

Diversidades alfa y beta en datos metagenomicos de shotgun de fresa

Camila Silva

2023-04-03

AHORA QUEREMOS VER DIFERENCIAS ENTRE DIFERENTES NIVELES TAXONOMICOS

```
library("phyloseq")
library("ggplot2")
library("igraph")

##
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':
## 
##     decompose, spectrum

## The following object is masked from 'package:base':
## 
##     union

library("readr")
library("patchwork")
library("vegan")

## Loading required package: permute

##
## Attaching package: 'permute'

## The following object is masked from 'package:igraph':
## 
##     permute

## Loading required package: lattice

## This is vegan 2.6-4
```

```

## 
## Attaching package: 'vegan'

## The following object is masked from 'package:igraph':
## 
##     diversity

library("GUniFrac")
library("pbkrtest")

## Loading required package: lme4

## Loading required package: Matrix

#library("BiodiversityR")
library("kableExtra")

## Registered S3 method overwritten by 'httr':
##   method      from
##   print.response rmutil

library("RColorBrewer")

```

Cargado de datos originales

```

setwd("/home/camila/GIT/Tesis_Maestria/Data/fresa_solenia/Data1")
outpath = "/home/camila/GIT/Tesis_Maestria/Analisis_Comparativo/Fresa_Solenia/Results_img"

fresa_kraken <- import_biom("fresa_kraken.biom")
colnames(fresa_kraken@tax_table@.Data) <- c("Kingdom", "Phylum", "Class", "Order", "Family", "Genus", "Species")
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_solenia/Data1/metadata.csv", header=1)
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)

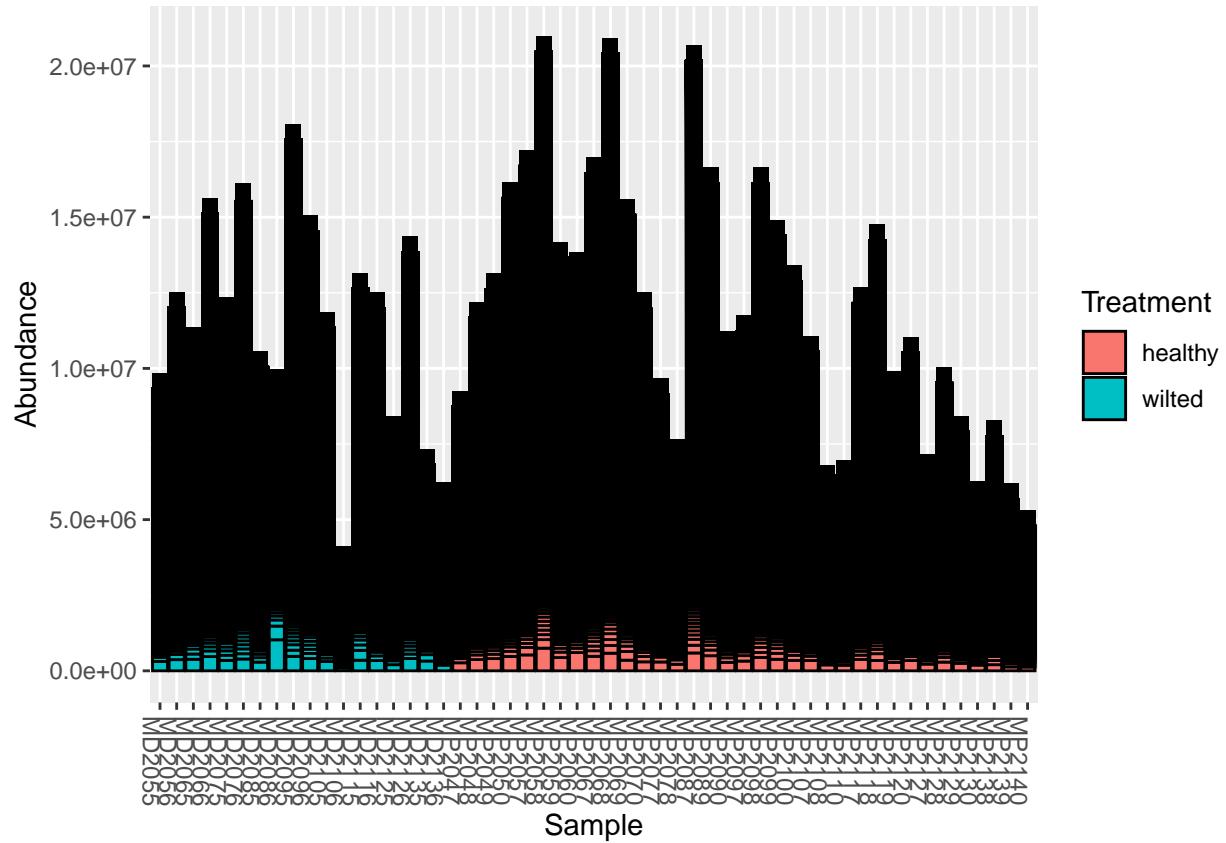
```

De aqui en adelante solo trabajaremos con los datos filtrados,

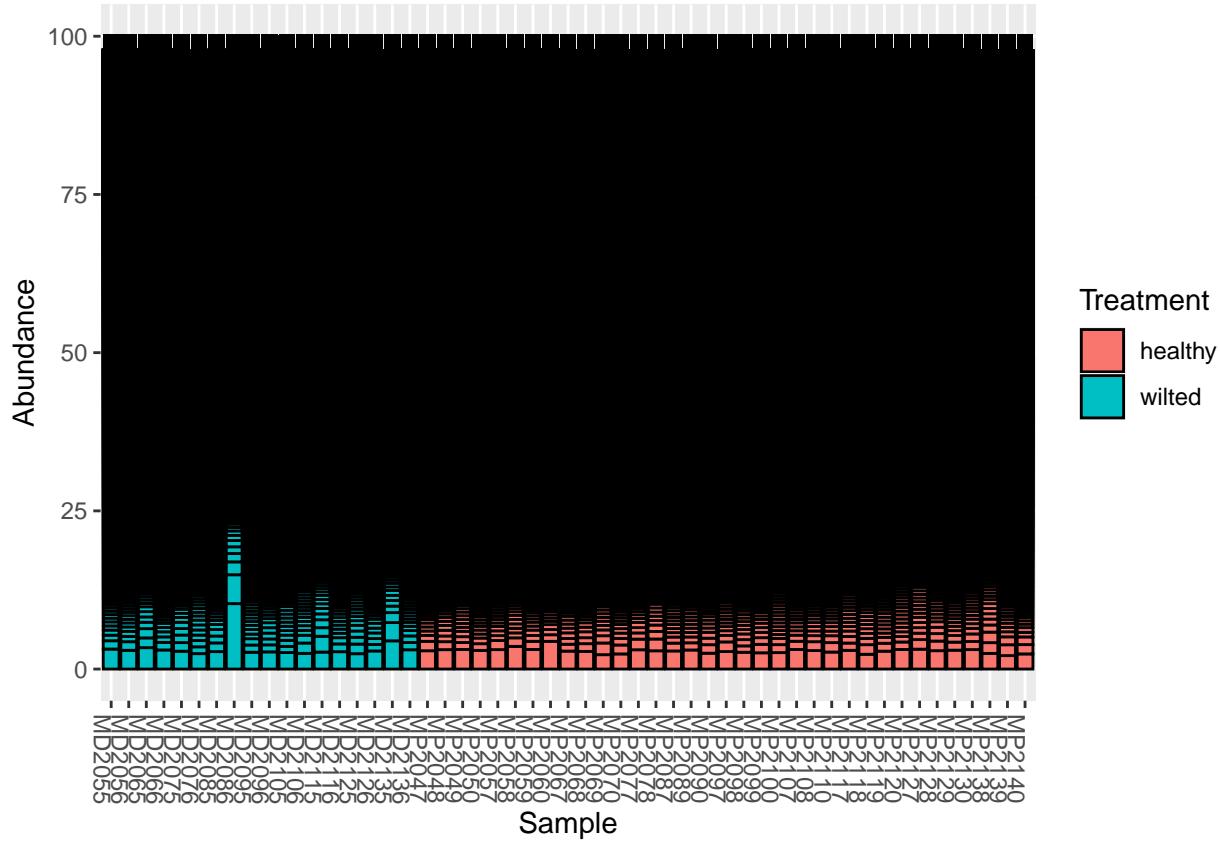
Grafico de barras de abundancia

Trazando las muestras en el eje x y las abundancias en el eje y Los valores de abundancia para cada OTUen cada muestra se apilan en el orden de mayor a menor, separados por una fina linea horizontal.

```
plot_bar(fresa_kraken_fil,fill="Treatment")
```



```
plot_bar(percentages_fil,fill="Treatment")
```



