Mawi vs. Frozen Stool Samples Final

Rigel Sison

1/18/2022

```
# Load packages
necessary_packages <-  c("directlabels", "knitr", "clustsig", "ellipse", "phyloseq", "ggplot2", "plyr",
packages <- lapply(necessary_packages, library, character.only = TRUE)
#("biomformat)
library(dplyr)
library(decontam)
library("RColorBrewer")
library("magrittr")
library("scales")
library("randomForest")
library(ggtern)
library(tidyr)
library(viridis)
library(cowplot)
library(tidyverse) # Easily Install and Load the 'Tidyverse'.
library(coin) # Conditional Inference Procedures in a Permutation Test Framework.
library(reshape2) # Flexibly Reshape Data: A Reboot of the Reshape Package.
library(ggnewscale)
library(VennDiagram)
library(gplots)
library(tibble)
source("a.diversity.anova.R")
```

Loading Data

```
table <- read.table(file = 'frozen_v_mawi.tsv', sep = '\t', header = TRUE)
copy <- table
rownames(copy) <- c(1:733)
rownames(copy)[1:9] <- paste0("00", rownames(copy)[1:9])
rownames(copy)[10:99] <- paste0("0", rownames(copy)[10:99])
rownames(copy) <- paste0("ST_", rownames(copy))
colnames(copy) <- gsub('MP_','', colnames(copy))
```

```
final <- copy
st.phylo <- final[, 2:8]
st.counts <- final[, 9:74]
st.counts <- trunc(st.counts)
st.counts.t <- t(st.counts)

map <- read.table("map_frozen_mawi.txt", header=TRUE, fill = TRUE)
map$Date <- as.POSIXct(map$Date,
                       format = "%Y-%m-%d")
rownames(map) <- map[,1]
map.samp <- map[,2:ncol(map)]
map.samp <- map.samp[rownames(st.counts.t),]

identical(rownames(st.counts.t),rownames(map))
```

```
## [1] TRUE
```

```
st.phylo <- as.matrix(st.phylo)

class(st.counts.t) <- "numeric"

OTU <- otu_table(st.counts, taxa_are_rows = TRUE)
TAX <- tax_table(st.phylo)
META <- sample_data(map)
st.physeq <- phyloseq(OTU, TAX, META)
```

```
sum(taxa_sums(st.physeq) == 0)
```

```
## [1] 0
```

```
sum(taxa_sums(st.physeq) == 1) #7
```

```
## [1] 7
```

```
sum(taxa_sums(st.physeq) == 2) #13
```

```
## [1] 13
```

```
sum(taxa_sums(st.physeq) == 3) #11
```

```
## [1] 11
```

```
st.physeq.f <- filter_taxa(st.physeq, function(x) sum(x > 3) > (0.02*length(x)), TRUE)
```

Subset the phyloseq data by RNA and DNA

```
st.physeq.dna <- subset_samples(st.physeq.f, Nucleic.Acid == "DNA")
st.physeq.rna <- subset_samples(st.physeq.f, Nucleic.Acid == "RNA")

day0 <- subset_samples(st.physeq.dna, Day = "0")
```

Filter OTUs not present in both day 0 DNA sample across Mawi and Frozen

```
library(MicrobiotaProcess)
vennlist <- get_vennlist(obj=day0, factorNames="Type")
ItemsList<- venn(vennlist, show.plot = FALSE)
intersections<- attributes(ItemsList)$intersections


detach("package:MicrobiotaProcess", unload = TRUE)

#filter OTUs not present in both day 0 samples across Mawi and Frozen
row.names.remove <- c(intersections$Mawi, intersections$Frozen)
copy.f <- copy[!(rownames(copy) %in% row.names.remove), ]
```

```
final <- copy.f
st.phylo <- final[, 2:8]
st.counts <- final[, 9:74]
st.counts <- trunc(st.counts)

st.counts.t <- t(st.counts)

map <- read.table("map_frozen_mawi.txt", header=TRUE, fill = TRUE)
rownames(map) <- map[,1]
map.samp <- map[,2:ncol(map)]
map.samp <- map.samp[rownames(st.counts.t),]

identical(rownames(st.counts.t),rownames(map))
```

```
## [1] TRUE
```

```
st.phylo <- as.matrix(st.phylo)

class(st.counts.t) <- "numeric"

OTU <- otu_table(st.counts, taxa_are_rows = TRUE)
TAX <- tax_table(st.phylo)
META <- sample_data(map)
st.physeq <- phyloseq(OTU, TAX, META)

sum(taxa_sums(st.physeq) == 0)
```

```
## [1] 0
```

```
sum(taxa_sums(st.physeq) == 1) #7
```

```
## [1] 7
```

```
sum(taxa_sums(st.physeq) == 2) #13
```

```
## [1] 13
```

```
sum(taxa_sums(st.physeq) == 3) #11
```

```
## [1] 11
```

```
st.physeq.f <- filter_taxa(st.physeq, function(x) sum(x > 3) > (0.02*length(x)), TRUE)


#Subset the phyloseq data by RNA and DNA


st.physeq.dna <- subset_samples(st.physeq.f, Nucleic.Acid == "DNA")
st.physeq.rna <- subset_samples(st.physeq.f, Nucleic.Acid == "RNA")
```

```
set.seed(13)
#rarefy
st.physeq.fr <- rarefy_even_depth(st.physeq.f, sample.size = 280000, rngseed = TRUE)
st.physeq.dna.r <- rarefy_even_depth(st.physeq.dna, sample.size = 280000, rngseed = TRUE)

st.physeq.dna.r.mawi <- subset_samples(st.physeq.dna.r, Type == "Mawi")
st.physeq.dna.r.frozen <- subset_samples(st.physeq.dna.r, Type == "Frozen")
```

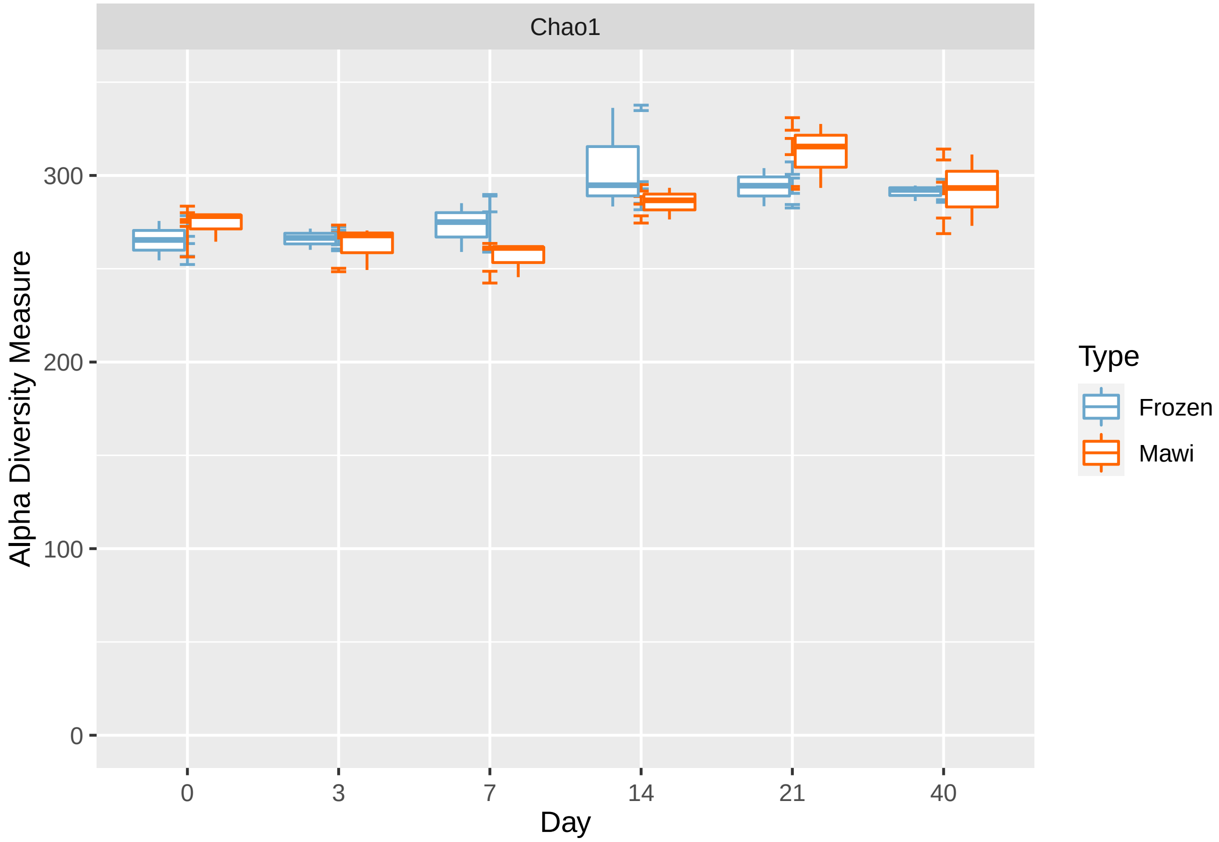
Alpha diversity metrics (Chao1, Inverse Simpson, Shannon)

```
sample_data(st.physeq.dna.r)$Day <- factor(sample_data(st.physeq.dna.r)$Day, levels = c("0", "3", "7",
sample_data(st.physeq.dna.r.mawi)$Day <- factor(sample_data(st.physeq.dna.r.mawi)$Day, levels = c("0",
sample_data(st.physeq.dna.r.frozen)$Day <- factor(sample_data(st.physeq.dna.r.frozen)$Day, levels = c("
```

```
richness.dna <- estimate_richness(st.physeq.dna.r, split = TRUE, measures = c("Chao1", "InvSimpson", "S

richness.dna.chao <- estimate_richness(st.physeq.dna.r, split = TRUE, measures = "Chao1")
richness.dna.chao$SampleID <- row.names(richness.dna.chao)
richness.dna.invs <- estimate_richness(st.physeq.dna.r, split = TRUE, measures = "InvSimpson")
richness.dna.invs$SampleID <- row.names(richness.dna.invs)
richness.dna.shannon <- estimate_richness(st.physeq.dna.r, split = TRUE, measures = "Shannon")
richness.dna.shannon$SampleID <- row.names(richness.dna.shannon)

write.csv(richness.dna, "richness_data.csv", row.names = TRUE)
```

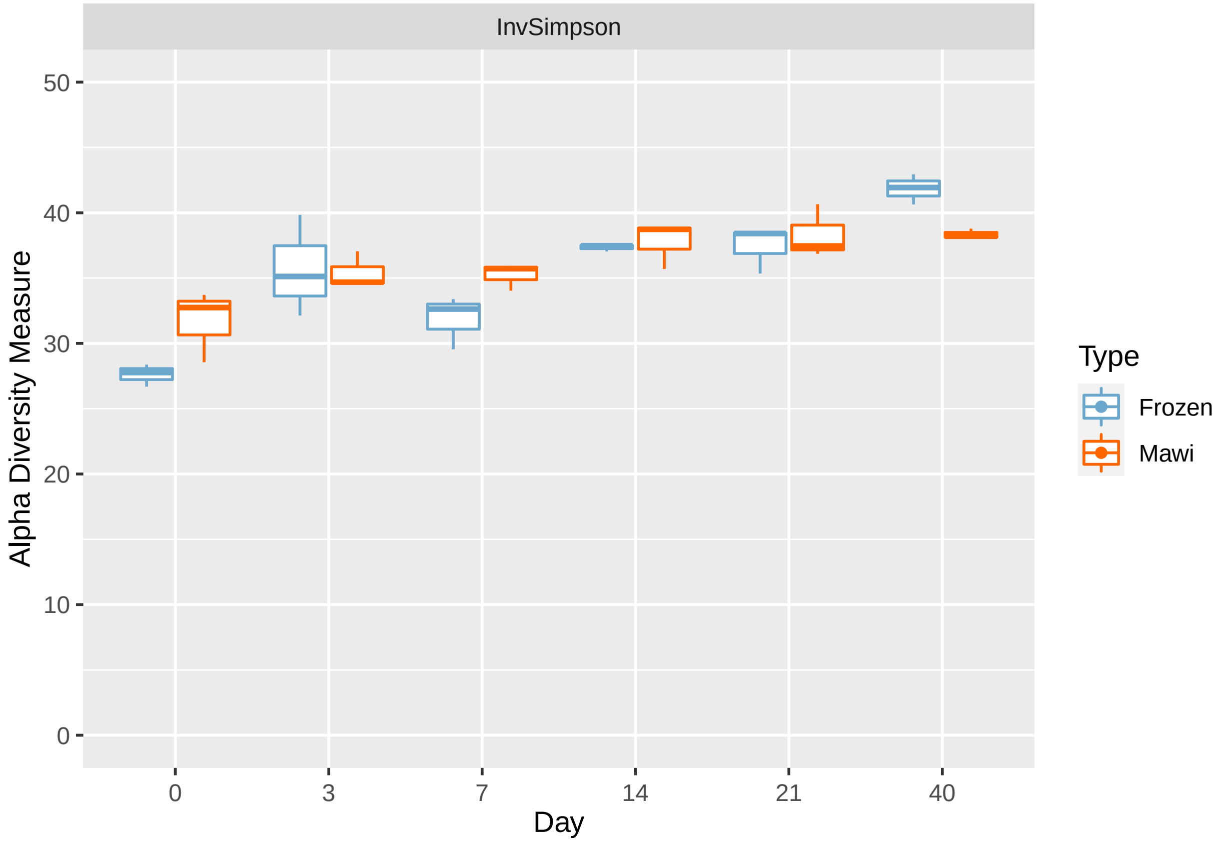
```
richness.dna.chao.plot <- plot_richness(st.physeq.dna.r,
                               measures = "Chao1",
                               x = "Day",
                               color = "Type",
                               shape = NA
                               ) +
  geom_boxplot() +
  coord_cartesian(ylim = c(0, 350)) +
  scale_color_manual(values = c('Frozen' = '#6BA7CC', 'Mawi' = '#ff6600')) +
  theme(axis.text.x = element_text(angle = 0, hjust = 0.5))
richness.dna.chao$layers <- richness.dna.chao$layers[-1]
richness.dna.chao$layers <- richness.dna.chao$layers[-1]

richness.dna.invs.plot <- plot_richness(st.physeq.dna.r,
                              measures = "invsimpson",
                              x = "Day",
                              color = "Type",
                              shape = NA
                              ) +
  geom_boxplot() +
  geom_point(position = position_dodge(width = 0.75)) +
  coord_cartesian(ylim = c(0, 50)) +
  scale_color_manual(values = c('Frozen' = '#6BA7CC', 'Mawi' = '#ff6600')) +
  theme(axis.text.x = element_text(angle = 0, hjust = 0.5))

richness.dna.shannon.plot <- plot_richness(st.physeq.dna.r,
                              measures = "Shannon",
                              x = "Day",
                              color = "Type",
                              shape = NA
                              ) +
  geom_boxplot() +
  scale_color_manual(values = c('Frozen' = '#6BA7CC', 'Mawi' = '#ff6600')) +
  theme(axis.text.x = element_text(angle = 0, hjust = 0.5))

fit.shannon <- aov(Shannon ~ SampleID, data = richness.dna.shannon)
fit.chao <- aov(Chao1 ~ SampleID, data = richness.dna.chao)
fit.invs <- aov(InvSimpson ~ SampleID, data = richness.dna.invs)
```

```
richness.dna.chao.plot
```



```
ggsave("richness_all_samples_dna_chao.pdf",
       width = 6,
       height = 4,
       units = "in")

richness.dna.invs.plot
```

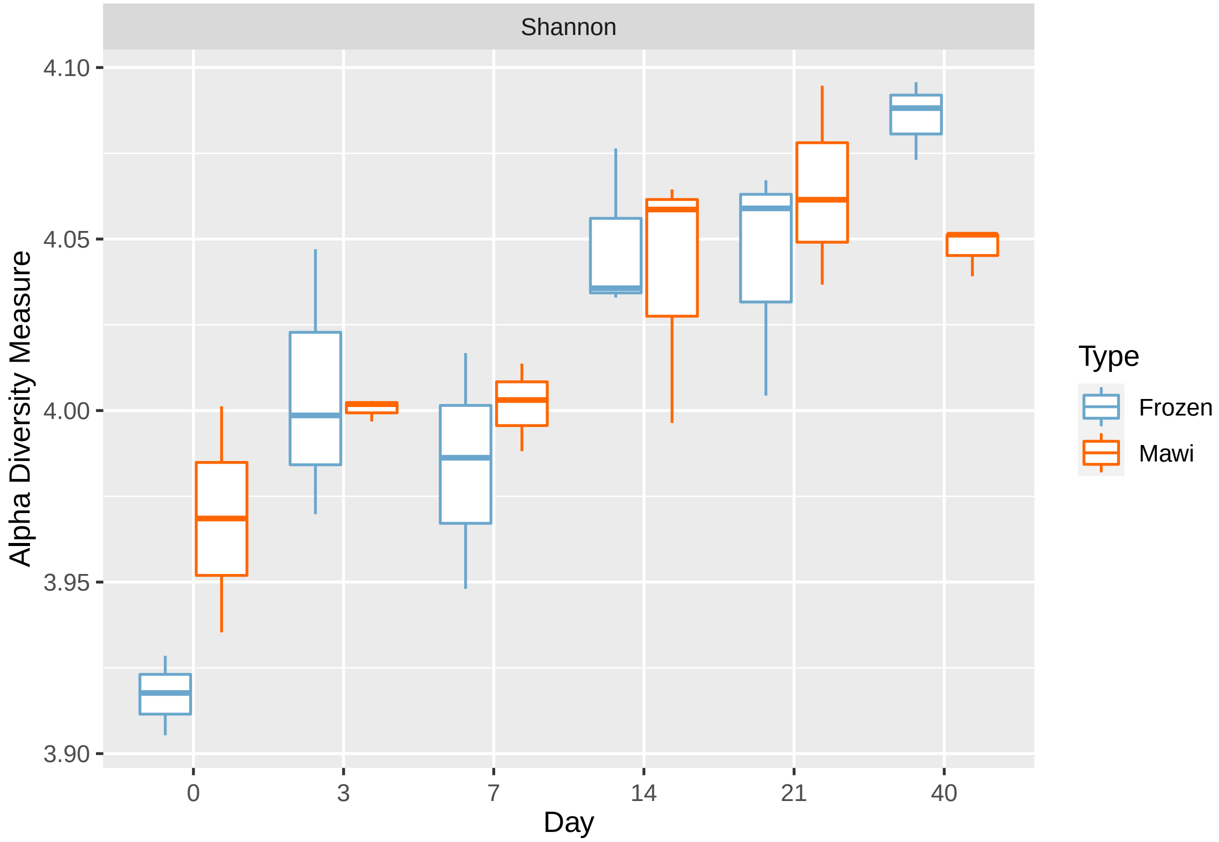
```
## Warning: Removed 36 rows containing missing values (geom_point).
```



```
ggsave("richness_all_samples_dna_invsimpson.pdf",
       width = 6,
       height = 4,
       units = "in")
```

```
## Warning: Removed 36 rows containing missing values (geom_point).
```

```
richness.dna.shannon.plot
```



