Diversidades alfa y beta para Fusarium y Actinobacteria

Camila Silva

2025-01-20

DIVERSIDADES ALFA Y BETA FUSARIUM

```
library("phyloseq")
library("ggplot2")
library("vegan")
```

```
## Loading required package: permute
```

```
## Loading required package: lattice
```

```
## This is vegan 2.6-8
```

```
#library("BiodiversityR")
library("RColorBrewer")
library("stringi")
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library("plyr")
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------------
```

```
##
## Attaching package: 'plyr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
setwd("/home/camila/GIT/Tesis_Maestria/Data/fresa_solena/Data1")
outpath = "/home/camila/GIT/Tesis_Maestria/Analisis_Comparativo/Fresa_Solena/Results_img"
```

```
### Cargado de datos originales
fresa_kraken <- import_biom("fresa_kraken.biom")
colnames(fresa_kraken@tax_table@.Data) <- c("Kingdom", "Phylum", "Class", "Order", "Family", "Genus", "S
fresa_kraken@tax_table@.Data <- substr(fresa_kraken@tax_table@.Data,4,100)
colnames(fresa_kraken@otu_table@.Data) <- substr(colnames(fresa_kraken@otu_table@.Data),1,6)
metadata_fresa <- read.csv2("/home/camila/GIT/Tesis_Maestria/Data/fresa_solena/Data1/metadata.csv",head
fresa_kraken@sam_data <- sample_data(metadata_fresa)
fresa_kraken@sam_data$Sample<-row.names(fresa_kraken@sam_data)
colnames(fresa_kraken@sam_data)<-c('Treatment','Samples')
samples_to_remove <- c("MP2079","MP2080","MP2088","MP2109","MP2137")
fresa_kraken_fil <- prune_samples(!(sample_names(fresa_kraken) %in% samples_to_remove), fresa_kraken)
percentages_fil <- transform_sample_counts(fresa_kraken_fil, function(x) x*100 / sum(x) )
percentages_df <- psmelt(percentages_fil)