Queremos explorar nuestras muestras a diferentes niveles taxonómicos específicos
 ## A nivel de Kingdom

```
percentages_glom_kingdom <- tax_glom(percentages_fil, taxrank = 'Kingdom')
##View(percentages_glom_kingdom@tax_table@.Data)
```

A nivel de Phylum

```
percentages_glom_phylum <- tax_glom(percentages_fil, taxrank = 'Phylum')
##View(percentages_glom_phylum@tax_table@.Data)
```

Creamos un dataframe con los porcentajes de phylum

```
percentages_df_phylum <- psmelt(percentages_glom_phylum)
str(percentages_df_phylum)
```

```
## 'data.frame': 2279 obs. of 7 variables:
## $ OTU      : chr "1883" "1883" "1883" "1883" ...
## $ Sample   : chr "MP2060" "MP2058" "MP2059" "MP2049" ...
## $ Abundance: num 56.3 52.7 52.7 52.7 52.4 ...
## $ Treatment: chr "healthy" "healthy" "healthy" "healthy" ...
## $ Samples  : chr "MP2060" "MP2058" "MP2059" "MP2049" ...
## $ Kingdom  : chr "Bacteria" "Bacteria" "Bacteria" "Bacteria" ...
## $ Phylum   : chr "Actinobacteria" "Actinobacteria" "Actinobacteria" "Actinobacteria" ...
```

```

absolute_grom_phylum <- tax_grom(physeq = fresa_kraken_fil, taxrank = "Phylum")
absolute_df_phylum <- psmelt(absolute_grom_phylum)
str(absolute_df_phylum)

## 'data.frame': 2279 obs. of 7 variables:
## $ OTU      : chr "1883" "1883" "1883" "374" ...
## $ Sample   : chr "MP2058" "MP2087" "MP2068" "MP2068" ...
## $ Abundance: num 11028897 10342054 9967507 9190530 8805431 ...
## $ Treatment: chr "healthy" "healthy" "healthy" "healthy" ...
## $ Samples  : chr "MP2058" "MP2087" "MP2068" "MP2068" ...
## $ Kingdom  : chr "Bacteria" "Bacteria" "Bacteria" "Bacteria" ...
## $ Phylum   : chr "Actinobacteria" "Actinobacteria" "Actinobacteria" "Proteobacteria" ...

absolute_df_phylum$Phylum <- as.factor(absolute_df_phylum$Phylum)
phylum_colors_abs <- colorRampPalette(brewer.pal(8,"Dark2")) (length(levels(absolute_df_phylum$Phylum)))

absolute_plot <- ggplot(data= absolute_df_phylum, aes(x=Sample, y=Abundance, fill=Phylum))+  

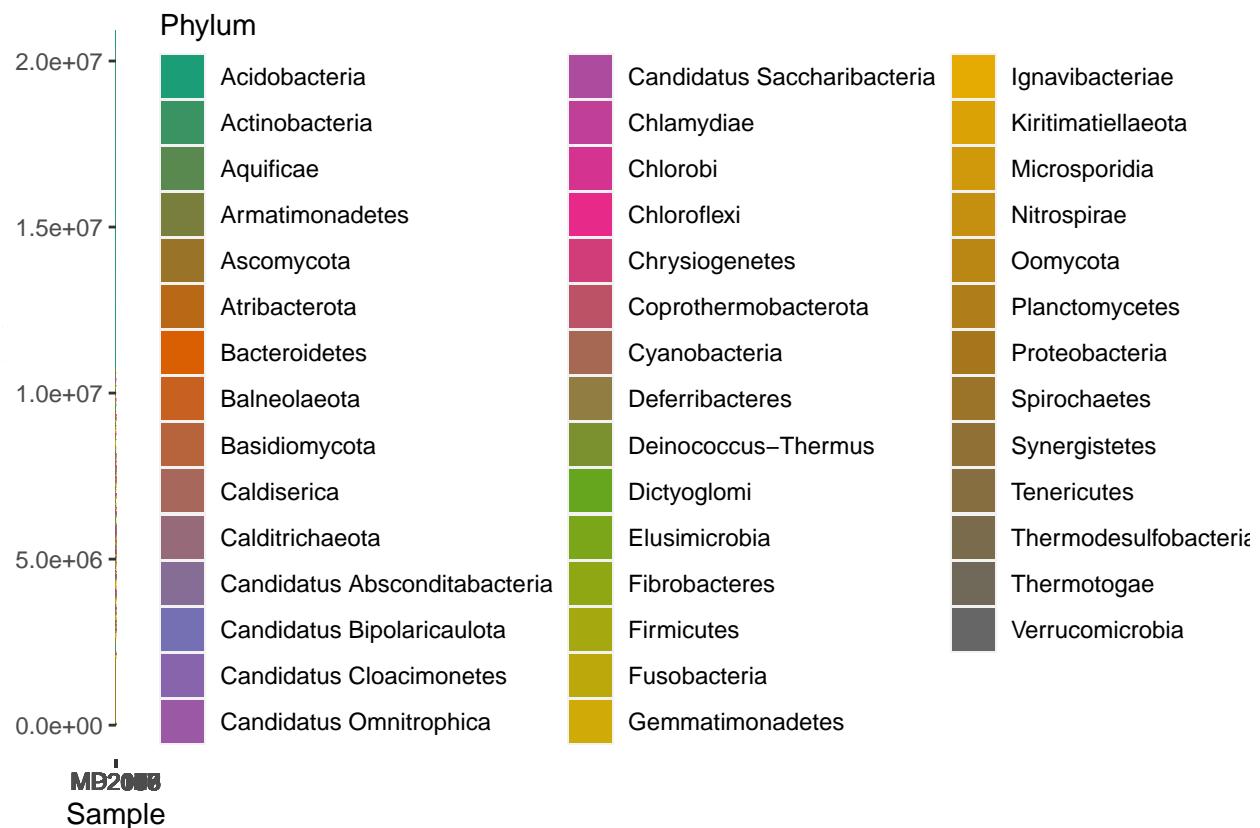
  geom_bar(aes(), stat="identity", position="stack")+
  scale_fill_manual(values = phylum_colors_abs)

percentages_df_phylum$Phylum <- as.factor(percentages_df_phylum$Phylum)
phylum_colors_rel<- colorRampPalette(brewer.pal(8,"Dark2")) (length(levels(percentages_df_phylum$Phylum)))
relative_plot <- ggplot(data=percentages_df_phylum, aes(x=Sample, y=Abundance, fill=Phylum))+  

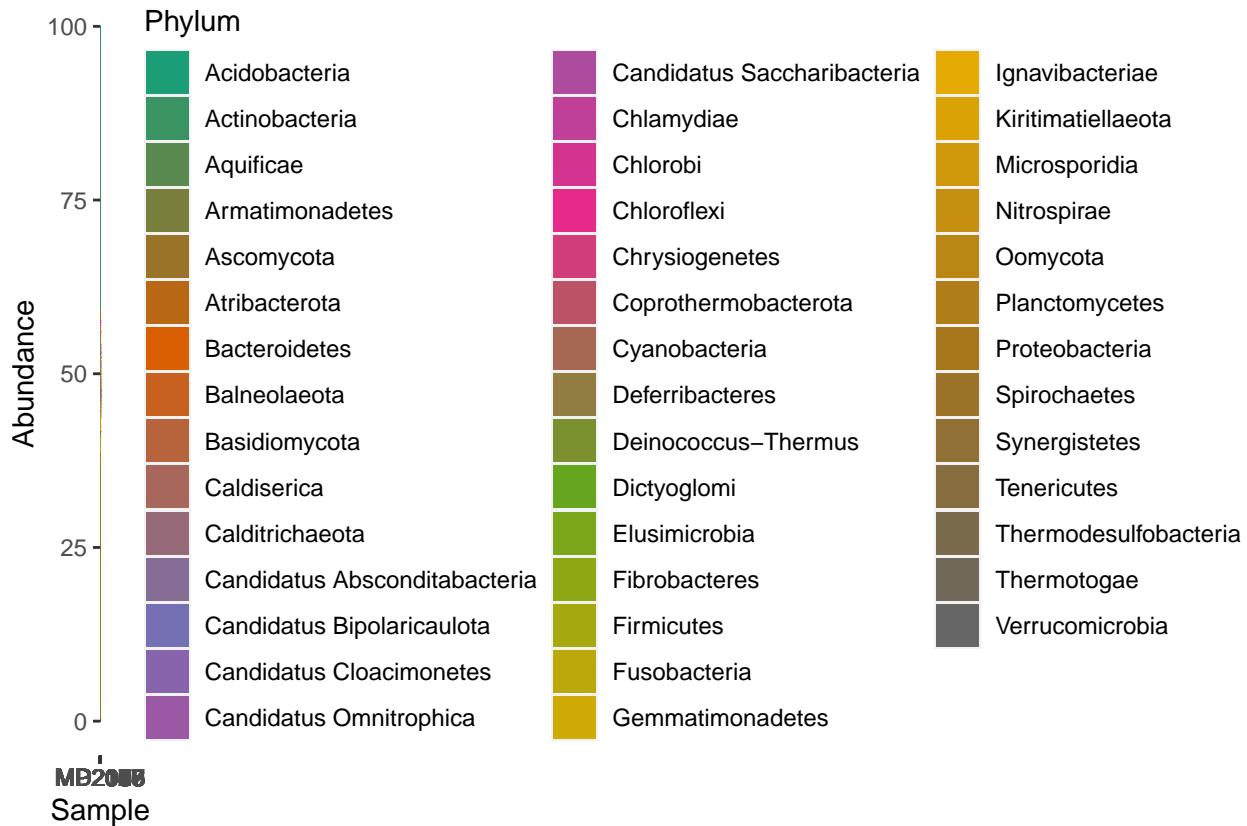
  geom_bar(aes(), stat="identity", position="stack")+
  scale_fill_manual(values = phylum_colors_rel)

absolute_plot