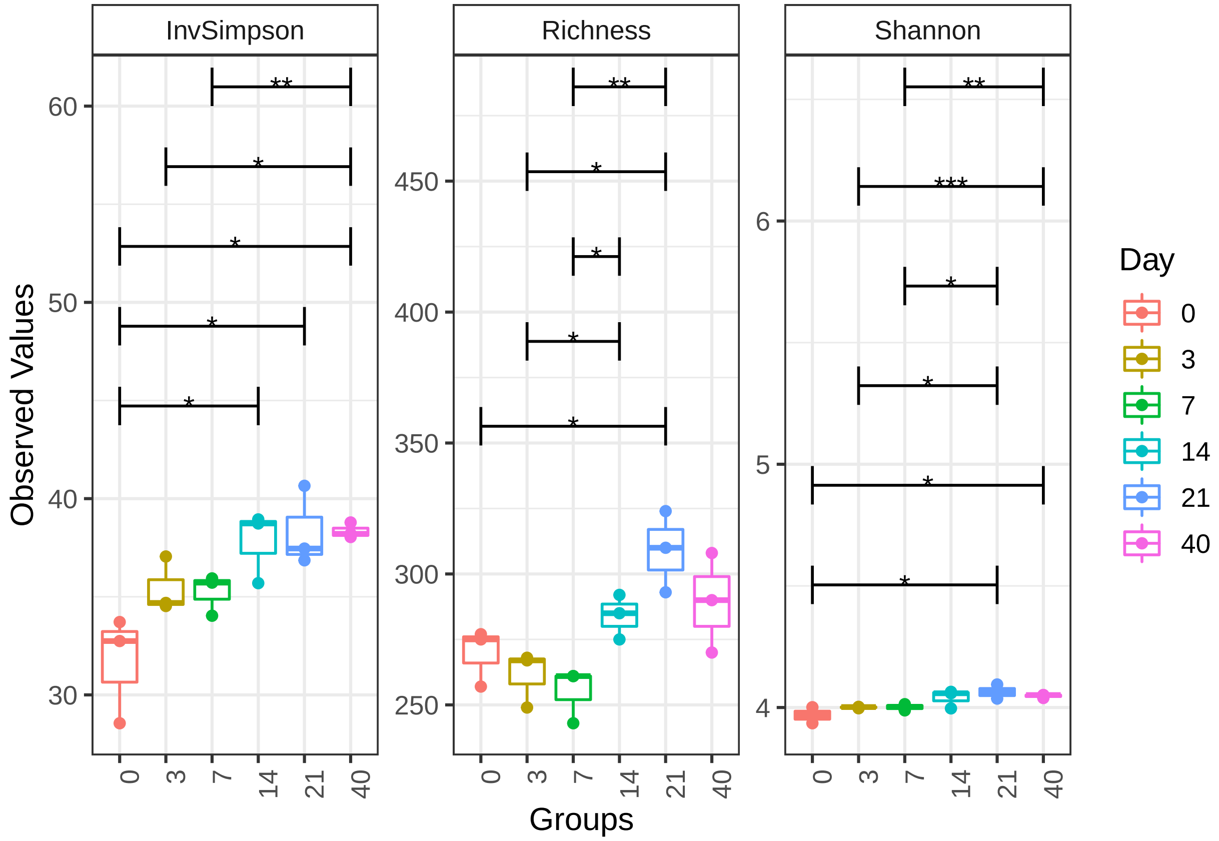
```
ggsave("richness_all_samples_dna_shannon.pdf",
       width = 6,
       height = 4,
       units = "in")
```

ANOVA

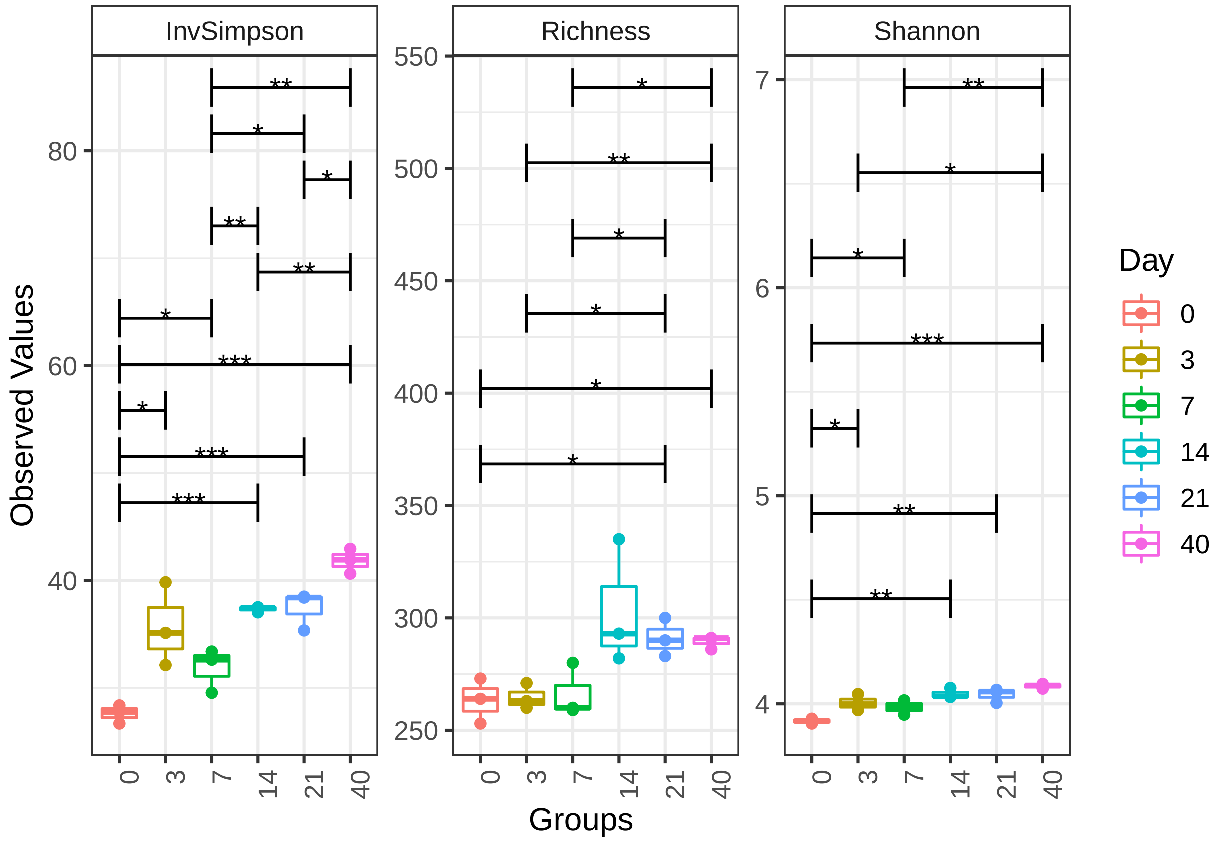
```
anova.mawi.plot <- plot_anova_diversity(st.physeq.dna.r.mawi, method = c("invsimpson", "shannon", "richn
anova.frozen.plot <- plot_anova_diversity(st.physeq.dna.r.frozen, c("invsimpson", "shannon", "richness")

st.physeq.dna.r.mawi.t <- t(st.physeq.dna.r.mawi)
st.physeq.dna.r.frozen.t <- t(st.physeq.dna.r.frozen)

a_div <- alpha_div(physeq = st.physeq.dna.r.mawi.t, method = c("invsimpson", "shannon", "richness"))
meta_table <- sample_data(st.physeq.dna.r.mawi.t)

df <-data.frame(a_div,(meta_table[, "Day"])[as.character(a_div$sample),])
mawi.anova <- perform_anova(df,meta_table, "Day", 0.1)

a_div <- alpha_div(physeq = st.physeq.dna.r.frozen.t, method = c("invsimpson", "shannon", "richness"))
meta_table <- sample_data(st.physeq.dna.r.frozen.t)

df<-data.frame(a_div,(meta_table[, "Day"])[as.character(a_div$sample),])
frozen.anova <- perform_anova(df,meta_table, "Day", 0.1)
```

```
write.csv(mawi.anova$df_pw, "mawi_anova.csv", row.names = TRUE)
write.csv(frozen.anova$df_pw, "frozen_anova.csv", row.names = TRUE)
```

```
anova.mawi.plot
```



```
ggsave("anova_mawi.pdf",
       width = 16,
       height = 9,
       units = "in")

anova.frozen.plot
```

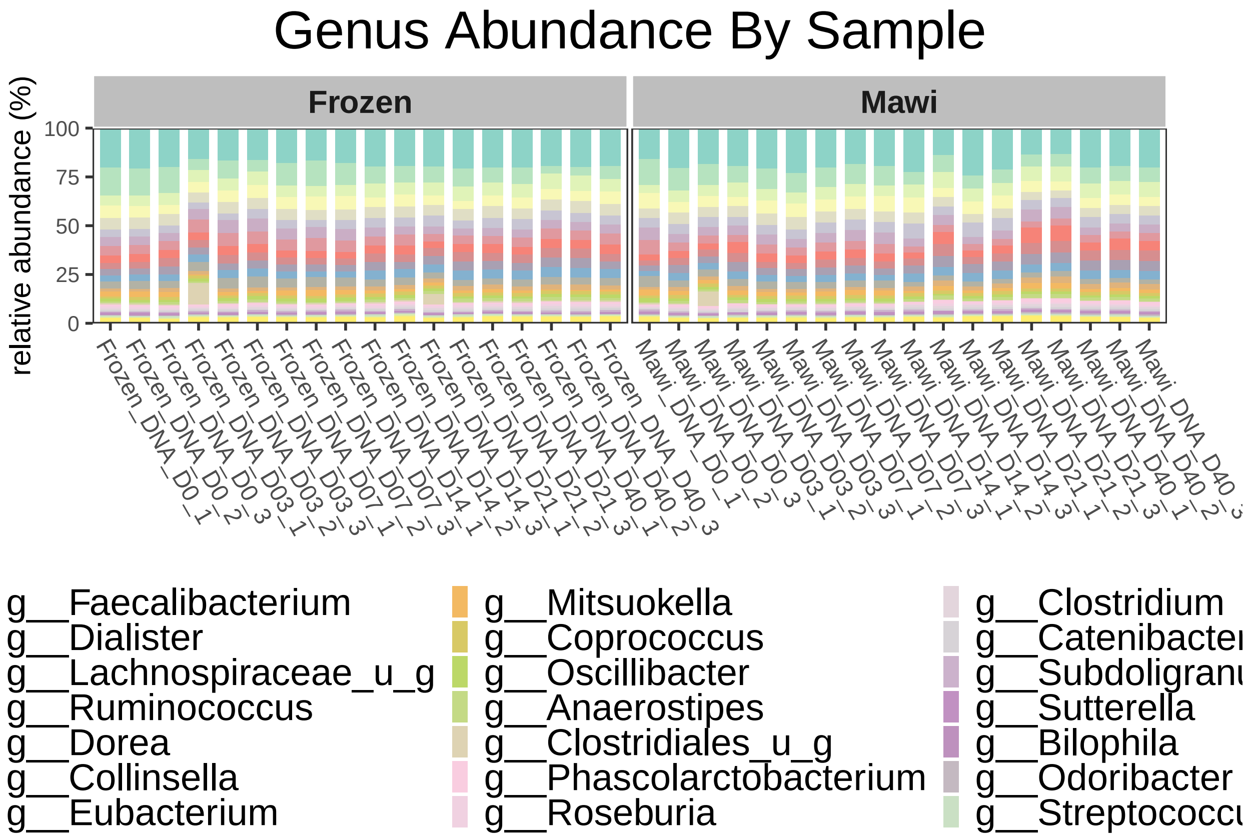


```
ggsave("anova_frozen.pdf",
       width = 16,
       height = 9,
       units = "in")
```

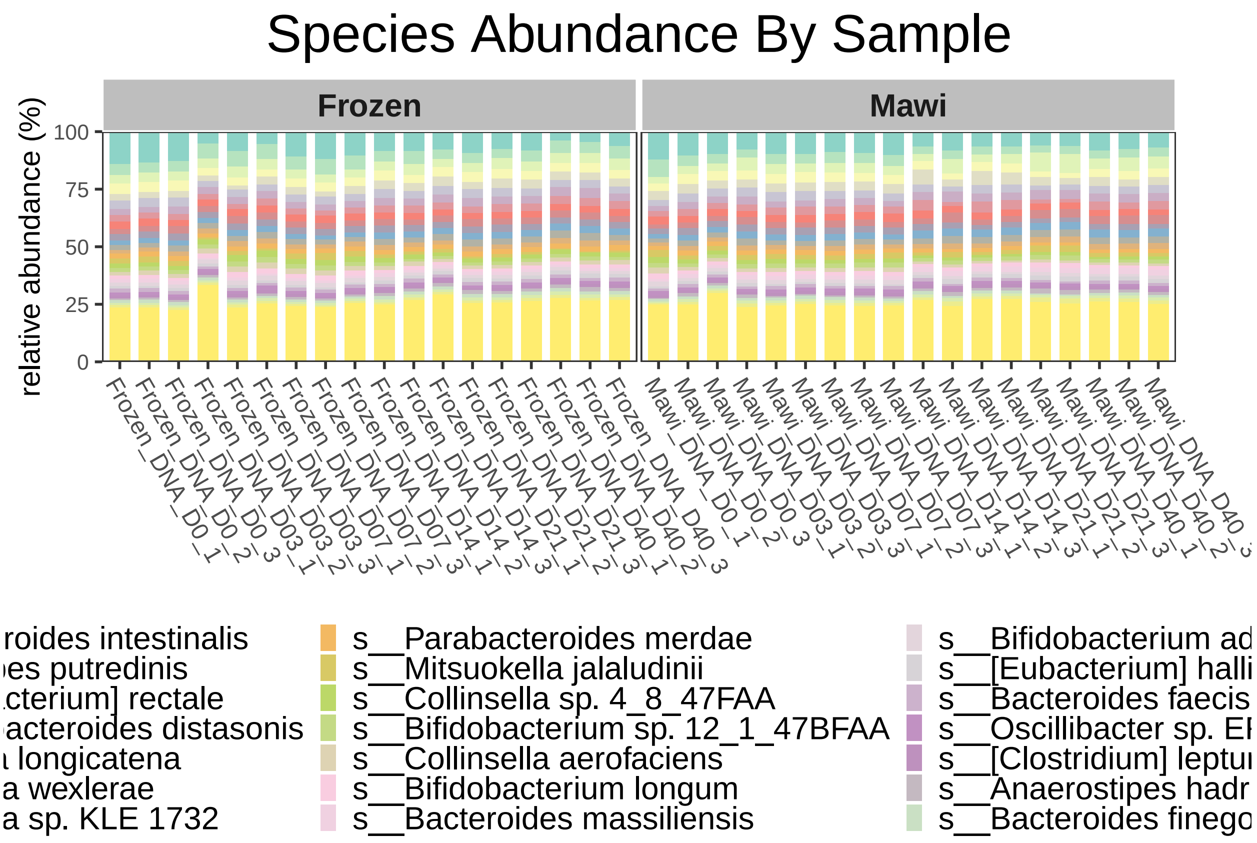
Community Composition

```
library(MicrobiotaProcess)

genustaxa <- get_taxadf(obj=st.physeq.dna.r, taxlevel=6)
pgenus <- ggbartax(obj=genustaxa, facetNames="Type", topn = 30) +
  xlab(NULL) +
  ylab("relative abundance (%)") +
  scale_fill_manual(values = c(colorRampPalette(RColorBrewer::brewer.pal(12, "Set3"))(31))) +
  guides(fill = guide_legend(keywidth = 0.5, keyheight = 0.5)) +
  ggtitle("Genus Abundance By Sample") +
  theme(plot.title = element_text(hjust = 0.5, size = 20)) +
  theme(legend.text = element_text(size = 14)) + #5
  theme(axis.text.x = element_text(angle = 300, vjust = 1, size = 9)) #7

speciestaxa <- get_taxadf(obj=st.physeq.dna.r, taxlevel=7)
pspecies <- ggbartax(obj=speciestaxa, facetNames="Type", topn = 30) +
  xlab(NULL) +
  ylab("relative abundance (%)") +
  scale_fill_manual(values = c(colorRampPalette(RColorBrewer::brewer.pal(12, "Set3"))(31))) +
  guides(fill = guide_legend(keywidth = 0.5, keyheight = 0.5)) +
  ggtitle("Species Abundance By Sample") +
  theme(plot.title = element_text(hjust = 0.5, size = 20)) +
  theme(legend.text = element_text(size = 12)) + #4
  theme(axis.text.x = element_text(angle = 300, vjust = 1, size = 9)) #7
```

```
pgenus
```



```
ggsave("ampvis_barchart_genus_top30.pdf",
       width = 16,
       height = 9,
       units = "in")


pspecies
```



```r
ggsave("ampvis_barchart_species_top30.pdf",
       width = 16,
       height = 9,
       units = "in")
```

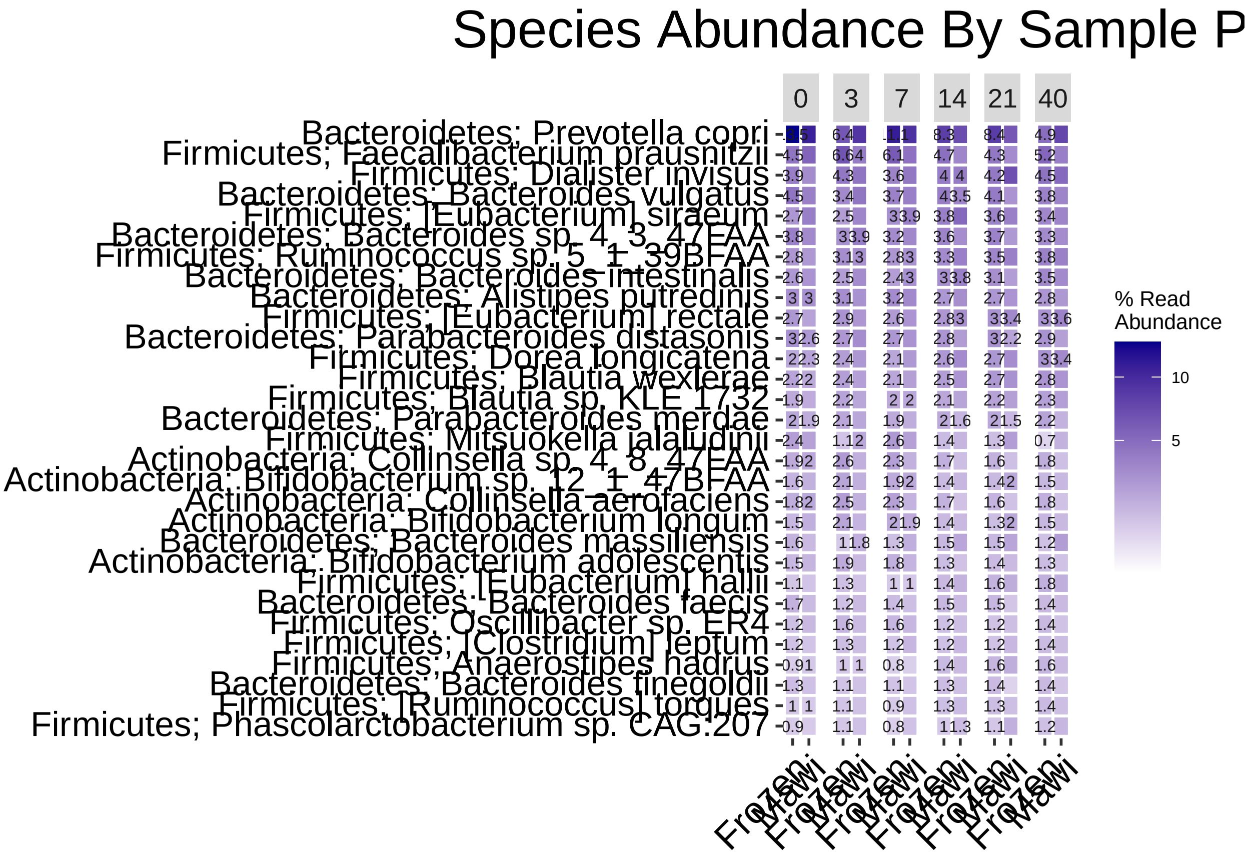
```r
detach("package:MicrobiotaProcess", unload = TRUE)
```

```r
obj <- st.physeq.dna.r
otutable <- data.frame(OTU = rownames(phyloseq::otu_table(obj)@.Data),
                       phyloseq::otu_table(obj)@.Data,
                       phyloseq::tax_table(obj)@.Data,
                       check.names = FALSE)

metadata <- data.frame(phyloseq::sample_data(obj),
                       check.names = FALSE)

my_ampvis2_object <- amp_load(otutable, metadata)

mawi_ampvis <- amp_subset_samples(my_ampvis2_object,
                                  Type %in% c("Mawi"),
                                  normalise = FALSE)
```

```
## 18 samples and 3 OTUs have been filtered
## Before: 36 samples and 405 OTUs
## After: 18 samples and 402 OTUs
```

```
ampvis_day <- amp_heatmap(my_ampvis2_object,
                          group_by = "Type",
                          facet_by = "Day",
                          tax_aggregate = "Species",
                          tax_add = c("Phylum"),
                          tax_show = 30,
                          color_vector = c("white", "darkblue"),
                          plot_colorscale = "sqrt",
                          plot_values = TRUE,
                          plot_values_size = 2) +
  theme(axis.text.x = element_text(angle = 45, size = 14, vjust = 1),
        axis.text.y = element_text(size = 13),
        legend.position = "right") +
  ggtitle("Species Abundance By Sample Per Day") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))
```

```
ampvis_day
```



```
ggsave("ampvis_heatmap_by_day.pdf",
       width = 16,
       height = 9,
       units = "in")
```

```
relative_abundance <- transform_sample_counts(st.physeq.dna.r, function(x) 100 * x/sum(x))
relative_abundance.phylum <- tax_glom(relative_abundance, "Phylum")
relative_abundance.family <- tax_glom(relative_abundance, "Family")
relative_abundance.genus <- tax_glom(relative_abundance, "Genus")

write.csv(tax_table(relative_abundance.phylum), "relative_abundance_phylum_tax.csv", row.names = TRUE)
write.csv(otu_table(relative_abundance.phylum), "relative_abundance_phylum_otu.csv", row.names = TRUE)

write.csv(tax_table(relative_abundance.family), "relative_abundance_family_tax.csv", row.names = TRUE)
write.csv(otu_table(relative_abundance.family), "relative_abundance_family_otu.csv", row.names = TRUE)

write.csv(tax_table(relative_abundance.genus), "relative_abundance_genus_tax.csv", row.names = TRUE)
write.csv(otu_table(relative_abundance.genus), "relative_abundance_genus_otu.csv", row.names = TRUE)
```