```

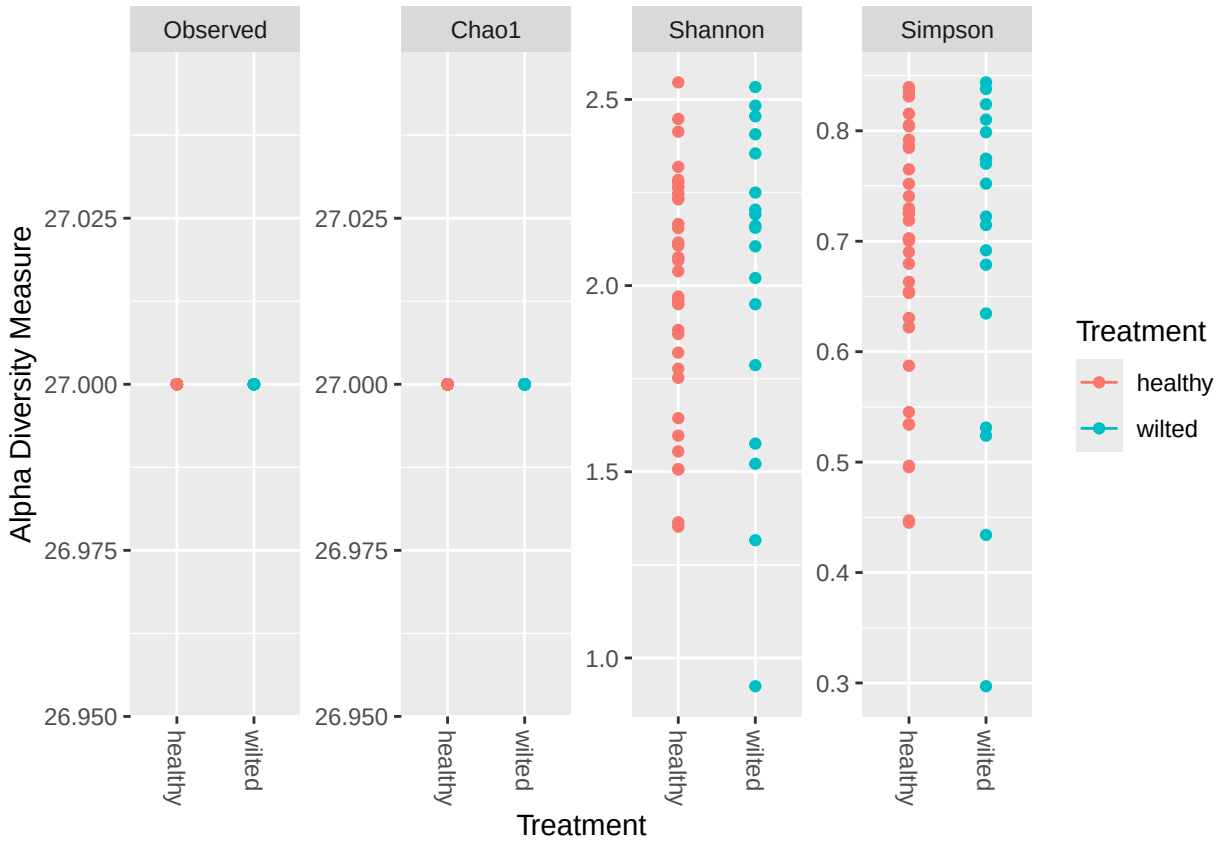
```
## Subconjunto de "Eukaryota"
merge_Eukaryota<-subset_taxa(fresa_kraken_fil,Kingdom=="Eukaryota")
## Subconjunto de "Bacteria"
merge_Bacteria<-subset_taxa(fresa_kraken_fil,Kingdom=="Bacteria")
```

```
## Aglomeramos Fusarium
#glom <- tax_glom(merge_Eukaryota, taxrank = 'Genus')
glom_Fusarium <- subset_taxa(merge_Eukaryota, Genus == 'Fusarium')

## se calcula diversidad alfa con el glom de fusarium
plot_alpha_Fusarium <- plot_richness(physeq = glom_Fusarium, measures = c("Observed","Chao1","Shannon",
```

```
## Warning in estimate_richness(physeq, split = TRUE, measures = measures): The data you have provided c
## any singletons. This is highly suspicious. Results of richness
## estimates (for example) are probably unreliable, or wrong, if you have already
## trimmed low-abundance taxa from the data.
##
## We recommended that you find the un-trimmed data and retry.
```

```
plot_alpha_Fusarium
```

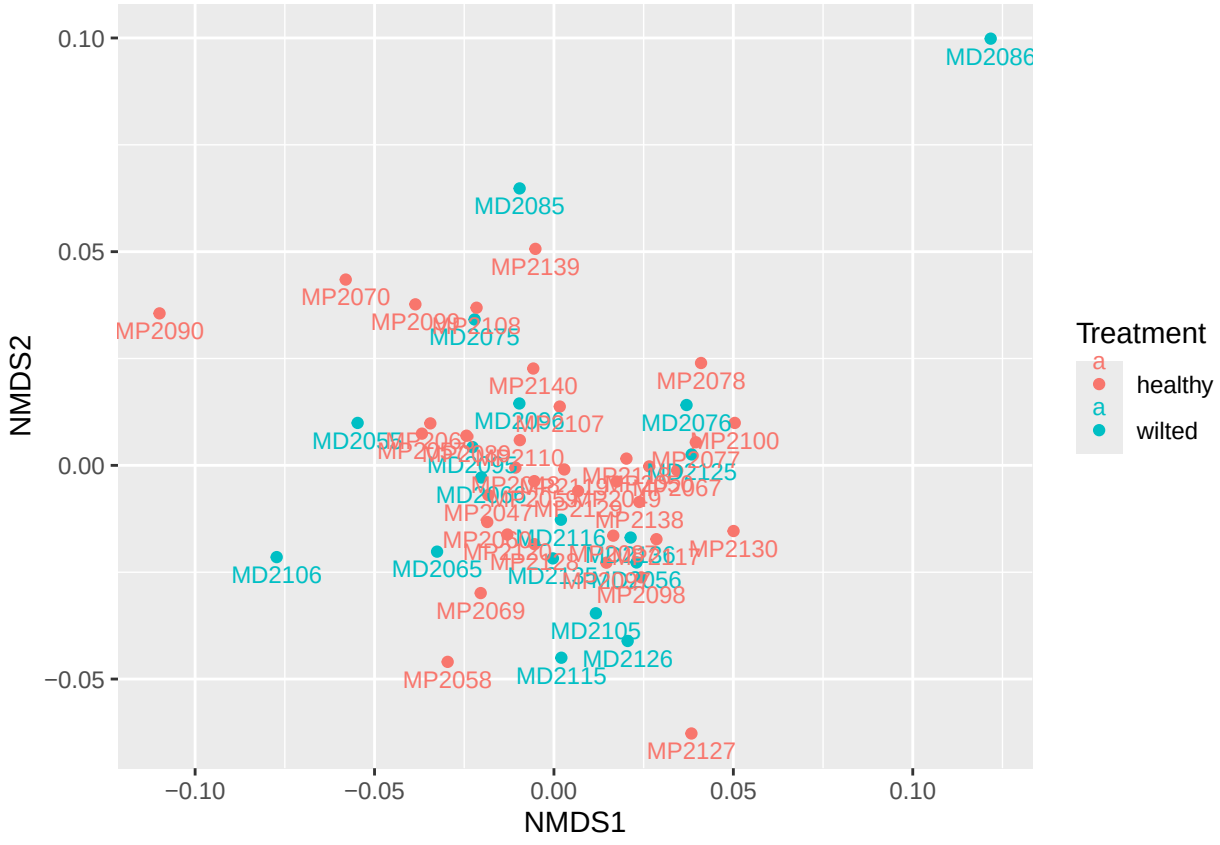



```
## se calcula la beta diversidad para Fusarium
## sacamos los porcentajes
percentages_Fusarium <- transform_sample_counts(glom_Fusarium, function(x) x*100 / sum(x) )
percentages_Fusarium_df <- psmelt(percentages_Fusarium)
meta_ord_Fusarium <- ordinate(physeq = percentages_Fusarium, method = "NMDS", distance = 'bray')
```

```
## Square root transformation
## Wisconsin double standardization
## Run 0 stress 0.1365369
## Run 1 stress 0.1365365
## ... New best solution
## ... Procrustes: rmse 0.001332495  max resid 0.007494057
## ... Similar to previous best
## Run 2 stress 0.1596839
## Run 3 stress 0.1502047
## Run 4 stress 0.1365368
## ... Procrustes: rmse 9.700695e-05  