```



relative_plot



Usaremos un comando llamado unique() para explorar cuántos filos y reinos tenemos.

```
unique(fresa_kraken_fil@tax_table@Data[, "Kingdom"])
```

```
## [1] "Bacteria" "Eukaryota"
```

```
unique(fresa_kraken_fil@tax_table@Data[, "Phylum"])
```

```
## [1] "Actinobacteria"           "Firmicutes"
## [3] "Deinococcus-Thermus"      "Cyanobacteria"
## [5] "Chloroflexi"              "Armatimonadetes"
## [7] "Tenericutes"               "Proteobacteria"
## [9] "Planctomycetes"            "Verrucomicrobia"
## [11] "Kiritimatiellaeota"        "Chlamydiae"
## [13] "Candidatus Omnitrophica"   "Bacteroidetes"
## [15] "Chlorobi"                  "Balneolaeota"
## [17] "Ignavibacteriae"          "Gemmatimonadetes"
## [19] "Fibrobacteres"             "Candidatus Cloacimonetes"
## [21] "Acidobacteria"             "Nitrospirae"
## [23] "Spirochaetes"              "Synergistetes"
## [25] "Candidatus Bipolaricaulota" "Candidatus Saccharibacteria"
## [27] ""                          "Candidatus Absconditabacteria"
## [29] "Thermotogae"                "Aquificae"
## [31] "Thermodesulfobacteria"       "Deferribacteres"
## [33] "Fusobacteria"                 "ChrysioGenetes"
```

```

## [35] "Calditrichaeota"           "Elusimicrobia"
## [37] "Caldiserica"              "Coprothermobacterota"
## [39] "Atribacterota"             "Dictyoglomi"
## [41] "Ascomycota"                "Basidiomycota"
## [43] "Microsporidia"              "Oomycota"

```

con esto podemos ver cuantos “Eukaryota” tenemos en “Kingdom”

```
sum(fresa_kraken_fil@tax_table@Data[,"Kingdom"] == "Eukaryota")
```

```
## [1] 181
```

y cuantos “Bacteria”

```
sum(fresa_kraken_fil@tax_table@Data[,"Kingdom"] == "Bacteria")
```

```
## [1] 8822
```

Diversidad Beta

veremos aqui solo el reino Eucariota

```
merge_Eukaryota <- subset_taxa(fresa_kraken_fil,Kingdom=="Eukaryota")
```

sacamos las abundancias relativas

```
percentages_Eukaryota <- transform_sample_counts(merge_Eukaryota, function(x) x*100 / sum(x) )
percentages_Eukaryota_df <- psmelt(percentages_Eukaryota)
```

beta diversidad de Eukaryota

```

meta_ord_Eukaryota <- ordinate(physeq = percentages_Eukaryota, method = "NMDS", distance = "bray")

## Wisconsin double standardization
## Run 0 stress 0.1341389
## Run 1 stress 0.1532966
## Run 2 stress 0.1345939
## ... Procrustes: rmse 0.0148127 max resid 0.08044787
## Run 3 stress 0.1651012
## Run 4 stress 0.161598
## Run 5 stress 0.1341977
## ... Procrustes: rmse 0.01193859 max resid 0.08063864
## Run 6 stress 0.1534922
## Run 7 stress 0.1398622
## Run 8 stress 0.139463
## Run 9 stress 0.134595
## ... Procrustes: rmse 0.01476233 max resid 0.08009187
## Run 10 stress 0.1682146