Anosim (analysis of similarities)

```
day0 <- subset_samples(st.physeq.dna.r, Day == 0)
day3 <- subset_samples(st.physeq.dna.r, Day == 3)
day7 <- subset_samples(st.physeq.dna.r, Day == 7)
day14 <- subset_samples(st.physeq.dna.r, Day == 14)
day21 <- subset_samples(st.physeq.dna.r, Day == 21)
day40 <- subset_samples(st.physeq.dna.r, Day == 40)

type_group0 <- get_variable(day0, "Type")
anosim.day0 <- anosim(phyloseq::distance(day0, "bray"), type_group0)
```

```
## ’nperm’ >= set of all permutations: complete enumeration.
```

```
## Set of permutations < ’minperm’. Generating entire set.
```

```
anosim.day0$signif #0.1
```

```
## [1] 0.1
```

```
anosim.day0$statistic #0.3703704
```

```
## [1] 0.3703704
```

```
type_group3 <- get_variable(day3, "Type")
anosim.day3 <- anosim(phyloseq::distance(day3, "bray"), type_group3)
```

```
## ’nperm’ >= set of all permutations: complete enumeration.
## Set of permutations < ’minperm’. Generating entire set.
```

```
anosim.day3$signif #0.1
```

```
## [1] 0.1
```

```
anosim.day3$statistic #0.8518519
```

```
## [1] 0.8518519
```

```
type_group7 <- get_variable(day7, "Type")
anosim.day7 <- anosim(phyloseq::distance(day7, "bray"), type_group7)
```

```
## ’nperm’ >= set of all permutations: complete enumeration.
## Set of permutations < ’minperm’. Generating entire set.
```

```
anosim.day7$signif #0.1
```

```
## [1] 0.1
```

```
anosim.day7$statistic #0.962963
```

```
## [1] 0.962963
```

```
type_group14 <- get_variable(day14, "Type")
anosim.day14 <- anosim(phyloseq::distance(day14, "bray"), type_group14)
```

```
## ’nperm’ >= set of all permutations: complete enumeration.
## Set of permutations < ’minperm’. Generating entire set.
```

```
anosim.day14$signif #0.1
```

```
## [1] 0.1
```

```
anosim.day14$statistic #0.3333333
```

```
## [1] 0.3333333
```

```
type_group21 <- get_variable(day21, "Type")
anosim.day21 <- anosim(phyloseq::distance(day21, "bray"), type_group21)
```

```
## ’nperm’ >= set of all permutations: complete enumeration.
## Set of permutations < ’minperm’. Generating entire set.
```

```
anosim.day21$signif #0.1
```

```
## [1] 0.1
```

```
anosim.day21$statistic #0.7777778
```

```
## [1] 0.7777778
```

```r
type_group40 <- get_variable(day40, "Type")
anosim.day40 <- anosim(phyloseq::distance(day40, "bray"), type_group40)
```

```
## 'nperm' >= set of all permutations: complete enumeration.
## Set of permutations < 'minperm'. Generating entire set.
```

```r
anosim.day40$signif #0.1
```

```
## [1] 0.1
```

```r
anosim.day40$statistic #1
```

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

```r
days <- c(0, 3, 7, 14, 21, 40)
pairs <- as.data.frame(combn(days, 2))

iter <- 1
all_stats <- list()

for (pair in pairs){

  val1 <- pairs[,iter][1]
  val2 <- pairs[,iter][2]

  subset <- subset_samples(st.physeq.dna.r.mawi, Day == val1 | Day == val2)
  var <- get_variable(subset, "Day")
  anosim <- anosim(phyloseq::distance(subset, "bray"), var)

  dat <- as.data.frame(anosim$statistic)
  dat$pair <- as.character(pairs[iter])

  all_stats[[iter]] <- dat

  iter <- iter + 1
}

mawi_stats <- do.call(rbind, all_stats)

iter <- 1
all_stats <- list()

for (pair in pairs){

  val1 <- pairs[,iter][1]
  val2 <- pairs[,iter][2]

  subset <- subset_samples(st.physeq.dna.r.frozen, Day == val1 | Day == val2)
  var <- get_variable(subset, "Day")
  anosim <- anosim(phyloseq::distance(subset, "bray"), var)
  dat <- as.data.frame(anosim$statistic)
  dat$pair <- as.character(pairs[iter])

  all_stats[[iter]] <- dat

  iter <- iter + 1
}

frozen_stats <- do.call(rbind, all_stats)
```

```
mawi.anosim <- read_csv("mawi_anosim.csv")
```

```
## Rows: 36 Columns: 3
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (3): Var1, Var2, value
```

```
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
mawi.anosim <- as.data.frame(mawi.anosim)

mawi.anosim$Var1 <- factor(mawi.anosim$Var1, levels = c("0", "3", "7", "14", "21", "40"))
mawi.anosim$Var2 <- factor(mawi.anosim$Var2, levels = c("0", "3", "7", "14", "21", "40"))

frozen.anosim <- read_csv("frozen_anosim.csv")
```

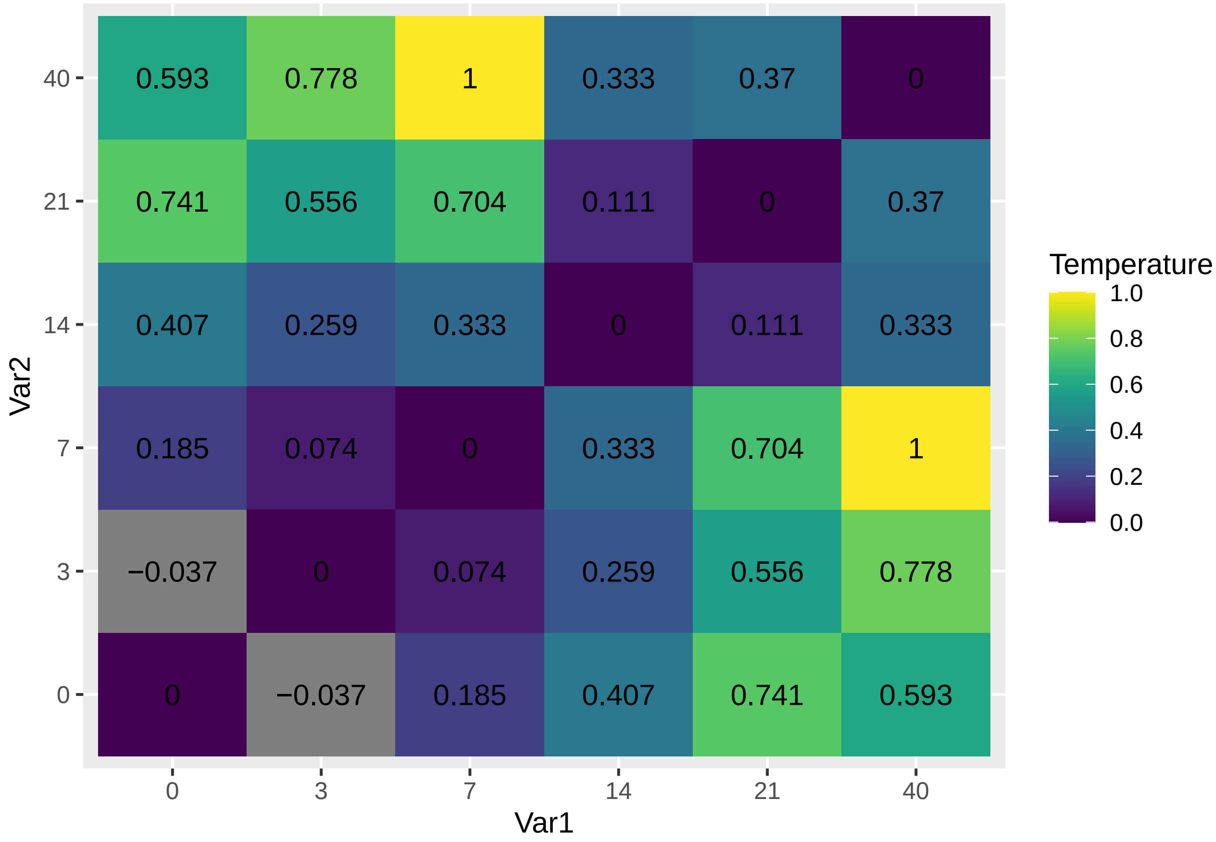
```
## Rows: 36 Columns: 3
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (3): Var1, Var2, value
```

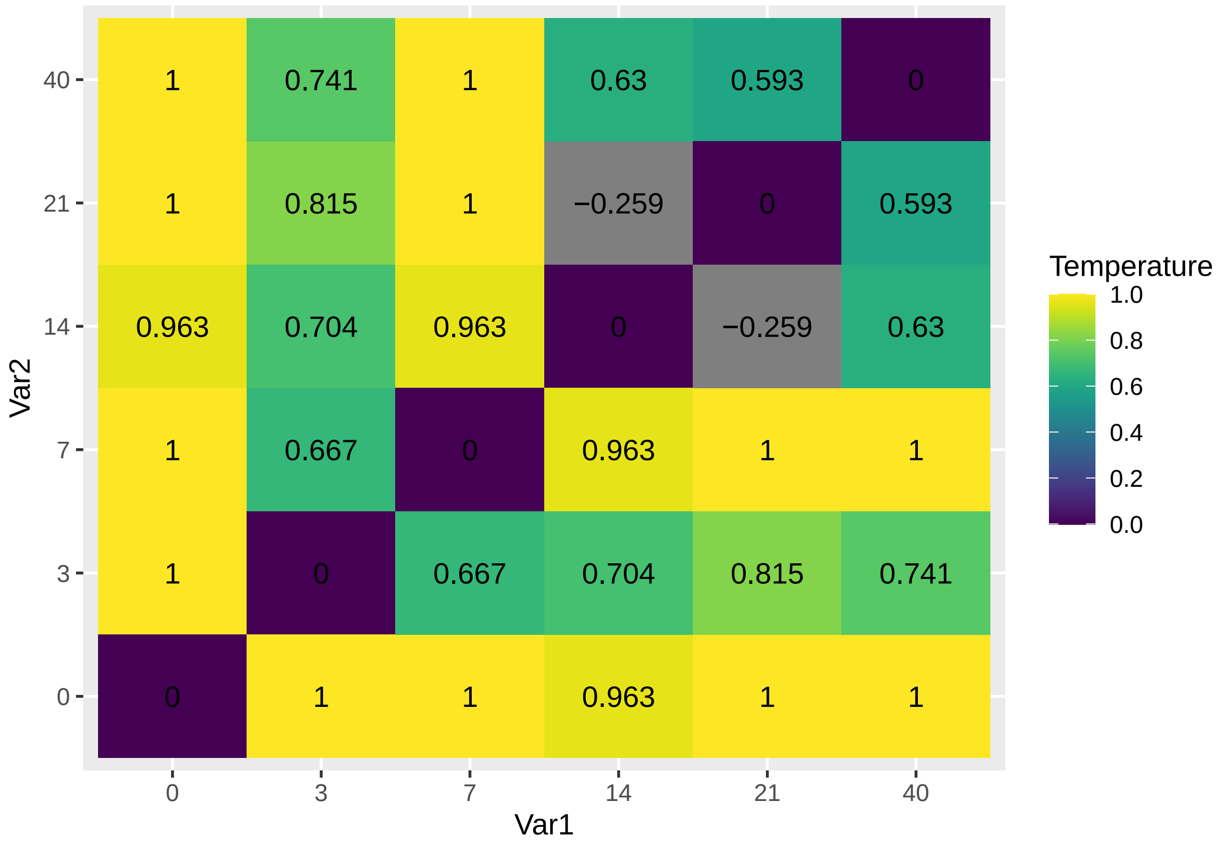
```
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
frozen.anosim <- as.data.frame(frozen.anosim)

frozen.anosim$Var1 <- factor(frozen.anosim$Var1, levels = c("0", "3", "7", "14", "21", "40"))
frozen.anosim$Var2 <- factor(frozen.anosim$Var2, levels = c("0", "3", "7", "14", "21", "40"))

rng <- range(mawi.anosim$value, frozen.anosim$value)

mawi.anosim.plot <- ggplot(mawi.anosim) +
  geom_tile(aes(x = Var1, y = Var2, fill = value)) +
  scale_fill_viridis(name="Temperature",
                     #midpoint = mean(rng),
                     breaks = seq(0, 1, 0.2),
                     limits = c(0, 1)) +
  geom_text(aes(x = Var1, y = Var2, label = round(value, 3)))

frozen.anosim.plot <- ggplot(frozen.anosim) +
  geom_tile(aes(x = Var1, y = Var2, fill = value)) +
  scale_fill_viridis(name="Temperature",
                     #midpoint = mean(rng),
                     breaks = seq(0, 1, 0.2),
                     limits = c(0, 1)) +  geom_text(aes(x = Var1, y = Var2, label = round(value, 3)))
```

```
mawi.anosim.plot
```



```
ggsave("mawi_anosim_plot.pdf",
       width = 6,
       height = 4,
       units = "in")

frozen.anosim.plot
```



```
ggsave("frozen_anosim_plot.pdf",
       width = 6,
       height = 4,
       units = "in")
```

Distance between T0 and other time points

Remove faulty replicates

```
filtered_reps <- subset_samples(st.physeq.dna, ID != "Frozen_DNA_D03_1" & ID != "Frozen_DNA_D03_3" & ID

st.species <- tax_glom(filtered_reps, "Species")
species30 <- names(sort(taxa_sums(st.species), TRUE)[1:30])
speciesbot <- names(sort(taxa_sums(st.species), TRUE)[31: length(taxa_sums(st.species))])
st.species30 <- prune_taxa(species30, st.species)
st.speciesbot <- prune_taxa(speciesbot, st.species)

#rarefy for even depth
set.seed(13)
st.species30.r = rarefy_even_depth(st.species30, sample.size = 280000, rngseed = TRUE)
st.speciesbot.r = rarefy_even_depth(st.speciesbot, sample.size = 90000, rngseed = TRUE)
st.physeq.dna.r.filtered = rarefy_even_depth(filtered_reps, sample.size = 400000, rngseed = TRUE)
```

Distances for only Top30 species

```r
# T0 vs T3, Mawi, species30

st.species30.t0_3.mawi <- subset_samples(st.species30.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi' &
dist.species30.t0_3.mawi <- phyloseq::distance(physeq = st.species30.t0_3.mawi, method = 'bray')
dist.species30.t0_3.mawi.df <- as.matrix(dist.species30.t0_3.mawi)
dist.species30.t0_3.mawi.list <- reshape2::melt(dist.species30.t0_3.mawi.df)[reshape2::melt(upper.tri(di
names(dist.species30.t0_3.mawi.list) <- c("c1", "c2", "distance")
dist.species30.t0_3.mawi.list <- cbind(dist.species30.t0_3.mawi.list, "comp" = "T0_vs_T03", "type" = "Ma

# T0 vs T7, Mawi, species30

st.species30.t0_7.mawi <- subset_samples(st.species30.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi' &
dist.species30.t0_7.mawi <- phyloseq::distance(physeq = st.species30.t0_7.mawi, method = 'bray')
dist.species30.t0_7.mawi.df <- as.matrix(dist.species30.t0_7.mawi)
dist.species30.t0_7.mawi.list <- reshape2::melt(dist.species30.t0_7.mawi.df)[reshape2::melt(upper.tri(di
names(dist.species30.t0_7.mawi.list) <- c("c1", "c2", "distance")
dist.species30.t0_7.mawi.list <- cbind(dist.species30.t0_7.mawi.list, "comp" = "T0_vs_T07", "type" = "Ma

# T0 vs T14, Mawi, species30

st.species30.t0_14.mawi <- subset_samples(st.species30.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi' &
dist.species30.t0_14.mawi <- phyloseq::distance(physeq = st.species30.t0_14.mawi, method = 'bray')
dist.species30.t0_14.mawi.df <- as.matrix(dist.species30.t0_14.mawi)
dist.species30.t0_14.mawi.list <- reshape2::melt(dist.species30.t0_14.mawi.df)[reshape2::melt(upper.tri
names(dist.species30.t0_14.mawi.list) <- c("c1", "c2", "distance")
dist.species30.t0_14.mawi.list <- cbind(dist.species30.t0_14.mawi.list, "comp" = "T0_vs_T14", "type" =

# T0 vs T21, Mawi, species30

st.species30.t0_21.mawi <- subset_samples(st.species30.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi' &
dist.species30.t0_21.mawi <- phyloseq::distance(physeq = st.species30.t0_21.mawi, method = 'bray')
dist.species30.t0_21.mawi.df <- as.matrix(dist.species30.t0_21.mawi)
dist.species30.t0_21.mawi.list <- reshape2::melt(dist.species30.t0_21.mawi.df)[reshape2::melt(upper.tri
names(dist.species30.t0_21.mawi.list) <- c("c1", "c2", "distance")
dist.species30.t0_21.mawi.list <- cbind(dist.species30.t0_21.mawi.list, "comp" = "T0_vs_T21", "type" =

# T0 vs T40, Mawi, species30

st.species30.t0_40.mawi <- subset_samples(st.species30.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi' &
dist.species30.t0_40.mawi <- phyloseq::distance(physeq = st.species30.t0_40.mawi, method = 'bray')
dist.species30.t0_40.mawi.df <- as.matrix(dist.species30.t0_40.mawi)
dist.species30.t0_40.mawi.list <- reshape2::melt(dist.species30.t0_40.mawi.df)[reshape2::melt(upper.tri
names(dist.species30.t0_40.mawi.list) <- c("c1", "c2", "distance")
dist.species30.t0_40.mawi.list <- cbind(dist.species30.t0_40.mawi.list, "comp" = "T0_vs_T40", "type" =

#combine all tables
dist.per.time_comp.species30.mawi <- rbind(dist.species30.t0_3.mawi.list, dist.species30.t0_7.mawi.list
                                           dist.species30.t0_40.mawi.list)

#plot
boxplot.species30.mawi <- ggplot(dist.per.time_comp.species30.mawi, aes(x = comp, y = distance, fill =
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") + theme_bw() + theme(axis.title.x=element_blan
  coord_cartesian(ylim = c(0, 0.27)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5))
```

