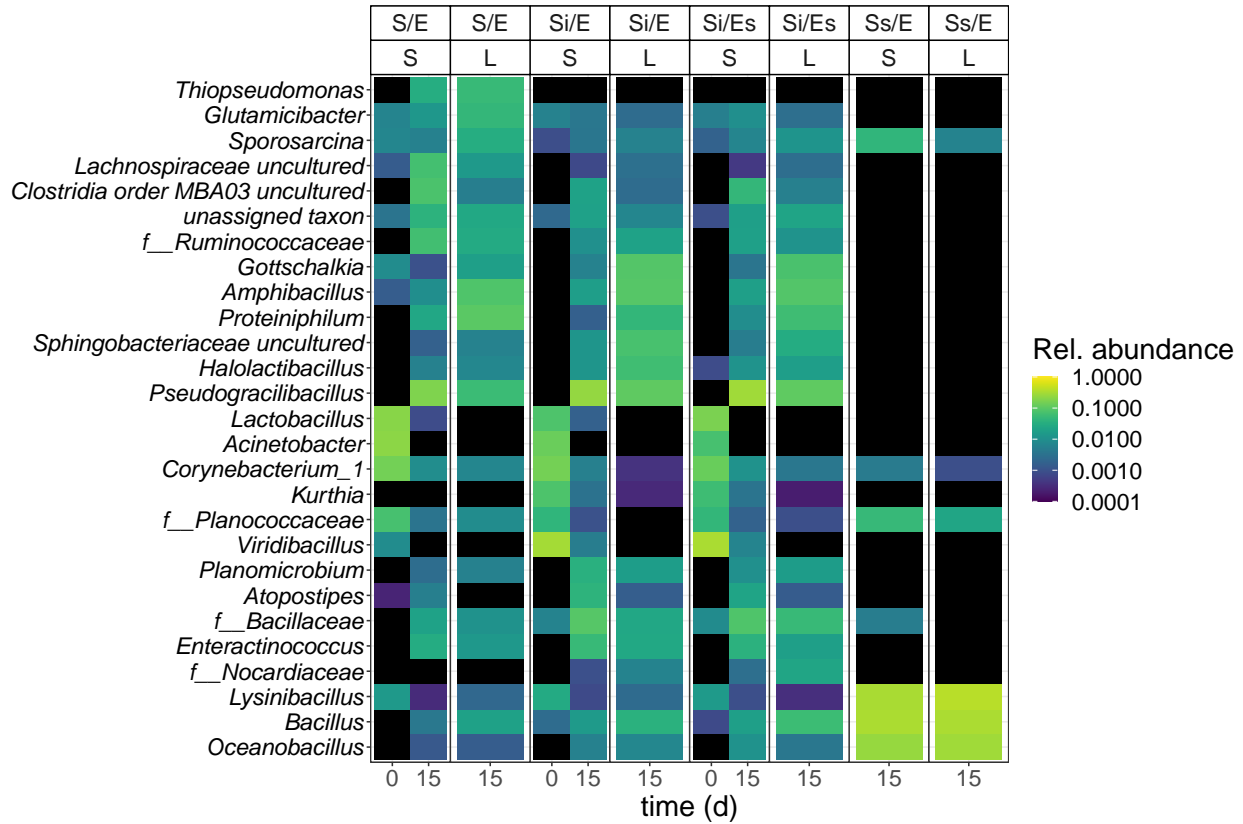


```
ggsave(plot = p.hm.CF, "./figures/Fig_6A_heatmap_CF.png", h = 6, w = 10)
ggsave(plot = p.hm.CF, "./figures/Fig_6A_heatmap_CF.pdf", w = 320, h = 240, u = "mm")
```

3.3. Chicken manure

Figure 6B in manuscript.

```
p.hm.CM <- ggplot(hm.CM.sum, aes(x = Timepoint, y = best_hit2)) +
  geom_tile(aes(fill = mean)) +
  scale_fill_viridis("Rel. abundance", option = "D", na.value = "black",
    trans = "log10", limits = c(0.0001, 1),
    labels = function(n){format(n, scientific = F)}) +
  facet_grid(~ Treatment + Type, scales = "free", labeller = labs_hm) +
  labs(x = "time (d)", y = NULL) + theme_hm
p.hm.CM
```

```
ggsave(plot = p.hm.CM, "./figures/Fig_6B_heatmap_CM.png", h = 8, w = 12)
ggsave(plot = p.hm.CM, "./figures/Fig_6B_heatmap_CM.pdf", w = 320, h = 280, u = "mm")
```

3.4. Both diets and all genera

Export table of the complete genus composition per treatment.

```
# create a dataframe with treatment by taxa and mean rel abundance values
ps.hm.all.sum.s <- ps.hm.all.sum[, -6] # remove column "median"

# make one column that has all treatments as levels
ps.hm.all.sum.s$treatment <- interaction(ps.hm.all.sum.s$Type,
                                         ps.hm.all.sum.s$Timepoint,
                                         ps.hm.all.sum.s$Treatment,
                                         ps.hm.all.sum.s$Diet, drop = T)

# create pivot table: cast()
ps.hm.all.sum.c <- reshape::cast(ps.hm.all.sum.s,
                                  formula = OTU ~ treatment, value = "mean")

# export dataframe to txt-file
write.table(ps.hm.all.sum.c, "./tables/Schreven_Ch4_RelAbd_Genus_all_mean.txt", sep="\t")
```