```

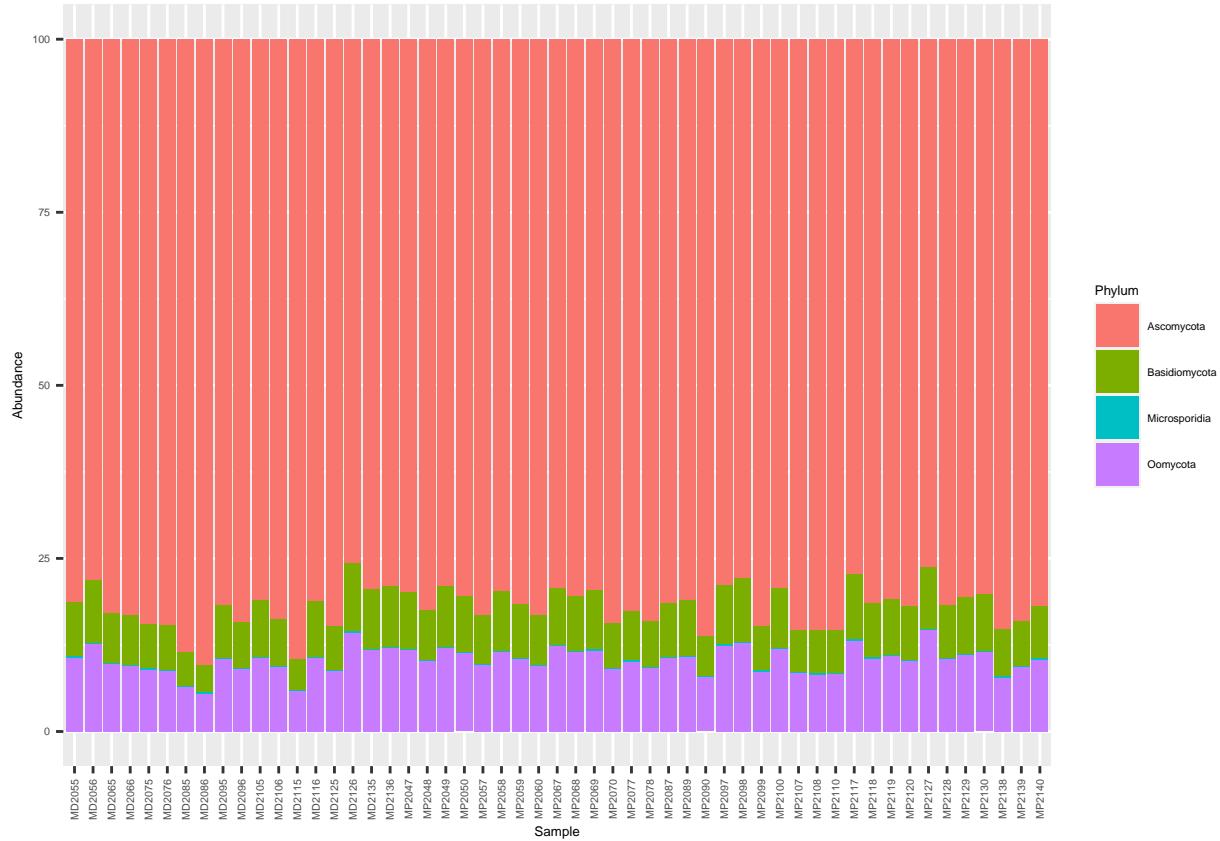
```
## Run 11 stress 0.1398632
## Run 12 stress 0.1538696
## Run 13 stress 0.170902
## Run 14 stress 0.1368375
## Run 15 stress 0.1392626
## Run 16 stress 0.1744738
## Run 17 stress 0.134176
## ... Procrustes: rmse 0.002296515 max resid 0.01145256
## Run 18 stress 0.1540229
## Run 19 stress 0.1341974
## ... Procrustes: rmse 0.01189633 max resid 0.08036325
## Run 20 stress 0.1691077
## *** Best solution was not repeated -- monoMDS stopping criteria:
##      6: no. of iterations >= maxit
##      14: stress ratio > sratmax
```

```
plot_ordination(physeq = percentages_Eukaryota, ordination = meta_ord_Eukaryota, color = "Treatment") +  
  geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@.Data)), size = 3, vjust = 1.5)
```



Eukaryota por Phylum

```
ggplot(data= percentages_Eukaryota_df, aes(x=Sample, y=Abundance, fill=Phylum))+  
  geom_bar(aes(), stat="identity", position="stack") +  
  theme(text = element_text(size = 5),  
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))
```



```
merge_Eukaryota_Phylum<-tax_glom(merge_Eukaryota, taxrank="Phylum")
```

sacamos las abundancias relativas

```
percentages_Eukaryota_Phylum <- transform_sample_counts(merge_Eukaryota_Phylum, function(x) x*100 / sum(x))
percentages_Eukaryota_Phylum_df <- psmelt(percentages_Eukaryota_Phylum)
meta_ord_Eukaryota_Phylum <- ordinate(physeq = percentages_Eukaryota_Phylum, method = "NMDS", distance = "euclidean")

## Square root transformation
## Wisconsin double standardization
## Run 0 stress 0.01148681
## Run 1 stress 6.884553e-05
## ... New best solution
## ... Procrustes: rmse 0.03466362 max resid 0.09090158
## Run 2 stress 9.896019e-05
## ... Procrustes: rmse 0.0002609088 max resid 0.0006615746
## ... Similar to previous best
## Run 3 stress 7.961944e-05
## ... Procrustes: rmse 3.289214e-05 max resid 8.256855e-05
## ... Similar to previous best
## Run 4 stress 0.002104582
## Run 5 stress 9.8706e-05
## ... Procrustes: rmse 8.717609e-05 max resid 0.0002328892
## ... Similar to previous best
## Run 6 stress 9.989201e-05
```