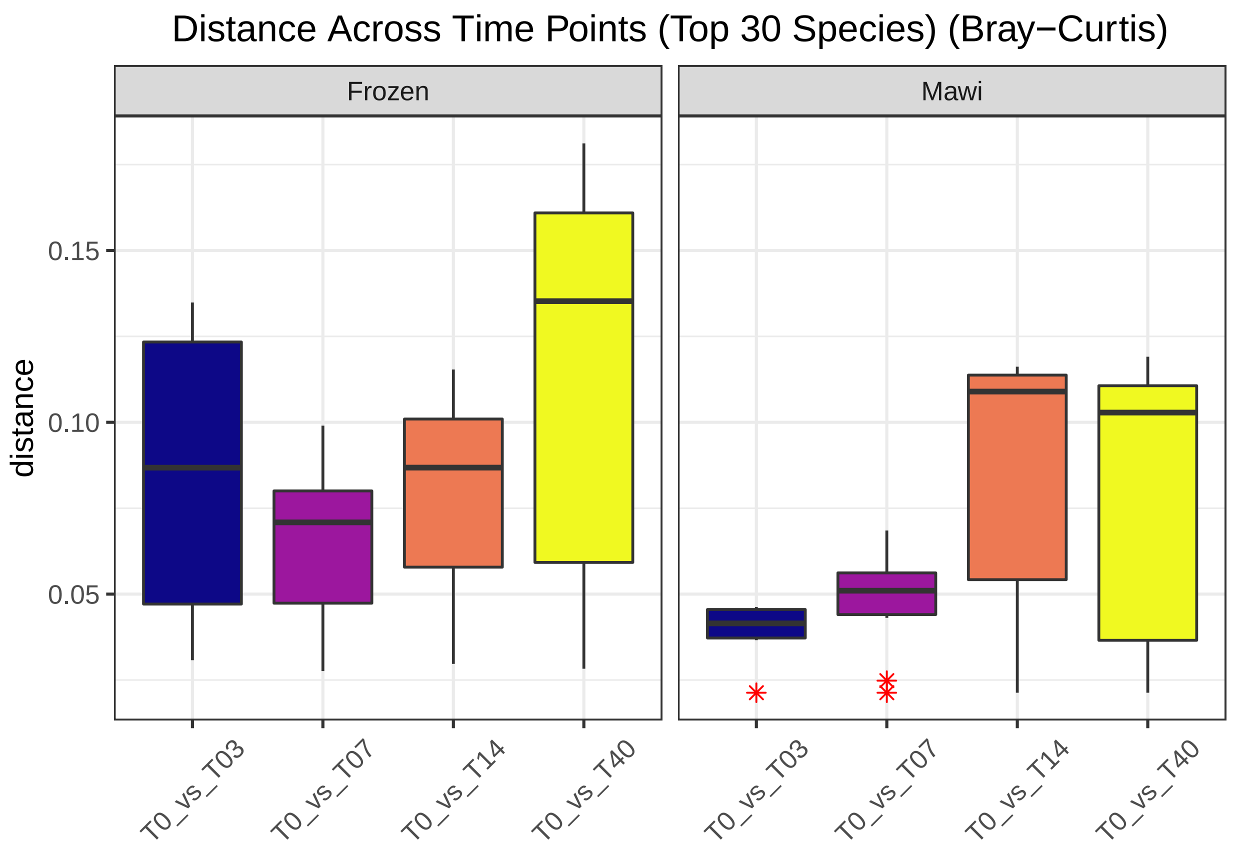
```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```

```
# T0 vs T3, Frozen, species30

st.species30.t0_3.frozen <- subset_samples(st.species30.r, Type == 'Frozen' & Day == '0' | Type == 'Fro
dist.species30.t0_3.frozen <- phyloseq::distance(physeq = st.species30.t0_3.frozen, method = 'bray')
dist.species30.t0_3.frozen.df <- as.matrix(dist.species30.t0_3.frozen)
dist.species30.t0_3.frozen.list <- reshape2::melt(dist.species30.t0_3.frozen.df)[reshape2::melt(upper.t
names(dist.species30.t0_3.frozen.list) <- c("c1", "c2", "distance")
dist.species30.t0_3.frozen.list <- cbind(dist.species30.t0_3.frozen.list, "comp" = "T0_vs_T03", "type" 

# T0 vs T7, Frozen, species30

st.species30.t0_7.frozen <- subset_samples(st.species30.r, Type == 'Frozen' & Day == '0' | Type == 'Fro
dist.species30.t0_7.frozen <- phyloseq::distance(physeq = st.species30.t0_7.frozen, method = 'bray')
dist.species30.t0_7.frozen.df <- as.matrix(dist.species30.t0_7.frozen)
dist.species30.t0_7.frozen.list <- reshape2::melt(dist.species30.t0_7.frozen.df)[reshape2::melt(upper.t
names(dist.species30.t0_7.frozen.list) <- c("c1", "c2", "distance")
dist.species30.t0_7.frozen.list <- cbind(dist.species30.t0_7.frozen.list, "comp" = "T0_vs_T07", "type" 

# T0 vs T14, Frozen, species30

st.species30.t0_14.frozen <- subset_samples(st.species30.r, Type == 'Frozen' & Day == '0' | Type == 'Fr
dist.species30.t0_14.frozen <- phyloseq::distance(physeq = st.species30.t0_14.frozen, method = 'bray')
dist.species30.t0_14.frozen.df <- as.matrix(dist.species30.t0_14.frozen)
dist.species30.t0_14.frozen.list <- reshape2::melt(dist.species30.t0_14.frozen.df)[reshape2::melt(upper
names(dist.species30.t0_14.frozen.list) <- c("c1", "c2", "distance")
dist.species30.t0_14.frozen.list <- cbind(dist.species30.t0_14.frozen.list, "comp" = "T0_vs_T14", "type

# T0 vs T21, Frozen, species30

st.species30.t0_21.frozen <- subset_samples(st.species30.r, Type == 'Frozen' & Day == '0' | Type == 'Fr
dist.species30.t0_21.frozen <- phyloseq::distance(physeq = st.species30.t0_21.frozen, method = 'bray')
dist.species30.t0_21.frozen.df <- as.matrix(dist.species30.t0_21.frozen)
dist.species30.t0_21.frozen.list <- reshape2::melt(dist.species30.t0_21.frozen.df)[reshape2::melt(upper
names(dist.species30.t0_21.frozen.list) <- c("c1", "c2", "distance")
dist.species30.t0_21.frozen.list <- cbind(dist.species30.t0_21.frozen.list, "comp" = "T0_vs_T21", "type

# T0 vs T40, Frozen, species30

st.species30.t0_40.frozen <- subset_samples(st.species30.r, Type == 'Frozen' & Day == '0' | Type == 'Fr
dist.species30.t0_40.frozen <- phyloseq::distance(physeq = st.species30.t0_40.frozen, method = 'bray')
dist.species30.t0_40.frozen.df <- as.matrix(dist.species30.t0_40.frozen)
dist.species30.t0_40.frozen.list <- reshape2::melt(dist.species30.t0_40.frozen.df)[reshape2::melt(upper
names(dist.species30.t0_40.frozen.list) <- c("c1", "c2", "distance")
dist.species30.t0_40.frozen.list <- cbind(dist.species30.t0_40.frozen.list, "comp" = "T0_vs_T40", "type
#combine all tables
dist.per.time_comp.species30.frozen <- rbind(dist.species30.t0_3.frozen.list, dist.species30.t0_7.frozen
                                             dist.species30.t0_40.frozen.list)

#plot
boxplot.species30.frozen <- ggplot(dist.per.time_comp.species30.frozen, aes(x = comp, y = distance, fil
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") + theme_bw() + theme(axis.title.x=element_blank
  coord_cartesian(ylim = c(0, 0.27)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5))
```

```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```

```
dist.per.time_comp.species30 <- rbind(dist.per.time_comp.species30.frozen, dist.per.time_comp.species30

boxplot.species30 <- ggplot(dist.per.time_comp.species30, aes(x = comp, y = distance, fill = comp, )) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") +
  facet_grid(. ~ type) +
  theme_bw() +
  theme(axis.title.x = element_blank(), legend.position = "none") +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5)) +
  ggtitle("Distance Across Time Points (Top 30 Species) (Bray-Curtis)") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
boxplot.species30
```



```
ggsave("boxplot_across_time_points (bray) (reps removed) (top 30).png",
       width = 16,
       height = 9,
       units = "in")
```

Distances for all species not in Top 30

```
# T0 vs T3, Mawi, speciesbot

st.speciesbot.t0_3.mawi <- subset_samples(st.speciesbot.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi'
dist.speciesbot.t0_3.mawi <- phyloseq::distance(physeq = st.speciesbot.t0_3.mawi, method = 'bray')
dist.speciesbot.t0_3.mawi.df <- as.matrix(dist.speciesbot.t0_3.mawi)
dist.speciesbot.t0_3.mawi.list <- reshape2::melt(dist.speciesbot.t0_3.mawi.df)[reshape2::melt(upper.tri
names(dist.speciesbot.t0_3.mawi.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_3.mawi.list <- cbind(dist.speciesbot.t0_3.mawi.list, "comp" = "T0_vs_T03", "type" =

# T0 vs T7, Mawi, speciesbot

st.speciesbot.t0_7.mawi <- subset_samples(st.speciesbot.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi'
dist.speciesbot.t0_7.mawi <- phyloseq::distance(physeq = st.speciesbot.t0_7.mawi, method = 'bray')
dist.speciesbot.t0_7.mawi.df <- as.matrix(dist.speciesbot.t0_7.mawi)
dist.speciesbot.t0_7.mawi.list <- reshape2::melt(dist.speciesbot.t0_7.mawi.df)[reshape2::melt(upper.tri
names(dist.speciesbot.t0_7.mawi.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_7.mawi.list <- cbind(dist.speciesbot.t0_7.mawi.list, "comp" = "T0_vs_T07", "type" =
# T0 vs T14, Mawi, speciesbot

st.speciesbot.t0_14.mawi <- subset_samples(st.speciesbot.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi
dist.speciesbot.t0_14.mawi <- phyloseq::distance(physeq = st.speciesbot.t0_14.mawi, method = 'bray')
dist.speciesbot.t0_14.mawi.df <- as.matrix(dist.speciesbot.t0_14.mawi)
dist.speciesbot.t0_14.mawi.list <- reshape2::melt(dist.speciesbot.t0_14.mawi.df)[reshape2::melt(upper.t
names(dist.speciesbot.t0_14.mawi.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_14.mawi.list <- cbind(dist.speciesbot.t0_14.mawi.list, "comp" = "T0_vs_T14", "type"

# T0 vs T21, Mawi, speciesbot

st.speciesbot.t0_21.mawi <- subset_samples(st.speciesbot.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi
dist.speciesbot.t0_21.mawi <- phyloseq::distance(physeq = st.speciesbot.t0_21.mawi, method = 'bray')
dist.speciesbot.t0_21.mawi.df <- as.matrix(dist.speciesbot.t0_21.mawi)
dist.speciesbot.t0_21.mawi.list <- reshape2::melt(dist.speciesbot.t0_21.mawi.df)[reshape2::melt(upper.t
names(dist.speciesbot.t0_21.mawi.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_21.mawi.list <- cbind(dist.speciesbot.t0_21.mawi.list, "comp" = "T0_vs_T21", "type"

# T0 vs T40, Mawi, speciesbot

st.speciesbot.t0_40.mawi <- subset_samples(st.speciesbot.r, Type == 'Mawi' & Day == '0' | Type == 'Mawi
dist.speciesbot.t0_40.mawi <- phyloseq::distance(physeq = st.speciesbot.t0_40.mawi, method = 'bray')
dist.speciesbot.t0_40.mawi.df <- as.matrix(dist.speciesbot.t0_40.mawi)
dist.speciesbot.t0_40.mawi.list <- reshape2::melt(dist.speciesbot.t0_40.mawi.df)[reshape2::melt(upper.t
names(dist.speciesbot.t0_40.mawi.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_40.mawi.list <- cbind(dist.speciesbot.t0_40.mawi.list, "comp" = "T0_vs_T40", "type"

#combine all tables
dist.per.time_comp.speciesbot.mawi <- rbind(dist.speciesbot.t0_3.mawi.list, dist.speciesbot.t0_7.mawi.l
                                            dist.speciesbot.t0_40.mawi.list)

#plot
boxplot.speciesbot.mawi <- ggplot(dist.per.time_comp.speciesbot.mawi, aes(x = comp, y = distance, fill
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") + theme_bw() + theme(axis.title.x=element_blan
  coord_cartesian(ylim = c(0, 0.27)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5))
```

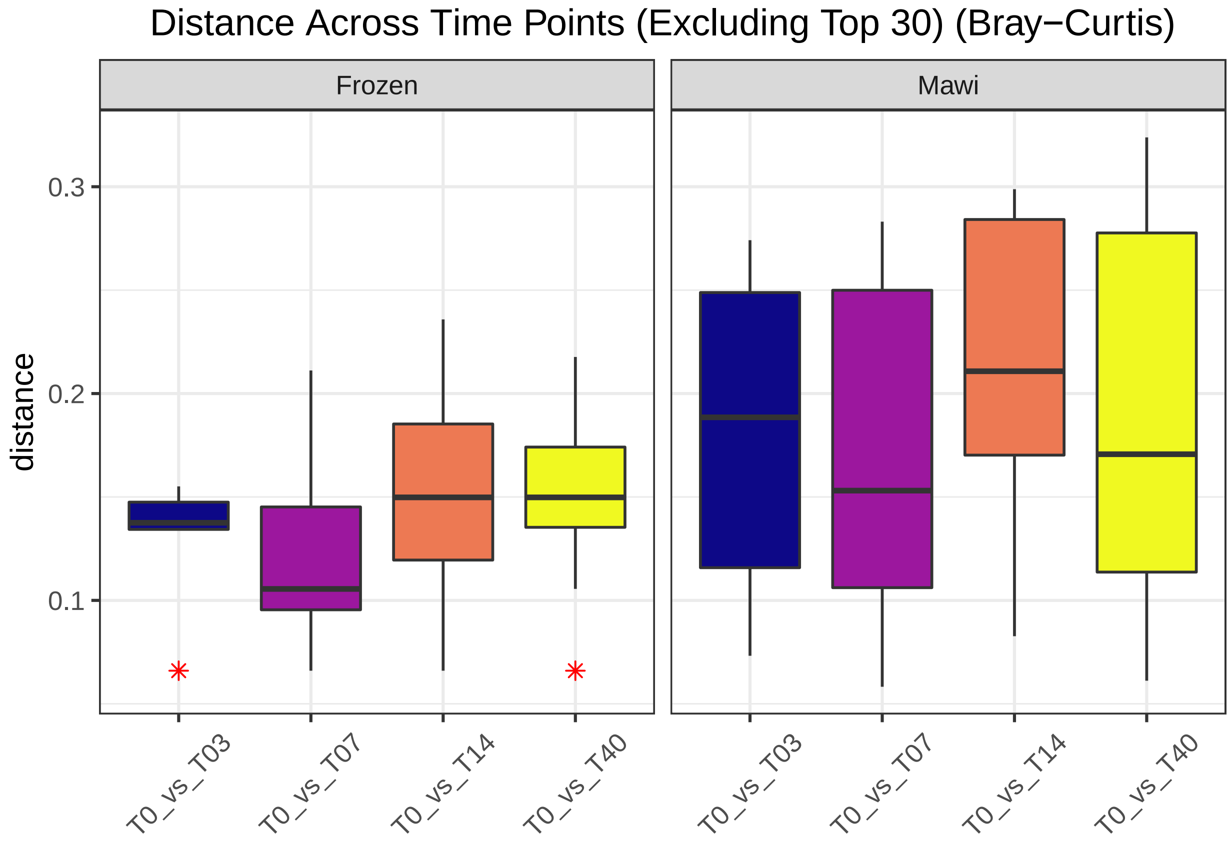
```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```

```
# T0 vs T3, Frozen, speciesbot

st.speciesbot.t0_3.frozen <- subset_samples(st.speciesbot.r, Type == 'Frozen' & Day == '0' | Type == 'F
dist.speciesbot.t0_3.frozen <- phyloseq::distance(physeq = st.speciesbot.t0_3.frozen, method = 'bray')
dist.speciesbot.t0_3.frozen.df <- as.matrix(dist.speciesbot.t0_3.frozen)
dist.speciesbot.t0_3.frozen.list <- reshape2::melt(dist.speciesbot.t0_3.frozen.df)[reshape2::melt(upper
names(dist.speciesbot.t0_3.frozen.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_3.frozen.list <- cbind(dist.speciesbot.t0_3.frozen.list, "comp" = "T0_vs_T03", "type

# T0 vs T7, Frozen, speciesbot

st.speciesbot.t0_7.frozen <- subset_samples(st.speciesbot.r, Type == 'Frozen' & Day == '0' | Type == 'F
dist.speciesbot.t0_7.frozen <- phyloseq::distance(physeq = st.speciesbot.t0_7.frozen, method = 'bray')
dist.speciesbot.t0_7.frozen.df <- as.matrix(dist.speciesbot.t0_7.frozen)
dist.speciesbot.t0_7.frozen.list <- reshape2::melt(dist.speciesbot.t0_7.frozen.df)[reshape2::melt(upper
names(dist.speciesbot.t0_7.frozen.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_7.frozen.list <- cbind(dist.speciesbot.t0_7.frozen.list, "comp" = "T0_vs_T07", "type

# T0 vs T14, Frozen, speciesbot

st.speciesbot.t0_14.frozen <- subset_samples(st.speciesbot.r, Type == 'Frozen' & Day == '0' | Type == 'F
dist.speciesbot.t0_14.frozen <- phyloseq::distance(physeq = st.speciesbot.t0_14.frozen, method = 'bray')
dist.speciesbot.t0_14.frozen.df <- as.matrix(dist.speciesbot.t0_14.frozen)
dist.speciesbot.t0_14.frozen.list <- reshape2::melt(dist.speciesbot.t0_14.frozen.df)[reshape2::melt(uppe
names(dist.speciesbot.t0_14.frozen.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_14.frozen.list <- cbind(dist.speciesbot.t0_14.frozen.list, "comp" = "T0_vs_T14", "typ

# T0 vs T21, Frozen, speciesbot

st.speciesbot.t0_21.frozen <- subset_samples(st.speciesbot.r, Type == 'Frozen' & Day == '0' | Type == 'F
dist.speciesbot.t0_21.frozen <- phyloseq::distance(physeq = st.speciesbot.t0_21.frozen, method = 'bray')
dist.speciesbot.t0_21.frozen.df <- as.matrix(dist.speciesbot.t0_21.frozen)
dist.speciesbot.t0_21.frozen.list <- reshape2::melt(dist.speciesbot.t0_21.frozen.df)[reshape2::melt(uppe
names(dist.speciesbot.t0_21.frozen.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_21.frozen.list <- cbind(dist.speciesbot.t0_21.frozen.list, "comp" = "T0_vs_T21", "typ

# T0 vs T40, Frozen, speciesbot

st.speciesbot.t0_40.frozen <- subset_samples(st.speciesbot.r, Type == 'Frozen' & Day == '0' | Type == 'F
dist.speciesbot.t0_40.frozen <- phyloseq::distance(physeq = st.speciesbot.t0_40.frozen, method = 'bray')
dist.speciesbot.t0_40.frozen.df <- as.matrix(dist.speciesbot.t0_40.frozen)
dist.speciesbot.t0_40.frozen.list <- reshape2::melt(dist.speciesbot.t0_40.frozen.df)[reshape2::melt(uppe
names(dist.speciesbot.t0_40.frozen.list) <- c("c1", "c2", "distance")
dist.speciesbot.t0_40.frozen.list <- cbind(dist.speciesbot.t0_40.frozen.list, "comp" = "T0_vs_T40", "typ

#combine all tables
dist.per.time_comp.speciesbot.frozen <- rbind(dist.speciesbot.t0_3.frozen.list, dist.speciesbot.t0_7.fro
                                              dist.speciesbot.t0_40.frozen.list)

#plot
boxplot.speciesbot.frozen <- ggplot(dist.per.time_comp.speciesbot.frozen, aes(x = comp, y = distance, fi
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") + theme_bw() + theme(axis.title.x=element_blank
  coord_cartesian(ylim = c(0, 0.27)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5))
```