max resid 0.0006081054
## ... Similar to previous best
## Run 5 stress 0.1365366
## ... Procrustes: rmse 0.0004112145  max resid 0.002579067
## ... Similar to previous best
## Run 6 stress 0.1502046
## Run 7 stress 0.1365365
## ... New best solution
## ... Procrustes: rmse 0.0003067185  max resid 0.001923182
## ... Similar to previous best
## Run 8 stress 0.1381407
```

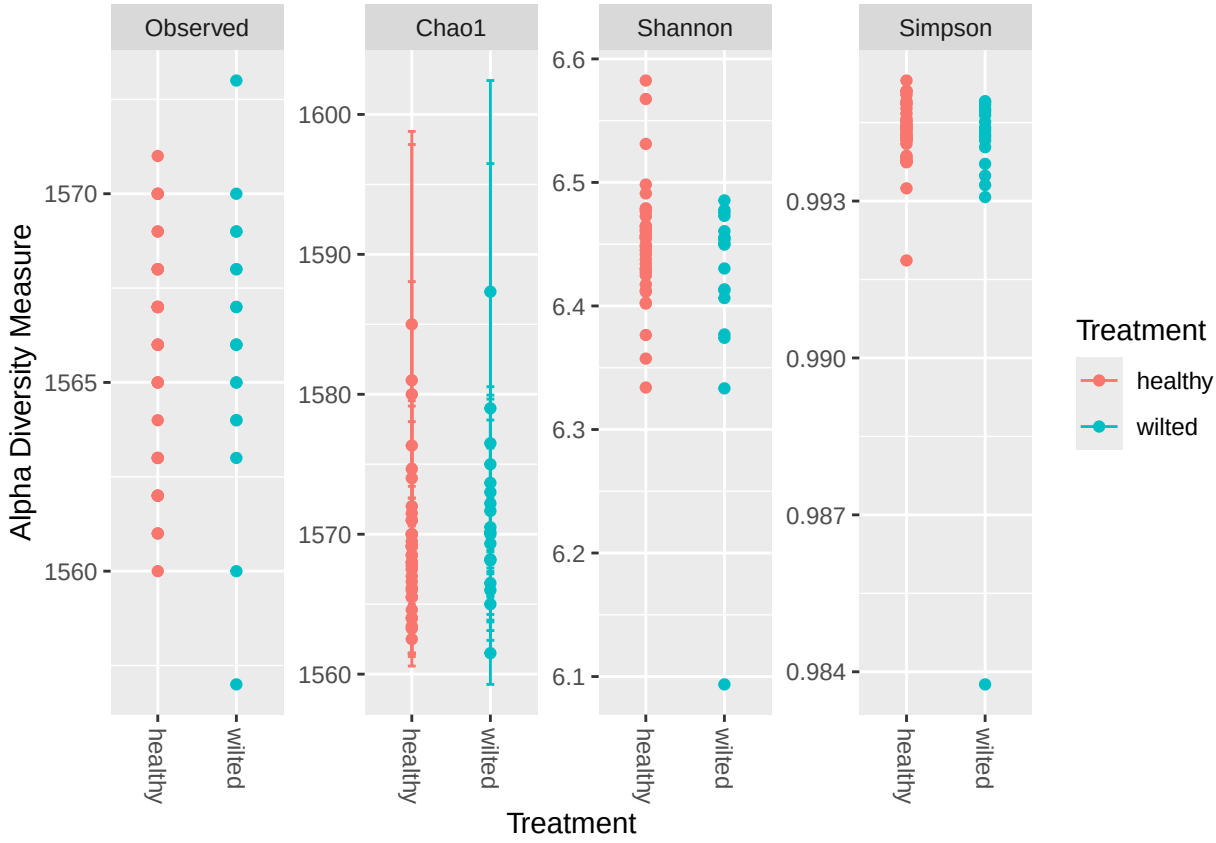
```
## Run 9 stress 0.1365365
## ... Procrustes: rmse 0.0002843466  max resid 0.001782476
## ... Similar to previous best
## Run 10 stress 0.1598128
## Run 11 stress 0.1365365
## ... Procrustes: rmse 7.632081e-06  max resid 4.39561e-05
## ... Similar to previous best
## Run 12 stress 0.1384263
## Run 13 stress 0.1381411
## Run 14 stress 0.1365364
## ... New best solution
## ... Procrustes: rmse 2.706862e-05  max resid 0.0001662397
## ... Similar to previous best
## Run 15 stress 0.138141
## Run 16 stress 0.1365364
## ... Procrustes: rmse 0.0002038817  max resid 0.001282127
## ... Similar to previous best
## Run 17 stress 0.1502046
## Run 18 stress 0.1622886
## Run 19 stress 0.1583245
## Run 20 stress 0.1382159
## *** Best solution repeated 2 times
```

```
plot_beta_Fusarium <- plot_ordination(physeq = percentages_Fusarium, ordination = meta_ord_Fusarium, col
  geom_text(mapping = aes(label = colnames(glom_Fusarium@otu_table@.Data)), size = 3, vjust = 1.5)
plot_beta_Fusarium
```



```
## Aglomeramos Actinobacteria
#Actinobacteria
#glom <- tax_glom(merge_Bacteria, taxrank = 'Phylum')
glom_Actinobacteria <- subset_taxa(merge_Bacteria, Phylum == 'Actinobacteria')

## se calcula diversidad alfa con el glom de Actinobacteria
plot_alpha_Actinobacteria <- plot_richness(physeq = glom_Actinobacteria, measures = c("Observed","Chao1