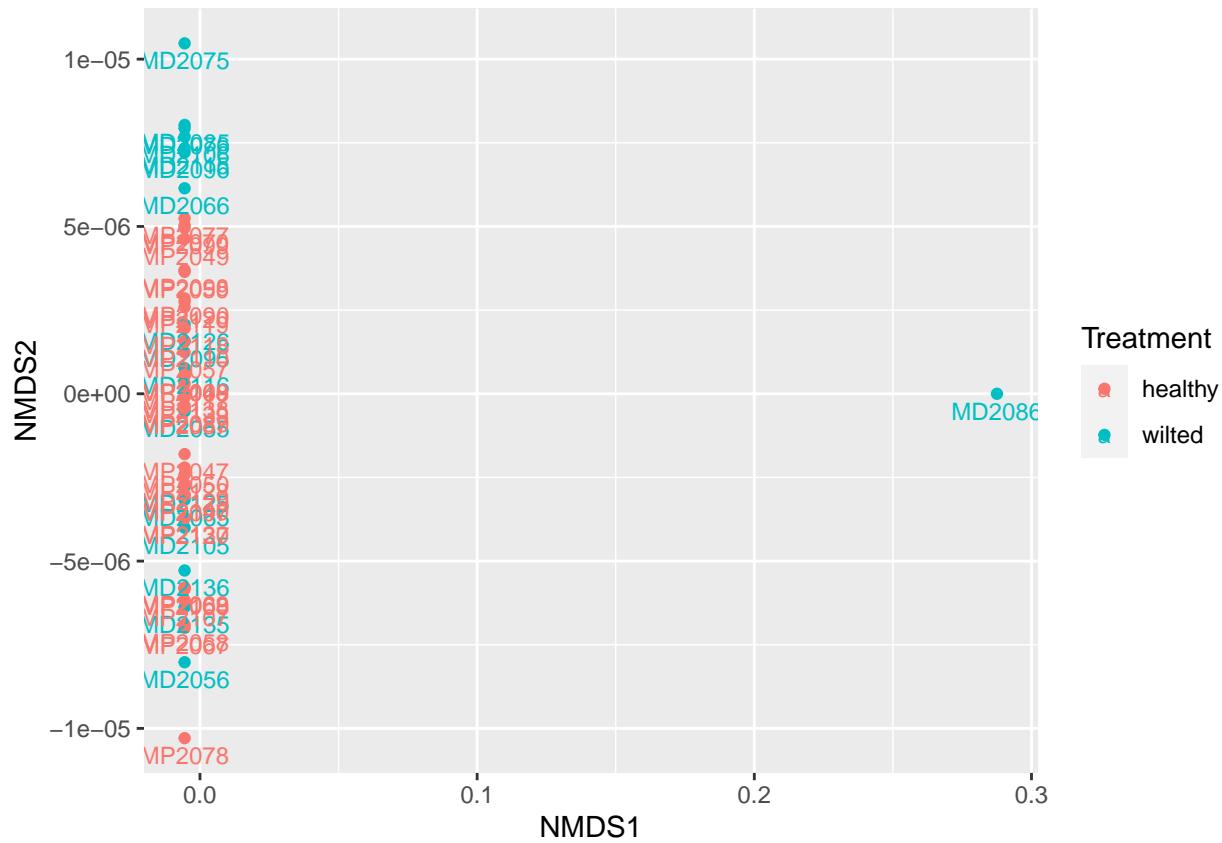
```

## ... Procrustes: rmse 0.0001158955 max resid 0.000312878
## ... Similar to previous best
## Run 7 stress 9.315593e-05
## ... Procrustes: rmse 8.28591e-05 max resid 0.00021891
## ... Similar to previous best
## Run 8 stress 9.905098e-05
## ... Procrustes: rmse 0.0001607623 max resid 0.0004292566
## ... Similar to previous best
## Run 9 stress 9.583544e-05
## ... Procrustes: rmse 0.0001231222 max resid 0.0003060431
## ... Similar to previous best
## Run 10 stress 0.0001070184
## ... Procrustes: rmse 0.0003112862 max resid 0.000797225
## ... Similar to previous best
## Run 11 stress 9.8134e-05
## ... Procrustes: rmse 0.0001086383 max resid 0.0002710078
## ... Similar to previous best
## Run 12 stress 7.707195e-05
## ... Procrustes: rmse 3.354619e-05 max resid 8.066368e-05
## ... Similar to previous best
## Run 13 stress 9.681288e-05
## ... Procrustes: rmse 3.759548e-05 max resid 7.556245e-05
## ... Similar to previous best
## Run 14 stress 7.93684e-05
## ... Procrustes: rmse 5.861831e-05 max resid 0.0001740127
## ... Similar to previous best
## Run 15 stress 9.869761e-05
## ... Procrustes: rmse 0.0001635045 max resid 0.000439497
## ... Similar to previous best
## Run 16 stress 0.0001067829
## ... Procrustes: rmse 0.0003145157 max resid 0.0008083435
## ... Similar to previous best
## Run 17 stress 9.223488e-05
## ... Procrustes: rmse 5.507838e-05 max resid 0.000156643
## ... Similar to previous best
## Run 18 stress 8.994926e-05
## ... Procrustes: rmse 8.329065e-05 max resid 0.000216376
## ... Similar to previous best
## Run 19 stress 9.592971e-05
## ... Procrustes: rmse 9.283918e-05 max resid 0.0002423359
## ... Similar to previous best
## Run 20 stress 9.650682e-05
## ... Procrustes: rmse 0.0001594628 max resid 0.0004286399
## ... Similar to previous best
## *** Best solution repeated 18 times

## Warning in metaMDS(veganifyOTU(physeq), distance, ...): stress is (nearly) zero:
## you may have insufficient data

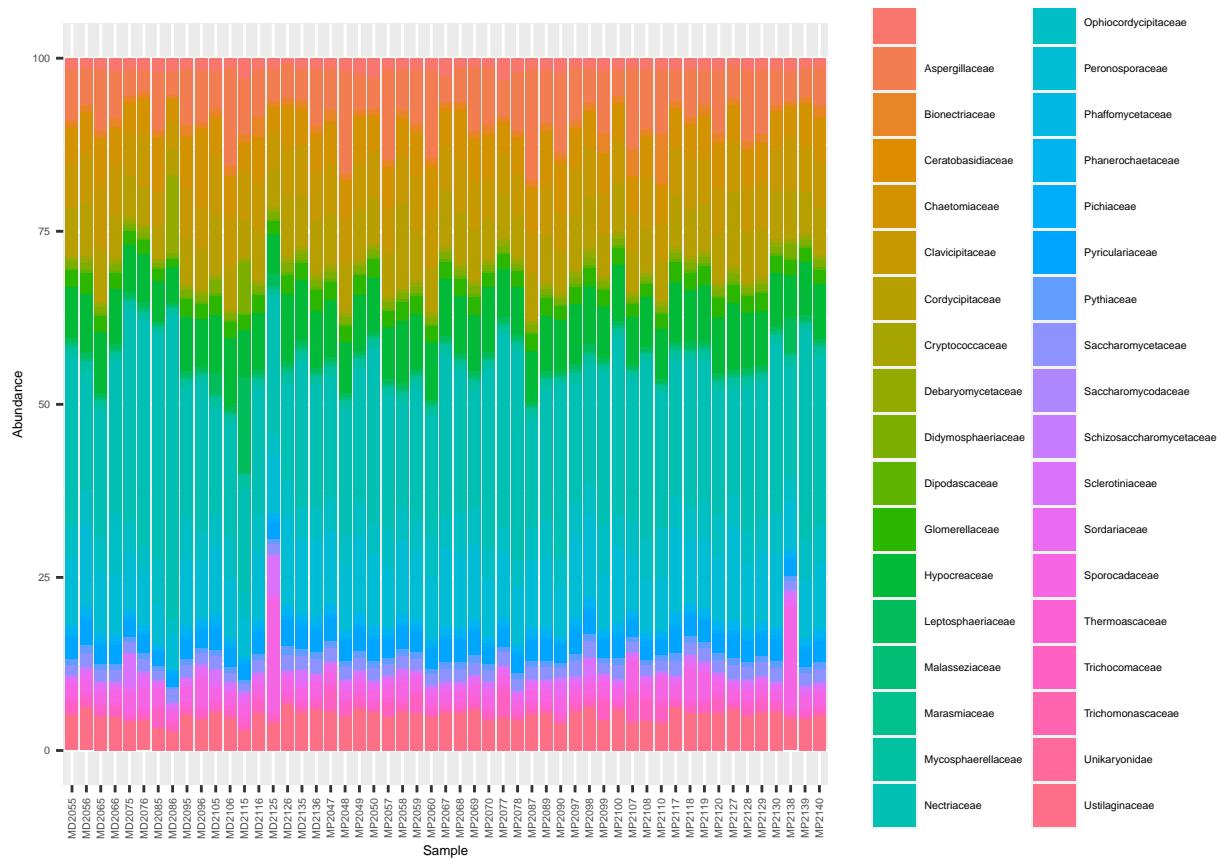
plot_ordination(physeq = percentages_Eukaryota_Phylum, ordination = meta_ord_Eukaryota_Phylum, color =
  geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@Data))), size = 3, vjust = 1.5)

```



Eukaryota por Family

```
ggplot(data= percentages_Eukaryota_df, aes(x=Sample, y=Abundance, fill=Family))+  
  geom_bar(aes(), stat="identity", position="stack") +  
  theme(text = element_text(size = 5),  
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))
```



```
merge_Eukaryota_Family<-tax_glom(merge_Eukaryota, taxrank="Family")
```

sacamos las abundancias relativas

```
percentages_Eukaryota_Family <- transform_sample_counts(merge_Eukaryota_Family, function(x) x*100 / sum(x))
percentages_Eukaryota_Family_df <- psmelt(percentages_Eukaryota_Family)
meta_ord_Eukaryota_Family <- ordinate(physeq = percentages_Eukaryota_Family, method = "NMDS", distance =
```