```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```

```
dist.per.time_comp.speciesbot <- rbind(dist.per.time_comp.speciesbot.frozen, dist.per.time_comp.species

boxplot.speciesbot <- ggplot(dist.per.time_comp.speciesbot, aes(x = comp, y = distance, fill = comp, ))
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") +
  facet_grid(. ~ type) +
  theme_bw() +
  theme(axis.title.x = element_blank(), legend.position = "none") +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5)) +
  ggtitle("Distance Across Time Points (Excluding Top 30) (Bray-Curtis)") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
boxplot.speciesbot
```



```
ggsave("boxplot_across_time_points (bray) (reps removed) (not top 30).png",
       width = 16,
       height = 9,
       units = "in")
```

Distances for all species

```
# T0 vs T3, Mawi, physeq.dna

st.physeq.dna.t0_3.mawi <- subset_samples(st.physeq.dna.r.filtered, Type == 'Mawi' & Day == '0' | Type =
dist.physeq.dna.t0_3.mawi <- phyloseq::distance(physeq = st.physeq.dna.t0_3.mawi, method = 'bray')
dist.physeq.dna.t0_3.mawi.df <- as.matrix(dist.physeq.dna.t0_3.mawi)
dist.physeq.dna.t0_3.mawi.list <- reshape2::melt(dist.physeq.dna.t0_3.mawi.df)[reshape2::melt(upper.tri
names(dist.physeq.dna.t0_3.mawi.list) <- c("c1", "c2", "distance")
```

```
dist.physeq.dna.t0_3.mawi.list <- cbind(dist.physeq.dna.t0_3.mawi.list, "comp" = "T0_vs_T03", "type" =

# T0 vs T7, Mawi, physeq.dna

st.physeq.dna.t0_7.mawi <- subset_samples(st.physeq.dna.r.filtered, Type == 'Mawi' & Day == '0' | Type =
dist.physeq.dna.t0_7.mawi <- phyloseq::distance(physeq = st.physeq.dna.t0_7.mawi, method = 'bray')
dist.physeq.dna.t0_7.mawi.df <- as.matrix(dist.physeq.dna.t0_7.mawi)
dist.physeq.dna.t0_7.mawi.list <- reshape2::melt(dist.physeq.dna.t0_7.mawi.df)[reshape2::melt(upper.tri
names(dist.physeq.dna.t0_7.mawi.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_7.mawi.list <- cbind(dist.physeq.dna.t0_7.mawi.list, "comp" = "T0_vs_T07", "type" =

# T0 vs T14, Mawi, physeq.dna

st.physeq.dna.t0_14.mawi <- subset_samples(st.physeq.dna.r.filtered, Type == 'Mawi' & Day == '0' | Type
dist.physeq.dna.t0_14.mawi <- phyloseq::distance(physeq = st.physeq.dna.t0_14.mawi, method = 'bray')
dist.physeq.dna.t0_14.mawi.df <- as.matrix(dist.physeq.dna.t0_14.mawi)
dist.physeq.dna.t0_14.mawi.list <- reshape2::melt(dist.physeq.dna.t0_14.mawi.df)[reshape2::melt(upper.t
names(dist.physeq.dna.t0_14.mawi.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_14.mawi.list <- cbind(dist.physeq.dna.t0_14.mawi.list, "comp" = "T0_vs_T14", "type"

# T0 vs T21, Mawi, physeq.dna

st.physeq.dna.t0_21.mawi <- subset_samples(st.physeq.dna.r.filtered, Type == 'Mawi' & Day == '0' | Type
dist.physeq.dna.t0_21.mawi <- phyloseq::distance(physeq = st.physeq.dna.t0_21.mawi, method = 'bray')
dist.physeq.dna.t0_21.mawi.df <- as.matrix(dist.physeq.dna.t0_21.mawi)
dist.physeq.dna.t0_21.mawi.list <- reshape2::melt(dist.physeq.dna.t0_21.mawi.df)[reshape2::melt(upper.t
names(dist.physeq.dna.t0_21.mawi.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_21.mawi.list <- cbind(dist.physeq.dna.t0_21.mawi.list, "comp" = "T0_vs_T21", "type"

# T0 vs T40, Mawi, physeq.dna

st.physeq.dna.t0_40.mawi <- subset_samples(st.physeq.dna.r.filtered, Type == 'Mawi' & Day == '0' | Type
dist.physeq.dna.t0_40.mawi <- phyloseq::distance(physeq = st.physeq.dna.t0_40.mawi, method = 'bray')
dist.physeq.dna.t0_40.mawi.df <- as.matrix(dist.physeq.dna.t0_40.mawi)
dist.physeq.dna.t0_40.mawi.list <- reshape2::melt(dist.physeq.dna.t0_40.mawi.df)[reshape2::melt(upper.t
names(dist.physeq.dna.t0_40.mawi.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_40.mawi.list <- cbind(dist.physeq.dna.t0_40.mawi.list, "comp" = "T0_vs_T40", "type"

#combine all tables
dist.per.time_comp.physeq.dna.mawi <- rbind(dist.physeq.dna.t0_3.mawi.list, dist.physeq.dna.t0_7.mawi.l
                                            dist.physeq.dna.t0_40.mawi.list)

#plot
boxplot.physeq.dna.mawi <- ggplot(dist.per.time_comp.physeq.dna.mawi, aes(x = comp, y = distance, fill
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") + theme_bw() + theme(axis.title.x=element_blan
  coord_cartesian(ylim = c(0, 0.27)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5))
```

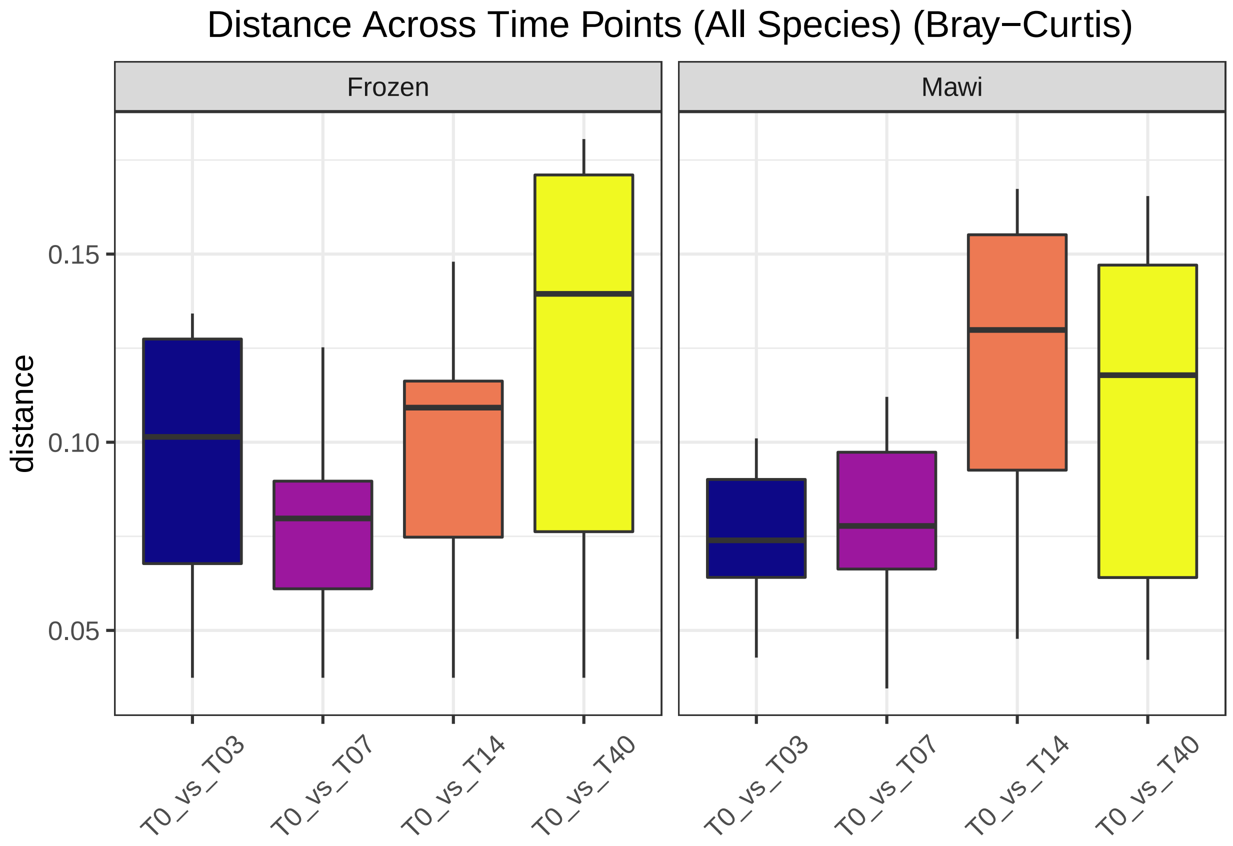
```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```

```
# T0 vs T3, Frozen, physeq.dna

st.physeq.dna.t0_3.frozen <- subset_samples(st.physeq.dna.r.filtered, Type == 'Frozen' & Day == '0' | Ty
dist.physeq.dna.t0_3.frozen <- phyloseq::distance(physeq = st.physeq.dna.t0_3.frozen, method = 'bray')
dist.physeq.dna.t0_3.frozen.df <- as.matrix(dist.physeq.dna.t0_3.frozen)
dist.physeq.dna.t0_3.frozen.list <- reshape2::melt(dist.physeq.dna.t0_3.frozen.df)[reshape2::melt(upper
names(dist.physeq.dna.t0_3.frozen.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_3.frozen.list <- cbind(dist.physeq.dna.t0_3.frozen.list, "comp" = "T0_vs_T03", "type

# T0 vs T7, Frozen, physeq.dna

st.physeq.dna.t0_7.frozen <- subset_samples(st.physeq.dna.r.filtered, Type == 'Frozen' & Day == '0' | Ty
dist.physeq.dna.t0_7.frozen <- phyloseq::distance(physeq = st.physeq.dna.t0_7.frozen, method = 'bray')
dist.physeq.dna.t0_7.frozen.df <- as.matrix(dist.physeq.dna.t0_7.frozen)
dist.physeq.dna.t0_7.frozen.list <- reshape2::melt(dist.physeq.dna.t0_7.frozen.df)[reshape2::melt(upper
names(dist.physeq.dna.t0_7.frozen.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_7.frozen.list <- cbind(dist.physeq.dna.t0_7.frozen.list, "comp" = "T0_vs_T07", "type

# T0 vs T14, Frozen, physeq.dna

st.physeq.dna.t0_14.frozen <- subset_samples(st.physeq.dna.r.filtered, Type == 'Frozen' & Day == '0' | T
dist.physeq.dna.t0_14.frozen <- phyloseq::distance(physeq = st.physeq.dna.t0_14.frozen, method = 'bray')
dist.physeq.dna.t0_14.frozen.df <- as.matrix(dist.physeq.dna.t0_14.frozen)
dist.physeq.dna.t0_14.frozen.list <- reshape2::melt(dist.physeq.dna.t0_14.frozen.df)[reshape2::melt(uppe
names(dist.physeq.dna.t0_14.frozen.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_14.frozen.list <- cbind(dist.physeq.dna.t0_14.frozen.list, "comp" = "T0_vs_T14", "ty

# T0 vs T21, Frozen, physeq.dna

st.physeq.dna.t0_21.frozen <- subset_samples(st.physeq.dna.r.filtered, Type == 'Frozen' & Day == '0' | T
dist.physeq.dna.t0_21.frozen <- phyloseq::distance(physeq = st.physeq.dna.t0_21.frozen, method = 'bray')
dist.physeq.dna.t0_21.frozen.df <- as.matrix(dist.physeq.dna.t0_21.frozen)
dist.physeq.dna.t0_21.frozen.list <- reshape2::melt(dist.physeq.dna.t0_21.frozen.df)[reshape2::melt(uppe
names(dist.physeq.dna.t0_21.frozen.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_21.frozen.list <- cbind(dist.physeq.dna.t0_21.frozen.list, "comp" = "T0_vs_T21", "ty

# T0 vs T40, Frozen, physeq.dna

st.physeq.dna.t0_40.frozen <- subset_samples(st.physeq.dna.r.filtered, Type == 'Frozen' & Day == '0' | T
dist.physeq.dna.t0_40.frozen <- phyloseq::distance(physeq = st.physeq.dna.t0_40.frozen, method = 'bray')
dist.physeq.dna.t0_40.frozen.df <- as.matrix(dist.physeq.dna.t0_40.frozen)
dist.physeq.dna.t0_40.frozen.list <- reshape2::melt(dist.physeq.dna.t0_40.frozen.df)[reshape2::melt(uppe
names(dist.physeq.dna.t0_40.frozen.list) <- c("c1", "c2", "distance")
dist.physeq.dna.t0_40.frozen.list <- cbind(dist.physeq.dna.t0_40.frozen.list, "comp" = "T0_vs_T40", "ty

#combine all tables
dist.per.time_comp.physeq.dna.frozen <- rbind(dist.physeq.dna.t0_3.frozen.list, dist.physeq.dna.t0_7.fro
                                              dist.physeq.dna.t0_40.frozen.list)

#plot
boxplot.physeq.dna.frozen <- ggplot(dist.per.time_comp.physeq.dna.frozen, aes(x = comp, y = distance, fi
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") + theme_bw() + theme(axis.title.x=element_blank
  coord_cartesian(ylim = c(0, 0.27)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5))
```

```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```

```
dist.per.time_comp.physeq.dna <- rbind(dist.per.time_comp.physeq.dna.frozen, dist.per.time_comp.physeq.d

boxplot.physeq.dna <- ggplot(dist.per.time_comp.physeq.dna, aes(x = comp, y = distance, fill = comp, ))
  geom_boxplot(outlier.colour = "red", outlier.shape = 8,
               outlier.size = 2) +
  scale_fill_viridis(discrete = TRUE, option = "plasma") +
  facet_grid(. ~ type) +
  theme_bw() +
  theme(axis.title.x = element_blank(), legend.position = "none") +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5)) +
  ggtitle("Distance Across Time Points (All Species) (Bray-Curtis)") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
boxplot.physeq.dna
```



```
ggsave("boxplot_across_time_points (bray) (reps removed) (all).png",
       width = 16,
       height = 9,
       units = "in")
```

NMDS (Bray-Curtis & Jaccard)

```
# Ordinate
set.seed(13)
st.physeq.nmds.bray.dna <- ordinate(
  physeq = st.physeq.dna.r,
  method = 'NMDS',
  distance = 'bray'
)
```

```
## Square root transformation
## Wisconsin double standardization
## Run 0 stress 0.155633
## Run 1 stress 0.1719016
## Run 2 stress 0.1756162
## Run 3 stress 0.1566107
## Run 4 stress 0.1879288
## Run 5 stress 0.1569191
## Run 6 stress 0.1718283
## Run 7 stress 0.1719818
## Run 8 stress 0.1840125
## Run 9 stress 0.1556477
## ... Procrustes: rmse 0.001812098  max resid 0.008082778
## ... Similar to previous best
## Run 10 stress 0.1566105
## Run 11 stress 0.1566106
## Run 12 stress 0.1725652
## Run 13 stress 0.1556477
## ... Procrustes: rmse 0.001805664  max resid 0.008050367
## ... Similar to previous best
## Run 14 stress 0.1847536
## Run 15 stress 0.1569191
## Run 16 stress 0.1560587
## ... Procrustes: rmse 0.01345158  max resid 0.06651076
## Run 17 stress 0.1860244
## Run 18 stress 0.2060177
## Run 19 stress 0.155633
## ... Procrustes: rmse 0.00004624573  max resid 0.0001889374
## ... Similar to previous best
## Run 20 stress 0.1556477
## ... Procrustes: rmse 0.001801793  max resid 0.008034905
## ... Similar to previous best
## *** Solution reached
```

```
bray_ord <- phyloseq::distance(physeq = st.physeq.dna.r, method = 'bray')

df_ord <- data.frame(sample_data(st.physeq.dna.r))
adonis(bray_ord ~ Type, data = df_ord)
```

```
##
## Call:
## adonis(formula = bray_ord ~ Type, data = df_ord)
```

```
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## Type       1   0.04651 0.046511  5.0681 0.12973  0.001 ***
## Residuals 34   0.31202 0.009177         0.87027
## Total     35   0.35854                  1.00000
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
beta <- betadisper(bray_ord, df_ord$Type)
permutest(beta)
```

```
##
## 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     1 0.001031 0.0010313 1.0043    999  0.288
## Residuals 34 0.034914 0.0010269
```

```
nmds.bray.dna <- plot_ordination(
  physeq = st.physeq.dna.r,
  ordination <- st.physeq.nmds.bray.dna,
  color = 'Nucleic.Acid',
  title = 'nMDS of Bacterial Communities in Stool DNA Samples (Bray-Curtis)'
) +
  scale_color_manual(values = c('Frozen' = '#6BA7CC', 'Mawi' = '#ff6600')) +
  geom_point(aes(color = Type), size = 3) +
  theme(plot.title = element_text(hjust = 0.5)) +
  annotate("text", x = 0.2, y = 0.11, size = 3, label = "\n Stress=0.15\n Adonis\n R^2=0.12\n p=0.001")
```

```
set.seed(13)
# Ordinate
st.physeq.nmds.jaccard.dna <- ordinate(
  physeq = st.physeq.dna.r,
  method = 'NMDS',
  distance = 'jaccard'
)
```

```
## Square root transformation
## Wisconsin double standardization
## Run 0 stress 0.1556331
## Run 1 stress 0.1719016
## Run 2 stress 0.1756162
## Run 3 stress 0.1566107
## Run 4 stress 0.1879288
## Run 5 stress 0.1569191
## Run 6 stress 0.1718283
## Run 7 stress 0.1719818
## Run 8 stress 0.1840125
## Run 9 stress 0.1556477
## ... Procrustes: rmse 0.001814411  max resid 0.008071431
## ... Similar to previous best
## Run 10 stress 0.1566105
## Run 11 stress 0.1566106
## Run 12 stress 0.1725652
## Run 13 stress 0.1556477
## ... Procrustes: rmse 0.001809768  max resid 0.008039023
## ... Similar to previous best
## Run 14 stress 0.1847536
## Run 15 stress 0.1569191
## Run 16 stress 0.1560587
## ... Procrustes: rmse 0.01349113  max resid 0.06669918
## Run 17 stress 0.1860244
## Run 18 stress 0.2060177
## Run 19 stress 0.155633
## ... New best solution
## ... Procrustes: rmse 0.00002810293  max resid 0.0001063567
## ... Similar to previous best
## Run 20 stress 0.1556477
## ... Procrustes: rmse 0.001804419  max resid 0.008027621
## ... Similar to previous best
## *** Solution reached
```

```
jaccard_ord <- phyloseq::distance(physeq = st.physeq.dna.r, method = 'jaccard')

df_ord <- data.frame(sample_data(st.physeq.dna.r))
adonis(jaccard_ord ~ Type, data = df_ord)
```

```
##
## Call:
## adonis(formula = jaccard_ord ~ Type, data = df_ord)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##           Df SumsOfSqs  MeanSqs F.Model      R2 Pr(>F)
## Type       1   0.12856 0.128561  4.6861 0.12113  0.001 ***
## Residuals 34   0.93277 0.027434         0.87887
## Total     35   1.06133                  1.00000
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

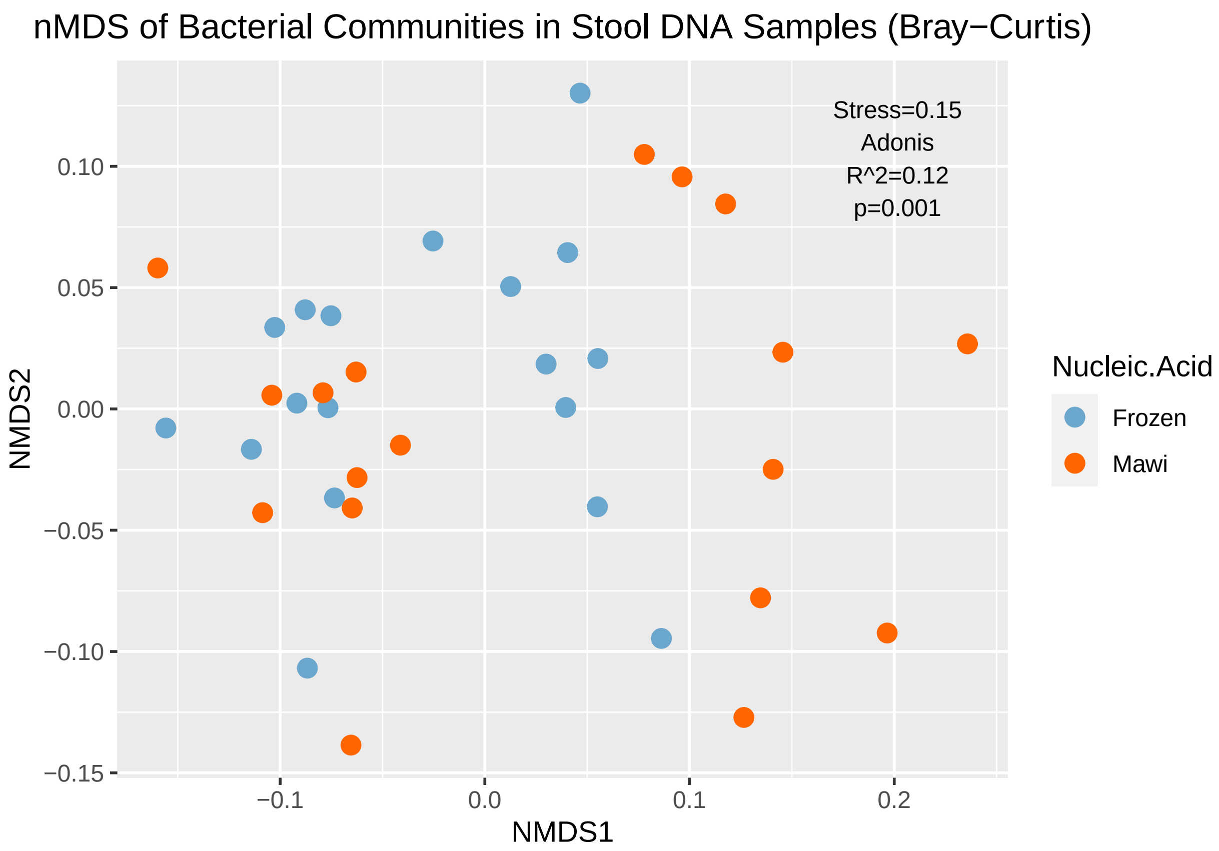
```
beta <- betadisper(jaccard_ord, df_ord$Type)
permutest(beta)
```