plot_alpha_Actinobacteria
```




```
## se calcula la beta diversidad para Actinobacteria
## sacamos los porcentajes
percentages_Actinobacteria <- transform_sample_counts(glom_Actinobacteria, function(x) x*100 / sum(x) )
percentages_Actinobacteria_df <- psmelt(percentages_Actinobacteria)
meta_ord_Actinobacteria <- ordinate(physeq = percentages_Actinobacteria, method = "NMDS", distance = 'bi
```

```
## Wisconsin double standardization
## Run 0 stress 0.1297898
## Run 1 stress 0.1292153
## ... New best solution
## ... Procrustes: rmse 0.1085984  max resid 0.6415185
## Run 2 stress 0.1292159
## ... Procrustes: rmse 0.0003119886  max resid 0.001913957
## ... Similar to previous best
## Run 3 stress 0.1345139
## Run 4 stress 0.127012
## ... New best solution
## ... Procrustes: rmse 0.1010332  max resid 0.6637212
```

```
## Run 5 stress 0.1345144
## Run 6 stress 0.149788
## Run 7 stress 0.1292159
## Run 8 stress 0.1292159
## Run 9 stress 0.1279187
## Run 10 stress 0.1345739
## Run 11 stress 0.1345144
## Run 12 stress 0.1292156
## Run 13 stress 0.1279188
## Run 14 stress 0.1727791
## Run 15 stress 0.1300133
## Run 16 stress 0.1699329
## Run 17 stress 0.1270122
## ... Procrustes: rmse 0.0001121115  max resid 0.0005309925
## ... Similar to previous best
## Run 18 stress 0.1292156
## Run 19 stress 0.1345139
## Run 20 stress 0.131241
## *** Best solution repeated 1 times
```

```
plot_beta_Actinobacteria <- plot_ordination(physeq = percentages_Actinobacteria, ordination = meta_ord_
  geom_text(mapping = aes(label = colnames(glom_Actinobacteria@otu_table@.Data)), size = 3, vjust = 1.5)
plot_beta_Actinobacteria
```