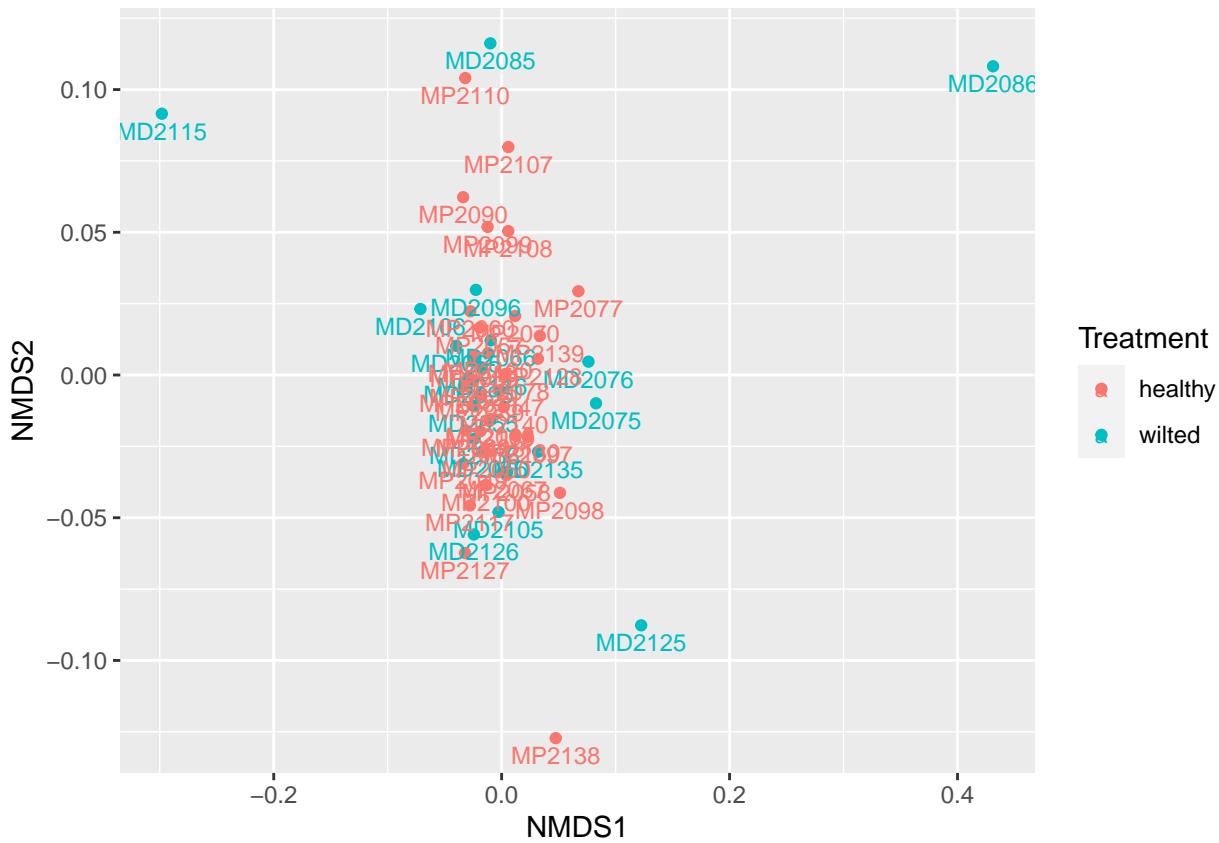
```
## Wisconsin double standardization
## Run 0 stress 0.1089081
## Run 1 stress 0.1352082
## Run 2 stress 0.1311828
## Run 3 stress 0.1089299
## ... Procrustes: rmse 0.00809935 max resid 0.0543764
## Run 4 stress 0.1090587
## ... Procrustes: rmse 0.004641029 max resid 0.0158981
## Run 5 stress 0.1352085
## Run 6 stress 0.133809
## Run 7 stress 0.1448148
## Run 8 stress 0.1091144
## ... Procrustes: rmse 0.009556559 max resid 0.05386817
## Run 9 stress 0.1089181
## ... Procrustes: rmse 0.001626776 max resid 0.008532758
## ... Similar to previous best
## Run 10 stress 0.108889
```

```

## ... New best solution
## ... Procrustes: rmse 0.00221236 max resid 0.009131389
## ... Similar to previous best
## Run 11 stress 0.1089296
## ... Procrustes: rmse 0.008440814 max resid 0.05465003
## Run 12 stress 0.1265859
## Run 13 stress 0.1448173
## Run 14 stress 0.1091257
## ... Procrustes: rmse 0.01001014 max resid 0.05406511
## Run 15 stress 0.1089493
## ... Procrustes: rmse 0.008143314 max resid 0.05419858
## Run 16 stress 0.1265942
## Run 17 stress 0.1089487
## ... Procrustes: rmse 0.008119745 max resid 0.05416392
## Run 18 stress 0.1088971
## ... Procrustes: rmse 0.002259855 max resid 0.008967621
## ... Similar to previous best
## Run 19 stress 0.1088885
## ... New best solution
## ... Procrustes: rmse 0.001587068 max resid 0.009066863
## ... Similar to previous best
## Run 20 stress 0.108897
## ... Procrustes: rmse 0.001707455 max resid 0.008985377
## ... Similar to previous best
## *** Best solution repeated 2 times

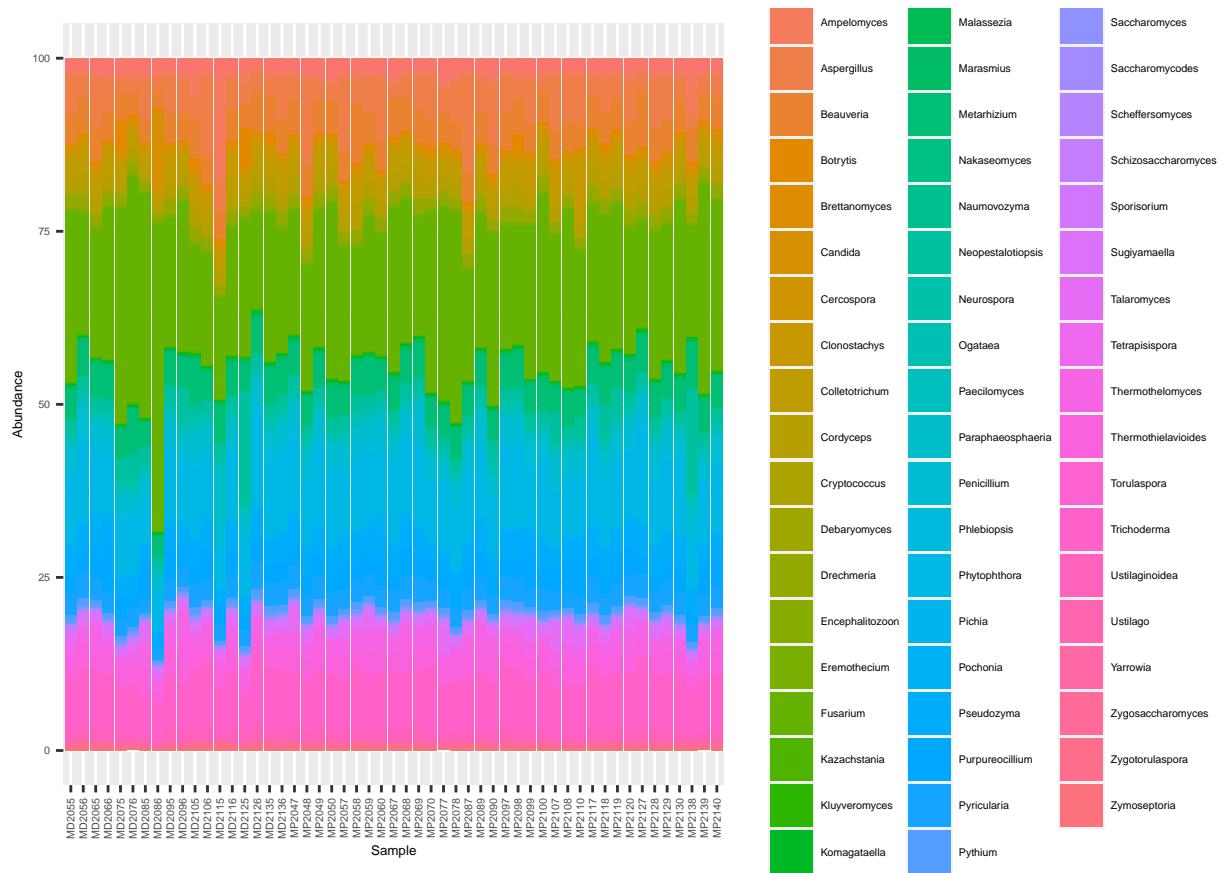
plot_ordination(physeq = percentages_Eukaryota_Family, ordination = meta_ord_Eukaryota_Family, color =
  geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@Data))), size = 3, vjust = 1.5)