```
##
```

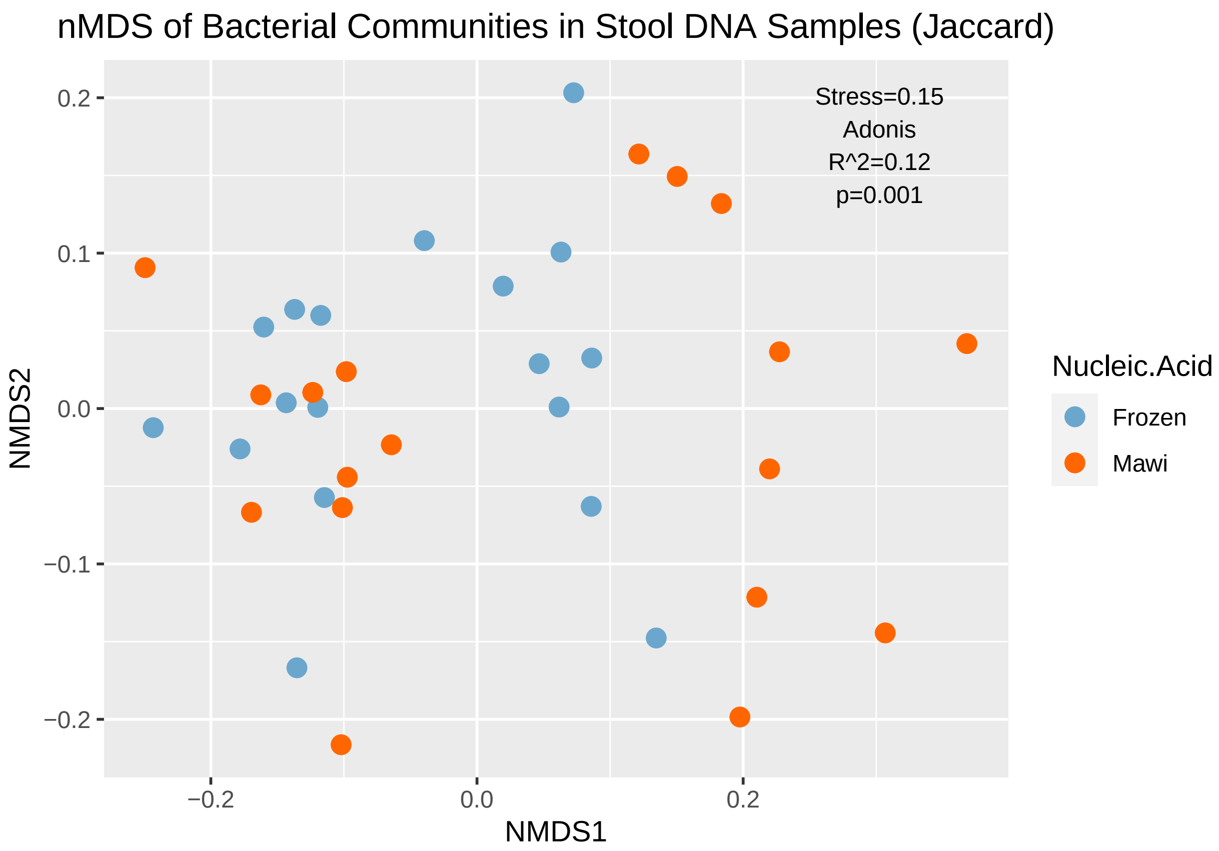
```
## Permutation test for homogeneity of multivariate dispersions
## Permutation: free
## Number of permutations: 999
##
## Response: Distances
##           Df   Sum Sq   Mean Sq      F N.Perm Pr(>F)
## Groups     1 0.002296 0.0022961 0.9504    999  0.297
## Residuals 34 0.082145 0.0024160
```

```
nmds.jaccard.dna <- plot_ordination(
  physeq = st.physeq.dna.r,
  ordination <- st.physeq.nmds.jaccard.dna,
  color = 'Nucleic.Acid',
  title = 'nMDS of Bacterial Communities in Stool DNA Samples (Jaccard)'
) +
  scale_color_manual(values = c('Frozen' = '#6BA7CC', 'Mawi' = '#ff6600')) +
  geom_point(aes(color = Type), size = 3) +
  theme(plot.title = element_text(hjust = 0.5)) +
  annotate("text", x = 0.30, y = 0.18, size = 3, label = "\n Stress=0.15\n Adonis\n R^2=0.12\n p=0.001"
```

```
nmds.bray.dna
```



```
ggsave("nmds_bray_dna_all.pdf",
       width = 6.5,
       height = 4,
       units = "in")

nmds.jaccard.dna
```



```
ggsave("nmds_jaccard_dna_all.pdf",
       width = 6.5,
       height = 4,
       units = "in")
```

Abundance Time Series

```
#merged.day <- merge_samples(st.physeq.dna.r, "Day.Unique")


#obj <- merged.day
#otutable <- data.frame(OTU = rownames(phyloseq::otu_table(obj)@.Data),
                       #phyloseq::otu_table(obj)@.Data,
                       #phyloseq::tax_table(obj)@.Data,
                       #check.names = FALSE)

#metadata <- data.frame(phyloseq::sample_data(obj),
                       #check.names = FALSE)
```

```r
#my_ampvis2_object_merged <- amp_load(otutable, metadata)

#amp_timeseries(my_ampvis2_object_merged,
  #time_variable = "Date",
  #group_by = "Type",
  #split = TRUE,
  #scales = "free_y",
  #tax_show = 12,
  #tax_aggregate = "Species"#,
  #tax_add = "Phylum"
#) +
  #scale_color_manual(values = c('Frozen' = '#6BA7CC', 'Mawi' = '#ff6600')) +
  #ggtitle("Relative Abundance Across Time") +
  #theme(plot.title = element_text(hjust = 0.5))

#ggsave("relative abundance time series.png",
 #      width = 16,
  #     height = 9,
   #    unit = "in")
```

SIMPER analysis

```r
st.physeq.dna.r.abund <- as(otu_table(st.physeq.dna.r), "matrix")

if(taxa_are_rows(st.physeq.dna.r)){st.physeq.dna.r.abund <- t(st.physeq.dna.r.abund)}

st.physeq.dna.r.abund <- as.data.frame(st.physeq.dna.r.abund)

map_dna <- subset(map, Nucleic.Acid == "DNA")

type_simper <- simper(st.physeq.dna.r.abund, map_dna$Type, permutations = 100)