```



Eukaryota por Genus

```
ggplot(data= percentages_Eukaryota_df, aes(x=Sample, y=Abundance, fill=Genus)) +  
  geom_bar(aes(), stat="identity", position="stack") +  
  theme(text = element_text(size = 5),  
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))
```



```
merge_Eukaryota_Genus <- tax_glom(merge_Eukaryota, taxrank = "Genus")
```

sacamos las abundancias relativas

```
percentages_Eukaryota_Genus <- transform_sample_counts(merge_Eukaryota_Genus, function(x) x*100 / sum(x))
percentages_Eukaryota_Genus_df <- psmelt(percentages_Eukaryota_Genus)
meta_ord_Eukaryota_Genus <- ordinate(physeq = percentages_Eukaryota_Genus, method = "NMDS", distance =
```

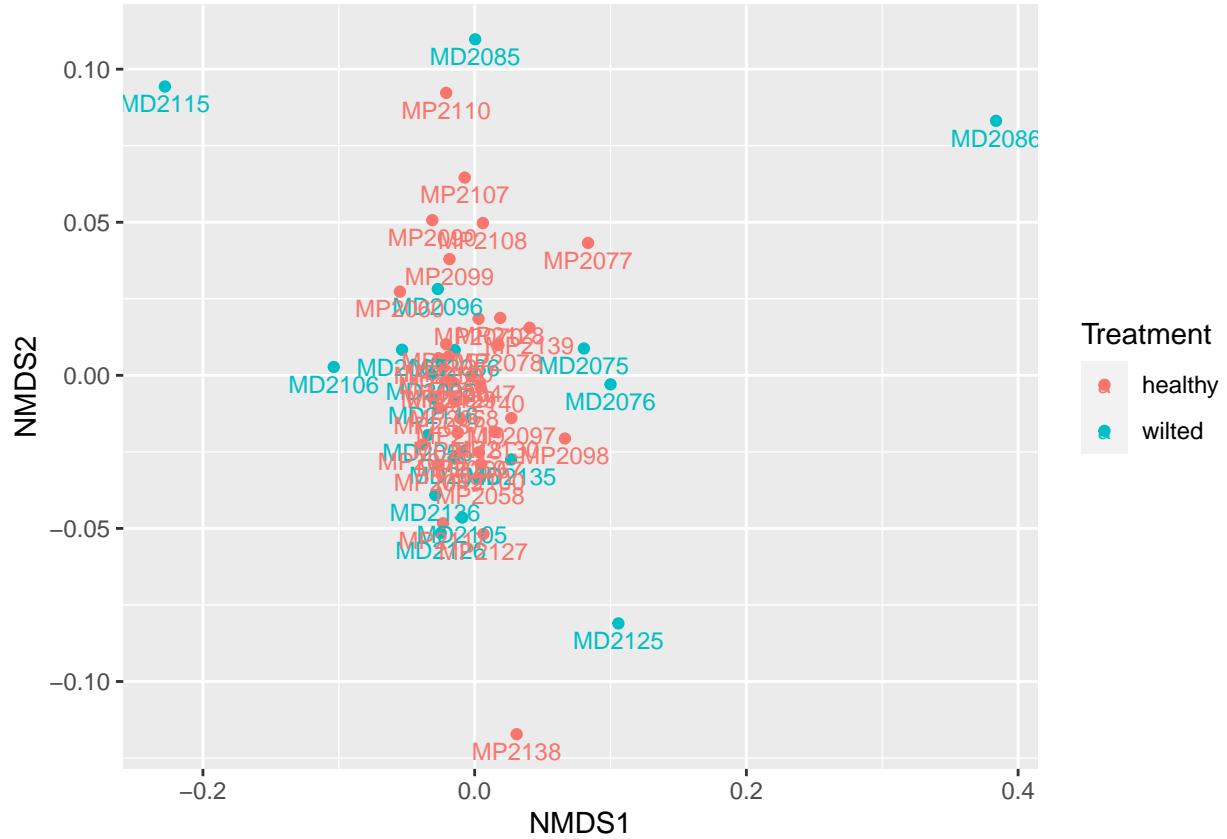
```
## Wisconsin double standardization
## Run 0 stress 0.110709
## Run 1 stress 0.1119791
## Run 2 stress 0.1107089
## ... New best solution
## ... Procrustes: rmse 0.0008993083 max resid 0.005141848
## ... Similar to previous best
## Run 3 stress 0.1222465
## Run 4 stress 0.1107092
## ... Procrustes: rmse 0.000849638 max resid 0.004145687
## ... Similar to previous best
## Run 5 stress 0.1107089
## ... Procrustes: rmse 1.807782e-05 max resid 9.561543e-05
## ... Similar to previous best
## Run 6 stress 0.1125077
## Run 7 stress 0.1235024
## Run 8 stress 0.1107095
```

```

## ... Procrustes: rmse 0.0002295058 max resid 0.001323315
## ... Similar to previous best
## Run 9 stress 0.1107091
## ... Procrustes: rmse 9.004012e-05 max resid 0.0005153199
## ... Similar to previous best
## Run 10 stress 0.1119811
## Run 11 stress 0.111907
## Run 12 stress 0.1107087
## ... New best solution
## ... Procrustes: rmse 0.0007528014 max resid 0.004294196
## ... Similar to previous best
## Run 13 stress 0.1109204
## ... Procrustes: rmse 0.009593333 max resid 0.05090563
## Run 14 stress 0.1107091
## ... Procrustes: rmse 0.0008273659 max resid 0.004726449
## ... Similar to previous best
## Run 15 stress 0.1223849
## Run 16 stress 0.1235032
## Run 17 stress 0.1119792
## Run 18 stress 0.1107099
## ... Procrustes: rmse 0.001116256 max resid 0.006357736
## ... Similar to previous best
## Run 19 stress 0.1107095
## ... Procrustes: rmse 0.001004428 max resid 0.005746087
## ... Similar to previous best
## Run 20 stress 0.1107102
## ... Procrustes: rmse 0.001162582 max resid 0.006843074
## ... Similar to previous best
## *** Best solution repeated 5 times

plot_ordination(physeq = percentages_Eukaryota_Genus, ordination = meta_ord_Eukaryota_Genus, color = "T"
geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@Data)), size = 3, vjust = 1.5)