summary(type_simper)
```

```
##
## Contrast: Frozen_Mawi
##
##              average           sd  ratio         ava         avb  cumsum
## ST_561 0.01415156526 0.0098618697 1.4350 24472.66667 23371.72222 0.09866
## ST_343 0.00732568342 0.0049837840 1.4699 14688.72222 11479.83333 0.14973
## ST_290 0.00609954806 0.0058154505 1.0489 11460.83333 13025.66667 0.19225
## ST_201 0.00570692240 0.0138325817 0.4126  2440.11111  1194.16667 0.23204
## ST_337 0.00495792549 0.0046708664 1.0615  8881.94444 11061.05556 0.26660
## ST_500 0.00413460097 0.0027512754 1.5028  4430.44444  5689.27778 0.29542
## ST_099 0.00400978836 0.0031522575 1.2720  7986.94444  8682.00000 0.32338
## ST_100 0.00322146164 0.0018200351 1.7700  3730.83333  5441.50000 0.34584
## ST_614 0.00311240079 0.0021704068 1.4340  8984.88889  9313.66667 0.36753
## ST_120 0.00306339286 0.0025514569 1.2006 10936.05556 10279.22222 0.38889
## ST_336 0.00304931658 0.0021407578 1.4244  7886.94444  8313.66667 0.41015
## ST_296 0.00291475970 0.0019601049 1.4870  6921.61111  8025.11111 0.43047
## ST_250 0.00285496032 0.0019775057 1.4437  5599.77778  4327.72222 0.45037
## ST_248 0.00282348986 0.0019501352 1.4478  5396.44444  4146.83333 0.47006
## ST_115 0.00264958113 0.0021116015 1.2548  9626.27778  9133.55556 0.48853
## ST_153 0.00232127425 0.0016703604 1.3897  6835.33333  7092.88889 0.50471
## ST_295 0.00227750220 0.0011166901 2.0395  2879.88889  1607.88889 0.52059
## ST_061 0.00219049824 0.0015719749 1.3935  3433.33333  3525.22222 0.53586
## ST_331 0.00207723765 0.0014246477 1.4581  3789.16667  4018.11111 0.55034
## ST_244 0.00198619929 0.0016810300 1.1815  1841.33333  2360.83333 0.56419
## ST_534 0.00189382716 0.0014309858 1.3234  7970.22222  7281.77778 0.57739
## ST_550 0.00183117284 0.0016146310 1.1341  2856.22222  3692.77778 0.59016
## ST_137 0.00182008377 0.0014092891 1.2915  4563.27778  4746.72222 0.60284
## ST_557 0.00175500441 0.0016132458 1.0879   704.22222   867.77778 0.61508
## ST_119 0.00172426146 0.0013640394 1.2641  2047.77778  2032.55556 0.62710
## ST_130 0.00162141755 0.0012463405 1.3009  4281.44444  3895.77778 0.63840
## ST_144 0.00161165123 0.0011867396 1.3580  4659.61111  4887.05556 0.64964
## ST_045 0.00160170855 0.0011540583 1.3879  8182.88889  8319.94444 0.66080
## ST_612 0.00159715608 0.0011748151 1.3595  1844.77778  2660.55556 0.67194
## ST_536 0.00159149030 0.0011050175 1.4402  5688.83333  4905.44444 0.68303
## ST_438 0.00155057319 0.0011481847 1.3505  2963.11111  3017.16667 0.69384
## ST_043 0.00134933862 0.0009299535 1.4510  1975.05556  1784.11111 0.70325
## ST_094 0.00132059083 0.0010794464 1.2234  3632.88889  3085.27778 0.71246
## ST_687 0.00127997134 0.0008418174 1.5205  1393.00000  2034.33333 0.72138
## ST_152 0.00127854938 0.0009176339 1.3933  5962.94444  5935.27778 0.73029
## ST_246 0.00123732363 0.0022005548 0.5623   415.44444   369.77778 0.73892
## ST_525 0.00121052690 0.0009020694 1.3419  3823.22222  3752.50000 0.74736
## ST_081 0.00117129630 0.0008418653 1.3913  2455.88889  2861.22222 0.75552
## ST_190 0.00114883157 0.0008125056 1.4139  2250.55556  2193.22222 0.76353
## ST_731 0.00112135141 0.0004158601 2.6965  1124.61111   599.27778 0.77135
## ST_330 0.00109272487 0.0007693984 1.4202   718.05556  1292.38889 0.77897
## ST_297 0.00108777557 0.0005702840 1.9074  1472.72222   867.00000 0.78655
## ST_093 0.00103747795 0.0007542233 1.3756  4056.11111  3698.38889 0.79378
## ST_615 0.00101765873 0.0007534942 1.3506  3273.66667  3368.77778 0.80088
## ST_236 0.00097309303 0.0006321061 1.5394  3450.00000  3830.66667 0.80766
## ST_518 0.00095641534 0.0005501435 1.7385  1030.72222  1544.88889 0.81433
## ST_680 0.00094736552 0.0006930326 1.3670  1865.44444  1970.61111 0.82094
## ST_033 0.00092755732 0.0006127896 1.5137  1090.88889   612.33333 0.82740
## ST_176 0.00091176146 0.0006604518 1.3805  1329.22222  1555.16667 0.83376
## ST_539 0.00083414903 0.0010769599 0.7745  2088.66667  1701.33333 0.83957
## ST_262 0.00080974427 0.0005785971 1.3995  2577.22222  2184.83333 0.84522
## ST_595 0.00079502866 0.0005727786 1.3880  1945.38889  1602.72222 0.85076
## ST_469 0.00079260362 0.0007633646 1.0383   312.83333   351.94444 0.85629
## ST_117 0.00076513448 0.0006245487 1.2251  2020.50000  2090.66667 0.86162
## ST_090 0.00072832892 0.0005741967 1.2684  2730.72222  2632.05556 0.86670
## ST_169 0.00072644400 0.0005388864 1.3480   550.00000   944.83333 0.87176
## ST_613 0.00072149471 0.0005303859 1.3603  1580.38889  1743.72222 0.87679
## ST_096 0.00066079145 0.0004877252 1.3548  1784.83333  1855.66667 0.88140
## ST_677 0.00064643959 0.0004933289 1.3104   963.94444  1150.16667 0.88591
## ST_104 0.00063165785 0.0004488762 1.4072  2027.61111  1991.88889 0.89031
## ST_593 0.00060940256 0.0005044705 1.2080   825.00000   538.00000 0.89456
## ST_556 0.00057078924 0.0003738151 1.5269  1326.16667  1046.50000 0.89854
## ST_261 0.00055418871 0.0003945126 1.4047  1923.44444  2179.05556 0.90240
## ST_626 0.00052742504 0.0003745522 1.4081   859.94444   862.94444 0.90608
## ST_335 0.00052472443 0.0003857035 1.3604  1000.61111   730.16667 0.90974
## ST_047 0.00052285053 0.0004095512 1.2766  1164.11111  1220.88889 0.91338
## ST_528 0.00052061287 0.0003350900 1.5537   876.16667   651.05556 0.91701
## ST_533 0.00049896384 0.0014056058 0.3550    1.61111  278.50000 0.92049
## ST_147 0.00048228616 0.0003375970 1.4286 1389.72222 1239.11111 0.92385
## ST_410 0.00035737434 0.0003065414 1.1658  111.22222  281.88889 0.92634
## ST_086 0.00035465168 0.0002839132 1.2492  193.55556  286.44444 0.92881
## ST_039 0.00030227072 0.0003070239 0.9845  255.94444  315.61111 0.93092
## ST_493 0.00027246473 0.0002944709 0.9253   77.05556  175.22222 0.93282
## ST_322 0.00026485891 0.0001960010 1.3513  162.83333  142.44444 0.93467
## ST_109 0.00025027557 0.0001864373 1.3424  569.88889  508.88889 0.93641
## ST_192 0.00024631834 0.0003567005 0.6905  106.33333   54.38889 0.93813
## ST_208 0.00023190035 0.0001936642 1.1974  312.55556  233.88889 0.93975
## ST_046 0.00020373677 0.0001460992 1.3945  411.88889  433.11111 0.94117
## ST_569 0.00020325176 0.0001430799 1.4205  240.72222  330.27778 0.94258
## ST_325 0.00020208333 0.0001460012 1.3841  362.66667  345.72222 0.94399
## ST_501 0.00019789462 0.0001258447 1.5725  202.88889  258.33333 0.94537
## ST_085 0.00019131393 0.0001270871 1.5054  481.72222  507.38889 0.94671
## ST_049 0.00017985009 0.0001431240 1.2566  175.44444  208.44444 0.94796
## ST_212 0.00017946429 0.0001242743 1.4441  343.27778  260.16667 0.94921
## ST_121 0.00017125220 0.0001556209 1.1004  433.00000  434.66667 0.95040
## ST_221 0.00014658289 0.0001040318 1.4090  151.44444   71.11111 0.95143
## ST_044 0.00014113757 0.0001089466 1.2955  278.11111  310.05556 0.95241
## ST_202 0.00014097222 0.0001419999 0.9928   67.55556   71.83333 0.95339
## ST_311 0.00013544974 0.0001025951 1.3202   93.94444  106.72222 0.95434
## ST_594 0.00013314594 0.0001210102 1.1003  127.77778  154.55556 0.95527
## ST_383 0.00012955247 0.0000990445 1.3080  118.83333  107.83333 0.95617
## ST_559 0.00011727293 0.0001005188 1.1667   60.44444   75.88889 0.95699
## ST_196 0.00011707451 0.0000812829 1.4403  504.38889  554.66667 0.95780
## ST_607 0.00011536596 0.0000984002 1.1724  172.00000  218.77778 0.95861
## ST_306 0.00010953483 0.0000779136 1.4059  148.38889  154.44444 0.95937
## ST_441 0.00010871914 0.0000702447 1.5477  306.00000  263.66667 0.96013
## ST_118 0.00010772707 0.0001071001 1.0059  141.50000  185.33333 0.96088
## ST_127 0.00010640432 0.0000789904 1.3471  306.88889  341.16667 0.96162
## ST_224 0.00010322972 0.0000630237 1.6379   78.50000   40.16667 0.96234
## ST_101 0.00010230379 0.0001301201 0.7862   29.33333   66.88889 0.96305
## ST_042 0.00009611993 0.0000759975 1.2648  148.61111  179.72222 0.96372
## ST_390 0.00009432319 0.0000732995 1.2868   64.05556   61.05556 0.96438
## ST_141 0.00009275794 0.0001873378 0.4951   57.66667   13.11111 0.96503
## ST_146 0.00009247134 0.0000639617 1.4457  187.27778  163.44444 0.96567
## ST_675 0.00009228395 0.0000610280 1.5122   75.33333   51.94444 0.96632
## ST_298 0.00009007937 0.0000751256 1.1991  126.11111  147.83333 0.96694
## ST_671 0.00008288139 0.0000551892 1.5018   65.94444   52.55556 0.96752
## ST_318 0.00008199956 0.0000625165 1.3116   71.22222   63.27778 0.96809
## ST_411 0.00008088624 0.0000548053 1.4759  103.27778   72.38889 0.96866
## ST_529 0.00007836199 0.0000561991 1.3944   61.83333   57.61111 0.96920
## ST_029 0.00007674162 0.0000752960 1.0192   90.94444   96.05556 0.96974
## ST_611 0.00007659832 0.0000990584 0.7733   43.61111   69.05556 0.97027
## ST_327 0.00007634480 0.0000538529 1.4177  353.11111  357.50000 0.97081
## ST_681 0.00007490079 0.0000531994 1.4079  210.27778  186.94444 0.97133
## ST_653 0.00007394180 0.0000570391 1.2963   45.72222   56.77778 0.97184
## ST_195 0.00007238757 0.0000511394 1.4155  244.38889  264.61111 0.97235
## ST_320 0.00007083333 0.0000488007 1.4515   90.88889   52.16667 0.97284
## ST_048 0.00006723986 0.0001360477 0.4942   20.88889   20.83333 0.97331
## ST_107 0.00006462743 0.0000555778 1.1628  172.72222  174.50000 0.97376
## ST_563 0.00006285273 0.0000757271 0.8300   19.83333   25.38889 0.97420
## ST_442 0.00005954586 0.0000430455 1.3833   85.38889  104.72222 0.97461
## ST_305 0.00005917108 0.0001040253 0.5688    28.00000    14.05556 0.97503
## ST_319 0.00005832231 0.0001048202 0.5564    22.66667    36.61111 0.97543
## ST_567 0.00005716490 0.0000409974 1.3944    97.33333   113.44444 0.97583
## ST_366 0.00005605159 0.0000605029 0.9264    12.55556    28.83333 0.97622
## ST_459 0.00005222663 0.0000800643 0.6523    15.00000    20.44444 0.97659
## ST_542 0.00005048501 0.0000410797 1.2290    89.88889   100.88889 0.97694
## ST_427 0.00005040785 0.0000522123 0.9654    25.61111    19.22222 0.97729
## ST_235 0.00004988977 0.0000421881 1.1826    48.27778    44.33333 0.97764
## ST_228 0.00004738757 0.0000387962 1.2214    69.05556    75.55556 0.97797
## ST_558 0.00004631834 0.0000389970 1.1877    13.11111    25.44444 0.97829
## ST_116 0.00004524912 0.0000334293 1.3536    73.83333    87.55556 0.97861
## ST_460 0.00004500661 0.0000580755 0.7750    11.61111    20.77778 0.97892
## ST_546 0.00004498457 0.0000482277 0.9328    32.72222    33.66667 0.97923
## ST_707 0.00004394841 0.0000496820 0.8846    23.88889     6.94444 0.97954
## ST_063 0.00004279101 0.0000752802 0.5684    15.16667    12.16667 0.97984
## ST_602 0.00004270282 0.0000322281 1.3250   114.33333   107.05556 0.98014
## ST_605 0.00004079586 0.0000357207 1.1421    37.50000    56.33333 0.98042
## ST_247 0.00003964947 0.0000422578 0.9383    15.33333    16.50000 0.98070
## ST_166 0.00003923060 0.0000353139 1.1109    38.44444    43.66667 0.98097
## ST_206 0.00003847002 0.0000257195 1.4958    77.88889    61.11111 0.98124
## ST_004 0.00003841490 0.0000364371 1.0543    18.11111    11.66667 0.98151
## ST_124 0.00003802910 0.0000942639 0.4034    16.88889     5.66667 0.98177
## ST_434 0.00003665123 0.0000251376 1.4580    50.55556    56.66667 0.98203
## ST_204 0.00003605600 0.0000369782 0.9751    30.61111    32.44444 0.98228
## ST_580 0.00003560406 0.0000260817 1.3651    83.94444    78.33333 0.98253
## ST_341 0.00003537257 0.0000251461 1.4067    59.72222    57.22222 0.98277
## ST_710 0.00003535053 0.0000268912 1.3146   139.88889   140.61111 0.98302
## ST_359 0.00003411596 0.0000997528 0.3420    11.05556     9.05556 0.98326
## ST_102 0.00003384039 0.0000258376 1.3097    58.00000    68.50000 0.98349
## ST_232 0.00003373016 0.0000249312 1.3529    62.00000    52.44444 0.98373
## ST_728 0.00003369709 0.0000241001 1.3982    22.38889    32.27778 0.98396
## ST_215 0.00003117284 0.0000201860 1.5443    61.66667    45.94444 0.98418
## ST_426 0.00003016975 0.0000253433 1.1904    14.61111    17.16667 0.98439
## ST_560 0.00003014771 0.0000276894 1.0888     9.50000    15.77778 0.98460
## ST_105 0.00002929894 0.0000209577 1.3980   102.55556    97.33333 0.98481
## ST_095 0.00002687390 0.0000193124 1.3915    12.88889    22.88889 0.98499
## ST_624 0.00002675265 0.0000170962 1.5648    13.27778    23.22222 0.98518
## ST_233 0.00002604718 0.0000234086 1.1127    29.33333    27.88889 0.98536
## ST_690 0.00002595899 0.0000185248 1.4013    16.16667    15.66667 0.98554
## ST_398 0.00002569444 0.0000710798 0.3615     5.38889     9.77778 0.98572
## ST_170 0.00002528660 0.0000201565 1.2545    72.83333    66.11111 0.98590
## ST_608 0.00002480159 0.0000155732 1.5926    29.83333    43.16667 0.98607
## ST_437 0.00002443783 0.0000164907 1.4819    27.11111    14.38889 0.98624
## ST_402 0.00002421737 0.0000166942 1.4507    42.88889    49.22222 0.98641
## ST_419 0.00002341270 0.0000318283 0.7356    12.16667     8.00000 0.98657
## ST_178 0.00002340168 0.0000194081 1.2058    38.00000    43.27778 0.98674
## ST_214 0.00002301587 0.0000322499 0.7137     5.50000    10.88889 0.98690
## ST_050 0.00002283951 0.0000188165 1.2138    25.16667    29.00000 0.98706
## ST_023 0.00002276235 0.0000207686 1.0960    27.00000    26.66667 0.98721
## ST_555 0.00002239859 0.0000315051 0.7110    11.11111     7.11111 0.98737
## ST_425 0.00002223325 0.0000173741 1.2797    16.55556    14.50000 0.98753
## ST_638 0.00002150573 0.0000174487 1.2325    11.33333    10.66667 0.98768
## ST_512 0.00002073413 0.0000317120 0.6538     7.22222     6.38889 0.98782
## ST_565 0.00002031526 0.0000225273 0.9018     8.05556     6.77778 0.98796
## ST_301 0.00002011684 0.0000151037 1.3319    16.27778     8.88889 0.98810
## ST_027 0.00002001764 0.0000280377 0.7140    17.16667    17.22222 0.98824
## ST_310 0.00001968695 0.0000416314 0.4729     5.00000     7.38889 0.98838
## ST_219 0.00001955467 0.0000148273 1.3188    52.11111    50.83333 0.98851
## ST_332 0.00001944444 0.0000145127 1.3398    21.27778    14.55556 0.98865
## ST_623 0.00001875000 0.0000168000 1.1161     7.94444     7.88889 0.98878
## ST_227 0.00001817681 0.0000135245 1.3440    30.88889    37.83333 0.98891
## ST_312 0.00001775794 0.0000141778 1.2525     9.27778     7.77778 0.98903
## ST_609 0.00001752646 0.0000131812 1.3297    15.27778    23.33333 0.98915
## ST_087 0.00001698633 0.0000148233 1.1459    34.88889    38.05556 0.98927
## ST_708 0.00001696429 0.0000245499 0.6910     3.77778     7.38889 0.98939
## ST_384 0.00001695326 0.0000219446 0.7725     6.72222     5.33333 0.98951
## ST_443 0.00001689815 0.0000123632 1.3668    15.33333    16.66667 0.98963
## ST_650 0.00001684303 0.0000253462 0.6645     9.72222    12.05556 0.98974
## ST_630 0.00001672178 0.0000142992 1.1694    21.50000    17.88889 0.98986
## ST_103 0.00001625882 0.0000114010 1.4261    23.38889    24.66667 0.98997
## ST_315 0.00001601631 0.0000112033 1.4296    17.88889    13.83333 0.99009
## ST_140 0.00001556437 0.0000171118 0.9096     4.94444     7.11111 0.99019
## ST_264 0.00001555335 0.0000101321 1.5351    23.77778    29.27778 0.99030
## ST_089 0.00001523369 0.0000113195 1.3458    29.50000    27.83333 0.99041
## ST_242 0.00001513448 0.0000114935 1.3168    21.83333    19.38889 0.99051
## ST_171 0.00001509039 0.0000148696 1.0149     5.88889    11.16667 0.99062
## ST_131 0.00001499118 0.0000184546 0.8123     7.88889     1.77778 0.99072
## ST_106 0.00001485891 0.0000119681 1.2415    19.83333    23.55556 0.99083
## ST_606 0.00001484788 0.0000106356 1.3961    17.05556    21.00000 0.99093
## ST_440 0.00001466049 0.0000110659 1.3248    22.16667    24.94444 0.99103
## ST_439 0.00001430776 0.0000099469 1.4384     8.55556    12.83333 0.99113
## ST_256 0.00001424162 0.0000173871 0.8191     3.61111     6.72222 0.99123
## ST_252 0.00001399912 0.0000132918 1.0532    16.77778    14.55556 0.99133
## ST_268 0.00001396605 0.0000111352 1.2542    21.94444    17.88889 0.99143
## ST_453 0.00001383377 0.0000301683 0.4586     1.77778     6.61111 0.99152
## ST_431 0.00001352513 0.0000115323 1.1728    10.00000    14.94444 0.99162
## ST_432 0.00001351411 0.0000313492 0.4311     4.33333     3.88889 0.99171
## ST_241 0.00001347002 0.0000103250 1.3046    15.72222    19.11111 0.99181
## ST_391 0.00001345899 0.0000118069 1.1399     5.50000     9.16667 0.99190
## ST_444 0.00001316138 0.0000172296 0.7639     4.61111     5.16667 0.99199
## ST_730 0.00001302910 0.0000230856 0.5644     2.44444     5.66667 0.99208
## ST_328 0.00001297399 0.0000093858 1.3823    31.11111    29.27778 0.99217
## ST_083 0.00001290785 0.0000102553 1.2587    12.61111    15.66667 0.99226
## ST_339 0.00001281966 0.0000099616 1.2869    11.94444    13.05556 0.99235
## ST_455 0.00001275353 0.0000115221 1.1069     3.05556     7.38889 0.99244
## ST_467 0.00001259921 0.0000197021 0.6395     3.44444     5.22222 0.99253
## ST_026 0.00001255511 0.0000103132 1.2174    11.16667     7.77778 0.99262
## ST_445 0.00001255511 0.0000167189 0.7510     4.38889     4.44444 0.99270
## ST_324 0.00001227954 0.0000121873 1.0076     6.00000     5.38889 0.99279
## ST_167 0.00001223545 0.0000090994 1.3446    21.72222    22.00000 0.99288
## ST_428 0.00001215829 0.0000090030 1.3505    16.94444    19.77778 0.99296
## ST_346 0.00001207011 0.0000086615 1.3935    21.38889    18.38889 0.99304
## ST_135 0.00001176146 0.0000089438 1.3150    24.50000    23.55556 0.99313
## ST_601 0.00001172840 0.0000086068 1.3627     9.66667     8.11111 0.99321
## ST_139 0.00001159612 0.0000232708 0.4983     3.55556     3.77778 0.99329
## ST_666 0.00001136464 0.0000167028 0.6804     9.50000     6.72222 0.99337
## ST_238 0.00001121032 0.0000082205 1.3637    13.77778    10.83333 0.99345
## ST_713 0.00001091270 0.0000099370 1.0982     8.05556     7.55556 0.99352
## ST_517 0.00001053792 0.0000082909 1.2710    16.16667    17.88889 0.99360
## ST_267 0.00001051587 0.0000109827 0.9575     3.83333     4.16667 0.99367
## ST_326 0.00001039462 0.0000077089 1.3484    14.77778    16.16667 0.99374
## ST_487 0.00001022928 0.0000074381 1.3752    15.66667    17.00000 0.99381
## ST_562 0.00001014109 0.0000164412 0.6168     0.66667     5.38889 0.99388
## ST_200 0.00000990961 0.0000083888 1.1813     6.61111     3.16667 0.99395
## ST_316 0.00000990961 0.0000068452 1.4477    11.66667     7.00000 0.99402
## ST_213 0.00000972222 0.0000074553 1.3041    13.33333    13.72222 0.99409
## ST_132 0.00000967813 0.0000165406 0.5851     4.16667     1.94444 0.99416
## ST_365 0.00000965608 0.0000243649 0.3963     2.50000     3.27778 0.99422
## ST_299 0.00000962302 0.0000111527 0.8628     0.66667     5.22222 0.99429
## ST_662 0.00000957892 0.0000227000 0.4220     1.66667     4.05556 0.99436
## ST_404 0.00000940256 0.0000121533 0.7737     2.55556     3.88889 0.99442
## ST_452 0.00000932540 0.0000108616 0.8586     3.88889     2.94444 0.99449
## ST_340 0.00000925926 0.0000074400 1.2445     7.55556    10.27778 0.99455
## ST_197 0.00000921517 0.0000092579 0.9954     3.16667     4.05556 0.99462
## ST_071 0.00000919312 0.0000083916 1.0955     3.77778     4.66667 0.99468
## ST_231 0.00000916005 0.0000065113 1.4068    12.44444    10.11111 0.99475
## ST_072 0.00000904982 0.0000077236 1.1717     2.33333     5.00000 0.99481
## ST_025 0.00000902778 0.0000071020 1.2712     5.22222     1.38889 0.99487
## ST_392 0.00000900573 0.0000069760 1.2910     4.16667     5.72222 0.99493
## ST_378 0.00000880732 0.0000080442 1.0949     5.22222     2.00000 0.99500
## ST_239 0.00000876323 0.0000134306 0.6525     3.05556     2.72222 0.99506
## ST_243 0.00000874118 0.0000106979 0.8171     0.33333     4.88889 0.99512
## ST_098 0.00000871914 0.0000066545 1.3103     4.94444     5.16667 0.99518
## ST_091 0.00000860891 0.0000064021 1.3447    14.00000    13.83333 0.99524
## ST_053 0.00000857584 0.0000250744 0.3420     2.27778     2.77778 0.99530
## ST_032 0.00000853175 0.0000069396 1.2294     7.11111     5.27778 0.99536
## ST_064 0.00000853175 0.0000083899 1.0169    10.16667    10.50000 0.99542
## ST_160 0.00000845459 0.0000074003 1.1425     1.83333     5.44444 0.99548
## ST_729 0.00000845459 0.0000059939 1.4105     9.38889     6.66667 0.99553
## ST_554 0.00000844356 0.0000066801 1.2640     6.22222     7.55556 0.99559
## ST_076 0.00000841049 0.0000087192 0.9646     5.22222     3.11111 0.99565
## ST_203 0.00000841049 0.0000064567 1.3026    10.05556     8.05556 0.99571
## ST_329 0.00000832231 0.0000059721 1.3935    12.16667    10.72222 0.99577
## ST_462 0.00000824515 0.0000085314 0.9664     3.33333     3.22222 0.99583
## ST_092 0.00000819004 0.0000059222 1.3829    12.00000    10.88889 0.99588
## ST_263 0.00000813492 0.0000060307 1.3489     9.55556    10.88889 0.99594
## ST_610 0.00000812390 0.0000065126 1.2474     5.27778     2.27778 0.99600
## ST_149 0.00000803571 0.0000056948 1.4111    11.50000     9.88889 0.99605
## ST_475 0.00000793651 0.0000059433 1.3354     4.11111     4.16667 0.99611
## ST_108 0.00000791446 0.0000058962 1.3423     6.22222     2.72222 0.99616
## ST_070 0.00000779321 0.0000049391 1.5779     3.38889     6.77778 0.99622
## ST_490 0.00000773810 0.0000068016 1.1377     3.83333     3.22222 0.99627
## ST_278 0.00000767196 0.0000069064 1.1108     0.11111     4.33333 0.99633
## ST_696 0.00000759480 0.0000179039 0.4242     3.38889     1.11111 0.99638
## ST_097 0.00000756173 0.0000058040 1.3029    10.27778     8.66667 0.99643
## ST_627 0.00000755071 0.0000085479 0.8833     2.33333     3.16667 0.99648
## ST_084 0.00000753968 0.0000048757 1.5464     9.66667     7.88889 0.99654
## ST_151 0.00000745150 0.0000057115 1.3046    10.50000    11.00000 0.99659
## ST_302 0.00000744048 0.0000050673 1.4683     4.77778     2.50000 0.99664
## ST_172 0.00000740741 0.0000098436 0.7525     1.33333     3.50000 0.99669
## ST_338 0.00000736332 0.0000133020 0.5535     2.11111     2.55556 0.99674
## ST_415 0.00000735229 0.0000055584 1.3227    13.50000    11.22222 0.99679
## ST_226 0.00000723104 0.0000054614 1.3240     4.83333     5.33333 0.99684
## ST_403 0.00000714286 0.0000061667 1.1583     2.66667     3.66667 0.99689
## ST_134 0.00000698854 0.0000053226 1.3130     7.55556     7.44444 0.99694
## ST_216 0.00000695547 0.0000057889 1.2015     3.72222     3.16667 0.99699
## ST_028 0.00000692240 0.0000052923 1.3080     6.72222     5.61111 0.99704
## ST_249 0.00000688933 0.0000050659 1.3600    11.05556     8.83333 0.99709
## ST_209 0.00000683422 0.0000053731 1.2719     9.77778     8.38889 0.99714
## ST_321 0.00000680115 0.0000050957 1.3347     5.55556     4.38889 0.99718
## ST_136 0.00000676808 0.0000050856 1.3308     6.11111     4.72222 0.99723
## ST_649 0.00000674603 0.0000051610 1.3071     6.27778     4.22222 0.99728
## ST_570 0.00000672399 0.0000077400 0.8687     2.77778     2.16667 0.99732
## ST_222 0.00000669092 0.0000049573 1.3497     8.16667     8.16667 0.99737
## ST_088 0.00000666887 0.0000056445 1.1815    10.66667    10.94444 0.99742
## ST_128 0.00000665785 0.0000055447 1.2008     3.44444     3.61111 0.99746
## ST_672 0.00000661376 0.0000058980 1.1214     3.77778     3.33333 0.99751
## ST_535 0.00000652557 0.0000048180 1.3544     6.00000     4.94444 0.99756
## ST_604 0.00000651455 0.0000052236 1.2471     0.88889     3.77778 0.99760
## ST_035 0.00000644841 0.0000172489 0.3738     2.33333     1.50000 0.99765
## ST_678 0.00000643739 0.0000058811 1.0946     4.94444     4.77778 0.99769
## ST_621 0.00000633818 0.0000208524 0.3040     3.33333     0.27778 0.99773
## ST_524 0.00000632716 0.0000043655 1.4494     7.38889     6.55556 0.99778
## ST_253 0.00000623898 0.0000054412 1.1466     8.66667     6.27778 0.99782
## ST_210 0.00000621693 0.0000044245 1.4051     6.83333     6.38889 0.99787
## ST_031 0.00000610670 0.0000062973 0.9697     2.83333     1.77778 0.99791
## ST_655 0.00000605159 0.0000051619 1.1723     2.11111     3.33333 0.99795
## ST_527 0.00000602954 0.0000044034 1.3693     3.77778     2.16667 0.99799
## ST_430 0.00000595238 0.0000054909 1.0840     1.72222     3.16667 0.99803
## ST_522 0.00000587522 0.0000042234 1.3911     3.77778     4.27778 0.99807
## ST_667 0.00000587522 0.0000043021 1.3657     4.83333     5.16667 0.99812
## ST_020 0.00000579806 0.0000060750 0.9544     3.27778     0.55556 0.99816
## ST_676 0.00000578704 0.0000114980 0.5033     1.16667     2.33333 0.99820
## ST_720 0.00000569885 0.0000077540 0.7350     1.83333     2.11111 0.99824
## ST_520 0.00000566578 0.0000045838 1.2360     6.77778     6.05556 0.99828
## ST_540 0.00000562169 0.0000111100 0.5060     1.66667     1.83333 0.99831
## ST_347 0.00000557760 0.0000122092 0.4568     0.55556     2.77778 0.99835
## ST_150 0.00000554453 0.0000041451 1.3376     7.77778     7.77778 0.99839
## ST_355 0.00000552249 0.0000054451 1.0142     1.16667     3.16667 0.99843
## ST_571 0.00000552249 0.0000097733 0.5651     1.55556     2.16667 0.99847
## ST_688 0.00000552249 0.0000052362 1.0547     2.22222     2.44444 0.99851
## ST_648 0.00000548942 0.0000083487 0.6575     1.94444     1.66667 0.99855
## ST_269 0.00000547840 0.0000044861 1.2212     2.55556     2.66667 0.99858
## ST_633 0.00000503748 0.0000064527 0.7807     0.77778     2.44444 0.99862
## ST_436 0.00000494929 0.0000099702 0.4964     1.38889     1.66667 0.99865
## ST_225 0.00000479497 0.0000041445 1.1570     2.44444     2.22222 0.99869
## ST_386 0.00000469577 0.0000062601 0.7501     1.83333     1.50000 0.99872
## ST_521 0.00000468474 0.0000037442 1.2512     3.05556     2.11111 0.99875
## ST_052 0.00000455247 0.0000091972 0.4950     1.33333     1.50000 0.99878
## ST_223 0.00000451940 0.0000067672 0.6678     1.50000     1.50000 0.99882
## ST_389 0.00000450838 0.0000068268 0.6604     1.05556     2.00000 0.99885
## ST_625 0.00000449735 0.0000043931 1.0237     1.61111     2.05556 0.99888
## ST_545 0.00000439815 0.0000095647 0.4598     1.00000     1.66667 0.99891
## ST_387 0.00000436508 0.0000045031 0.9694     2.16667     1.22222 0.99894
## ST_433 0.00000426587 0.0000047308 0.9017     1.66667     1.55556 0.99897
## ST_286 0.00000422178 0.0000033208 1.2713     2.77778     2.33333 0.99900
## ST_284 0.00000412257 0.0000068242 0.6041     1.33333     1.33333 0.99903
## ST_714 0.00000408951 0.0000069144 0.5914     0.38889     2.11111 0.99906
## ST_357 0.00000407848 0.0000040530 1.0063     2.33333     0.72222 0.99908
## ST_429 0.00000407848 0.0000091432 0.4461     1.72222     0.83333 0.99911
## ST_579 0.00000406746 0.0000040713 0.9991     1.33333     1.77778 0.99914
## ST_664 0.00000397928 0.0000062281 0.6389     1.27778     1.33333 0.99917
## ST_168 0.00000386905 0.0000056338 0.6868     0.22222     2.11111 0.99920
## ST_309 0.00000381393 0.0000078601 0.4852     0.88889     1.44444 0.99922
## ST_259 0.00000373677 0.0000041840 0.8931     1.11111     1.66667 0.99925
## ST_308 0.00000373677 0.0000072357 0.5164     1.05556     1.27778 0.99927
## ST_709 0.00000362654 0.0000062159 0.5834     0.44444     1.83333 0.99930
## ST_304 0.00000354938 0.0000061426 0.5778     0.44444     1.72222 0.99932
## ST_207 0.00000350529 0.0000071770 0.4884     1.77778     0.27778 0.99935
## ST_082 0.00000343915 0.0000060505 0.5684     0.61111     1.55556 0.99937
## ST_715 0.00000341711 0.0000035207 0.9706     1.16667     1.55556 0.99940
## ST_303 0.00000338404 0.0000036513 0.9268     1.38889     1.16667 0.99942
## ST_663 0.00000321869 0.0000057726 0.5576     0.66667     1.33333 0.99944
## ST_551 0.00000316358 0.0000079196 0.3995     1.55556     0.27778 0.99947
## ST_065 0.00000300926 0.0000040457 0.7438     1.11111     0.94444 0.99949
## ST_692 0.00000297619 0.0000044279 0.6721     0.16667     1.61111 0.99951
## ST_145 0.00000286596 0.0000031537 0.9088     1.16667     0.94444 0.99953
## ST_143 0.00000284392 0.0000057699 0.4929     0.16667     1.50000 0.99955
## ST_566 0.00000282187 0.0000049049 0.5753     0.33333     1.38889 0.99957
## ST_697 0.00000279982 0.0000044717 0.6261     0.94444     0.88889 0.99959
## ST_658 0.00000268959 0.0000032698 0.8226     0.50000     1.27778 0.99960
## ST_368 0.00000256834 0.0000028725 0.8941     0.61111     1.33333 0.99962
## ST_600 0.00000246914 0.0000036891 0.6693     1.00000     0.61111 0.99964
## ST_596 0.00000227072 0.0000030367 0.7478     0.61111     1.00000 0.99966
## ST_541 0.00000225970 0.0000049212 0.4592     0.22222     1.11111 0.99967
## ST_388 0.00000221561 0.0000037773 0.5866     1.00000     0.44444 0.99969
## ST_668 0.00000217152 0.0000053644 0.4048     0.22222     1.05556 0.99970
## ST_447 0.00000214947 0.0000043815 0.4906     0.61111     0.72222 0.99972
## ST_718 0.00000209436 0.0000033138 0.6320     1.05556     0.22222 0.99973
## ST_463 0.00000206129 0.0000035189 0.5858     0.83333     0.44444 0.99975
## ST_717 0.00000202822 0.0000046583 0.4354     0.33333     0.88889 0.99976
## ST_218 0.00000201720 0.0000045125 0.4470     0.44444     0.77778 0.99977
## ST_059 0.00000196208 0.0000021376 0.9179     0.61111     0.88889 0.99979
## ST_503 0.00000187390 0.0000025238 0.7425     0.83333     0.44444 0.99980
## ST_543 0.00000181878 0.0000033399 0.5446     0.72222     0.44444 0.99981
## ST_287 0.00000170855 0.0000021709 0.7870     0.83333     0.33333 0.99983
## ST_423 0.00000167549 0.0000040272 0.4160     0.44444     0.55556 0.99984
## ST_277 0.00000153219 0.0000029533 0.5188     0.16667     0.77778 0.99985
## ST_313 0.00000153219 0.0000047438 0.3230     0.61111     0.27778 0.99986
## ST_019 0.00000146605 0.0000026505 0.5531     0.72222     0.16667 0.99987
## ST_726 0.00000135582 0.0000029861 0.4540     0.05556     0.72222 0.99988
## ST_513 0.00000123457 0.0000036738 0.3360     0.27778     0.44444 0.99989
## ST_661 0.00000123457 0.0000036522 0.3380     0.16667     0.55556 0.99990
## ST_138 0.00000122354 0.0000030168 0.4056     0.16667     0.55556 0.99990
## ST_291 0.00000119048 0.0000019622 0.6067     0.55556     0.16667 0.99991
## ST_659 0.00000119048 0.0000021904 0.5435     0.22222     0.50000 0.99992
## ST_189 0.00000114638 0.0000030503 0.3758     0.11111     0.55556 0.99993
## ST_418 0.00000113536 0.0000031216 0.3637     0.38889     0.27778 0.99994
## ST_385 0.00000111332 0.0000026476 0.4205     0.22222     0.44444 0.99994
## ST_317 0.00000105820 0.0000025141 0.4209     0.05556     0.55556 0.99995
## ST_449 0.00000089286 0.0000018240 0.4895     0.27778     0.27778 0.99996
## ST_716 0.00000076058 0.0000022728 0.3347     0.27778     0.16667 0.99996
## ST_582 0.00000069444 0.0000018022 0.3853     0.38889     0.00000 0.99997
## ST_159 0.00000066138 0.0000010208 0.6479     0.27778     0.16667 0.99997
## ST_220 0.00000065035 0.0000015361 0.4234     0.22222     0.16667 0.99998
## ST_472 0.00000059524 0.0000016862 0.3530     0.33333     0.00000 0.99998
## ST_379 0.00000056217 0.0000014118 0.3982     0.16667     0.16667 0.99999
## ST_161 0.00000049603 0.0000014364 0.3453     0.27778     0.00000 0.99999
## ST_553 0.00000039683 0.0000009530 0.4164     0.00000     0.22222 0.99999
## ST_062 0.00000037478 0.0000010962 0.3419     0.11111     0.11111 0.99999
## ST_348 0.00000036376 0.0000007203 0.5050     0.16667     0.05556 1.00000
## ST_364 0.00000019841 0.0000008193 0.2422     0.00000     0.11111 1.00000
## ST_670 0.00000018739 0.0000005481 0.3419     0.05556     0.05556 1.00000
## ST_181 0.00000009921 0.0000004097 0.2422     0.00000     0.05556 1.00000
##               p
## ST_561 0.297030
## ST_343 0.009901 **
## ST_290 0.049505 *
## ST_201 0.980198
## ST_337 0.019802 *
## ST_500 0.029703 *
## ST_099 0.168317
## ST_100 0.009901 **
## ST_614 0.455446
## ST_120 0.415842
## ST_336 0.089109 .
## ST_296 0.019802 *
## ST_250 0.009901 **
## ST_248 0.009901 **
## ST_115 0.415842
## ST_153 0.653465
## ST_295 0.009901 **
## ST_061 0.851485
## ST_331 0.445545
## ST_244 0.138614
## ST_534 0.009901 **
## ST_550 0.009901 **
## ST_137 0.782178
## ST_557 0.554455
## ST_119 0.980198
## ST_130 0.336634
## ST_144 0.524752
## ST_045 0.871287
## ST_612 0.009901 **
## ST_536 0.009901 **
## ST_438 0.792079
## ST_043 0.118812
## ST_094 0.009901 **
## ST_687 0.009901 **
## ST_152 0.920792
## ST_246 0.792079
## ST_525 0.782178
## ST_081 0.029703 *
## ST_190 0.871287
## ST_731 0.009901 **
## ST_330 0.009901 **
## ST_297 0.009901 **
## ST_093 0.029703 *
## ST_615 0.673267
## ST_236 0.019802 *
## ST_518 0.009901 **
## ST_680 0.841584
## ST_033 0.009901 **
## ST_176 0.049505 *
## ST_539 0.009901 **
## ST_262 0.009901 **
## ST_595 0.009901 **
## ST_469 0.881188
## ST_117 0.069307 .
## ST_090 0.514851
## ST_169 0.009901 **
## ST_613 0.306931
## ST_096 0.613861
## ST_677 0.128713
## ST_104 0.613861
## ST_593 0.009901 **
## ST_556 0.009901 **
## ST_261 0.009901 **
## ST_626 0.910891
## ST_335 0.009901 **
## ST_047 0.900990
## ST_528 0.009901 **
## ST_533 0.544554
## ST_147 0.029703 *
## ST_410 0.009901 **
## ST_086 0.198020
## ST_039 0.425743
## ST_493 0.079208 .
## ST_322 0.594059
## ST_109 0.316832
## ST_192 0.396040
## ST_208 0.079208 .
## ST_046 0.584158
## ST_569 0.009901 **
## ST_325 0.960396
## ST_501 0.019802 *
## ST_085 0.227723
## ST_049 0.217822
## ST_212 0.019802 *
## ST_121 0.376238
## ST_221 0.009901 **
## ST_044 0.217822
## ST_202 0.762376
## ST_311 0.643564
## ST_594 0.336634
## ST_383 0.841584
## ST_559 0.504950
## ST_196 0.009901 **
## ST_607 0.009901 **
## ST_306 0.613861
## ST_441 0.029703 *
## ST_118 0.029703 *
## ST_127 0.039604 *
## ST_224 0.009901 **
## ST_101 0.079208 .
## ST_042 0.039604 *
## ST_390 0.108911
## ST_141 0.158416
## ST_146 0.089109 .
## ST_675 0.099010 .
## ST_298 0.297030
## ST_671 0.128713
## ST_318 0.524752
## ST_411 0.009901 **
## ST_529 0.475248
## ST_029 0.821782
## ST_611 0.089109 .
## ST_327 0.881188
## ST_681 0.108911
## ST_653 0.366337
## ST_195 0.128713
## ST_320 0.009901 **
## ST_048 1.000000
## ST_107 0.049505 *
## ST_563 0.673267
## ST_442 0.009901 **
## ST_305 0.514851
## ST_319 0.524752
## ST_567 0.089109 .
## ST_366 0.237624
## ST_459 0.821782
## ST_542 0.346535
## ST_427 0.742574
## ST_235 0.564356
## ST_228 0.574257
## ST_558 0.099010 .
## ST_116 0.019802 *
## ST_460 0.643564
## ST_546 0.603960
## ST_707 0.029703 *
## ST_063 0.554455
## ST_602 0.237624
## ST_605 0.009901 **
## ST_247 0.900990
## ST_166 0.643564
## ST_206 0.009901 **
## ST_004 0.178218
## ST_124 1.000000
## ST_434 0.118812
## ST_204 0.782178
## ST_580 0.287129
## ST_341 0.782178
## ST_710 0.712871
## ST_359 1.000000
## ST_102 0.079208 .
## ST_232 0.079208 .
## ST_728 0.059406 .
## ST_215 0.009901 **
## ST_426 0.712871
## ST_560 0.257426
## ST_105 0.079208 .
## ST_095 0.019802 *
## ST_624 0.009901 **
## ST_233 0.792079
## ST_690 0.237624
## ST_398 0.811881
## ST_170 0.316832
## ST_608 0.009901 **
## ST_437 0.009901 **
## ST_402 0.089109 .
## ST_419 0.792079
## ST_178 0.207921
## ST_214 0.495050
## ST_050 0.247525
## ST_023 0.801980
## ST_555 0.306931
## ST_425 0.584158
## ST_638 0.702970
## ST_512 0.841584
## ST_565 0.336634
## ST_301 0.059406 .
## ST_027 0.316832
## ST_310 0.752475
## ST_219 0.742574
## ST_332 0.019802 *
## ST_623 0.673267
## ST_227 0.039604 *
## ST_312 0.237624
## ST_609 0.009901 **
## ST_087 0.633663
## ST_708 0.396040
## ST_384 0.950495
## ST_443 0.920792
## ST_650 0.920792
## ST_630 0.039604 *
## ST_103 0.326733
## ST_315 0.059406 .
## ST_140 0.465347
## ST_264 0.019802 *
## ST_089 0.306931
## ST_242 0.544554
## ST_171 0.089109 .
## ST_131 0.059406 .
## ST_106 0.297030
## ST_606 0.148515
## ST_440 0.346535
## ST_439 0.128713
## ST_256 0.584158
## ST_252 0.128713
## ST_268 0.059406 .
## ST_453 0.554455
## ST_431 0.049505 *
## ST_432 1.000000
## ST_241 0.158416
## ST_391 0.207921
## ST_444 0.881188
## ST_730 0.435644
## ST_328 0.603960
## ST_083 0.316832
## ST_339 0.643564
## ST_455 0.059406 .
## ST_467 0.990099
## ST_026 0.039604 *
## ST_445 0.742574
## ST_324 0.891089
## ST_167 0.970297
## ST_428 0.178218
## ST_346 0.079208 .
## ST_135 0.613861
## ST_601 0.693069
## ST_139 0.851485
## ST_666 0.801980
## ST_238 0.118812
## ST_713 0.702970
## ST_517 0.475248
## ST_267 0.316832
## ST_326 0.207921
## ST_487 0.306931
## ST_562 0.049505 *
## ST_200 0.069307 .
## ST_316 0.019802 *
## ST_213 0.504950
## ST_132 0.643564
## ST_365 1.000000
## ST_299 0.019802 *
## ST_662 1.000000
## ST_404 0.495050
## ST_452 0.633663
## ST_340 0.168317
## ST_197 0.772277
## ST_071 0.811881
## ST_231 0.108911
## ST_072 0.099010 .
## ST_025 0.009901 **
## ST_392 0.445545
## ST_378 0.029703 *
## ST_239 1.000000
## ST_243 0.009901 **
## ST_098 0.881188
## ST_091 0.663366
## ST_053 1.000000
## ST_032 0.297030
## ST_064 0.623762
## ST_160 0.009901 **
## ST_729 0.099010 .
## ST_554 0.534653
## ST_076 0.227723
## ST_203 0.297030
## ST_329 0.356436
## ST_462 0.940594
## ST_092 0.297030
## ST_263 0.653465
## ST_610 0.049505 *
## ST_149 0.178218
## ST_475 0.603960
## ST_108 0.009901 **
## ST_070 0.009901 **
## ST_490 0.247525
## ST_278 0.009901 **
## ST_696 0.504950
## ST_097 0.306931
## ST_627 0.554455
## ST_084 0.019802 *
## ST_151 0.950495
## ST_302 0.019802 *
## ST_172 0.227723
## ST_338 1.000000
## ST_415 0.059406 .
## ST_226 0.891089
## ST_403 0.376238
## ST_134 0.990099
## ST_216 0.663366
## ST_028 0.465347
## ST_249 0.099010 .
## ST_209 0.485149
## ST_321 0.554455
## ST_136 0.198020
## ST_649 0.108911
## ST_570 0.900990
## ST_222 0.693069
## ST_088 0.752475
## ST_128 1.000000
## ST_672 0.970297
## ST_535 0.316832
## ST_604 0.009901 **
## ST_035 1.000000
## ST_678 0.990099
## ST_621 0.475248
## ST_524 0.356436
## ST_253 0.039604 *
## ST_210 0.336634
## ST_031 0.207921
## ST_655 0.277228
## ST_527 0.099010 .
## ST_430 0.277228
## ST_522 0.702970
## ST_667 0.980198
## ST_020 0.019802 *
## ST_676 0.435644
## ST_720 1.000000
## ST_520 0.900990
## ST_540 1.000000
## ST_347 0.247525
## ST_150 0.950495
## ST_355 0.049505 *
## ST_571 0.960396
## ST_688 0.960396
## ST_648 0.148515
## ST_269 0.663366
## ST_633 0.059406 .
## ST_436 1.000000
## ST_225 0.980198
## ST_386 0.821782
## ST_521 0.574257
## ST_052 1.000000
## ST_223 1.000000
## ST_389 0.980198
## ST_625 0.871287
## ST_545 1.000000
## ST_387 0.396040
## ST_433 0.980198
## ST_286 0.722772
## ST_284 1.000000
## ST_714 0.138614
## ST_357 0.029703 *
## ST_429 1.000000
## ST_579 0.415842
## ST_664 1.000000
## ST_168 0.029703 *
## ST_309 0.712871
## ST_259 0.772277
## ST_308 1.000000
## ST_709 0.198020
## ST_304 0.346535
## ST_207 0.277228
## ST_082 0.574257
## ST_715 1.000000
## ST_303 0.475248
## ST_663 0.514851
## ST_551 0.495050
## ST_065 0.712871
## ST_692 0.019802 *
## ST_145 0.475248
## ST_143 0.148515
## ST_566 0.376238
## ST_697 1.000000
## ST_658 0.099010 .
## ST_368 0.336634
## ST_600 0.574257
## ST_596 0.970297
## ST_541 0.366337
## ST_388 0.702970
## ST_668 0.336634
## ST_447 1.000000
## ST_718 0.188119
## ST_463 0.386139
## ST_717 1.000000
## ST_218 1.000000
## ST_059 0.900990
## ST_503 0.594059
## ST_543 1.000000
## ST_287 0.316832
## ST_423 1.000000
## ST_277 0.277228
## ST_313 1.000000
## ST_019 0.386139
## ST_726 0.247525
## ST_513 1.000000
## ST_661 1.000000
## ST_138 1.000000
## ST_291 0.168317
## ST_659 0.376238
## ST_189 1.000000
## ST_418 1.000000
## ST_385 1.000000
## ST_317 0.465347
## ST_449 1.000000
## ST_716 1.000000
## ST_582 0.188119
## ST_159 1.000000
## ST_220 1.000000
## ST_472 0.514851
## ST_379 1.000000
## ST_161 0.475248
## ST_553 0.267327
## ST_062 1.000000
## ST_348 1.000000
## ST_364 1.000000
## ST_670 1.000000
## ST_181 1.000000
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## Permutation: free
## Number of permutations: 100
```