```

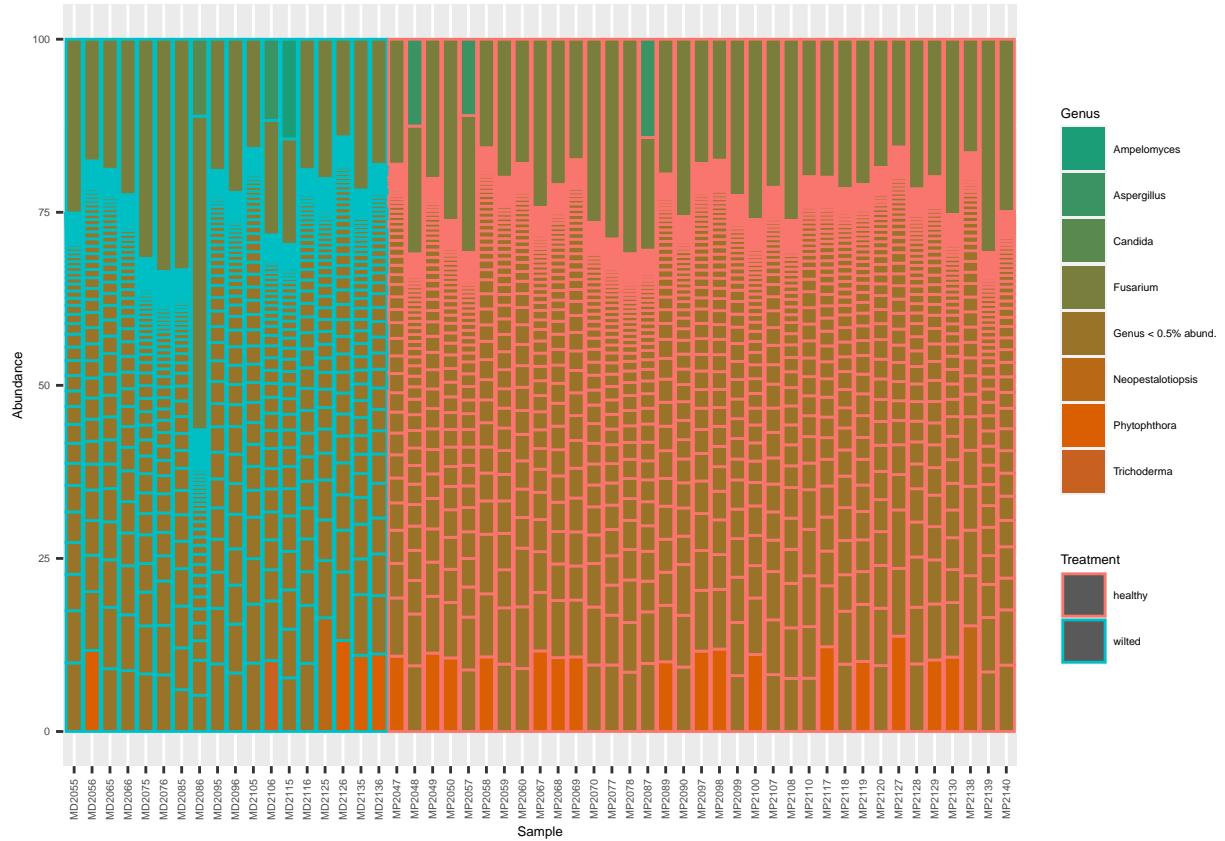


tomando diferentes porcentajes de abundancia

```
percentages_Eukaryota_Genus_df$Genus[percentages_Eukaryota_Genus_df$Abundance < 10.0] <- "Genus < 0.5%"
percentages_Eukaryota_Genus_df$Genus <- as.factor(percentages_Eukaryota_Genus_df$Genus)

genus_colors_rel <- colorRampPalette(brewer.pal(8, "Dark2")) (length(levels(percentages_Eukaryota_Genus_df$Genus))
relative_plot <- ggplot(data=percentages_Eukaryota_Genus_df, aes(x=Sample, y=Abundance, fill=Genus ,color=Genus))
  geom_bar(aes(), stat="identity", position="stack") +
  scale_fill_manual(values = phylum_colors_rel) +
  theme(text = element_text(size = 5),
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))

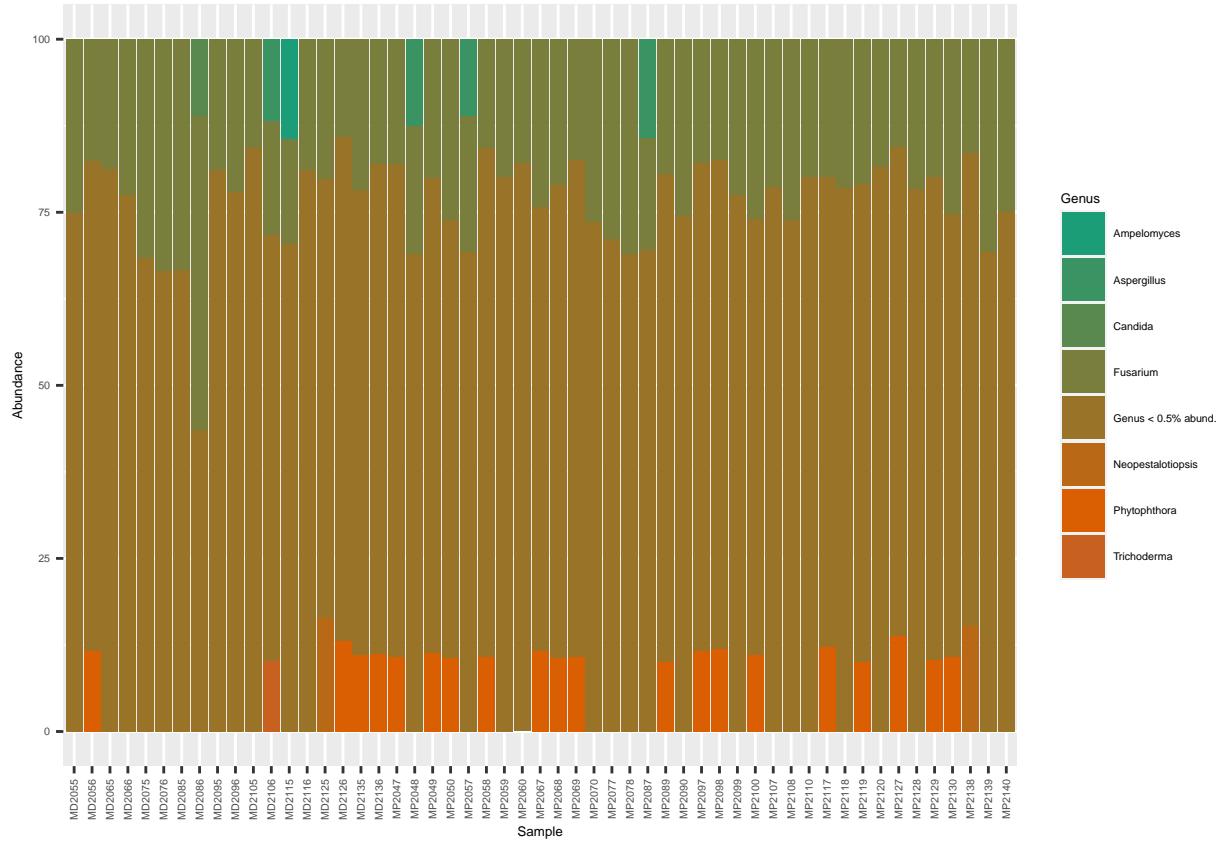
relative_plot
```



```

relative_plot <- ggplot(data=percentages_Eukaryota_Genus_df, aes(x=Sample, y=Abundance, fill=Genus))+
  geom_bar(aes(), stat="identity", position="stack") +
  scale_fill_manual(values = phylum_colors_rel) +
  theme(text = element_text(size = 5),
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))
relative_plot

```



sacando la beta diversidad con aglomerado de 10%

```
meta_ord_Eukaryota_Genus <- ordinate(physeq = percentages_Eukaryota_Genus, method = "NMDS", distance = 

## Wisconsin double standardization
## Run 0 stress 0.110709
## Run 1 stress 0.1107087
## ... New best solution
## ... Procrustes: rmse 0.0008065132 max resid 0.00461678
## ... Similar to previous best
## Run 2 stress 0.1107096
## ... Procrustes: rmse 0.001006654 max resid 0.005727743
## ... Similar to previous best
## Run 3 stress 0.12363
## Run 4 stress 0.1119821
## Run 5 stress 0.1235019
## Run 6 stress 0.1107091
## ... Procrustes: rmse 0.0001875669 max resid 0.001072649
## ... Similar to previous best
## Run 7 stress 0.1222456
## Run 8 stress 0.11071
## ... Procrustes: rmse 0.0004354107 max resid 0.002487007
## ... Similar to previous best
## Run 9 stress 0.1119944
## Run 10 stress 0.110709
## ... Procrustes: rmse 0.0001635567 max resid 0.0009301629
```

```

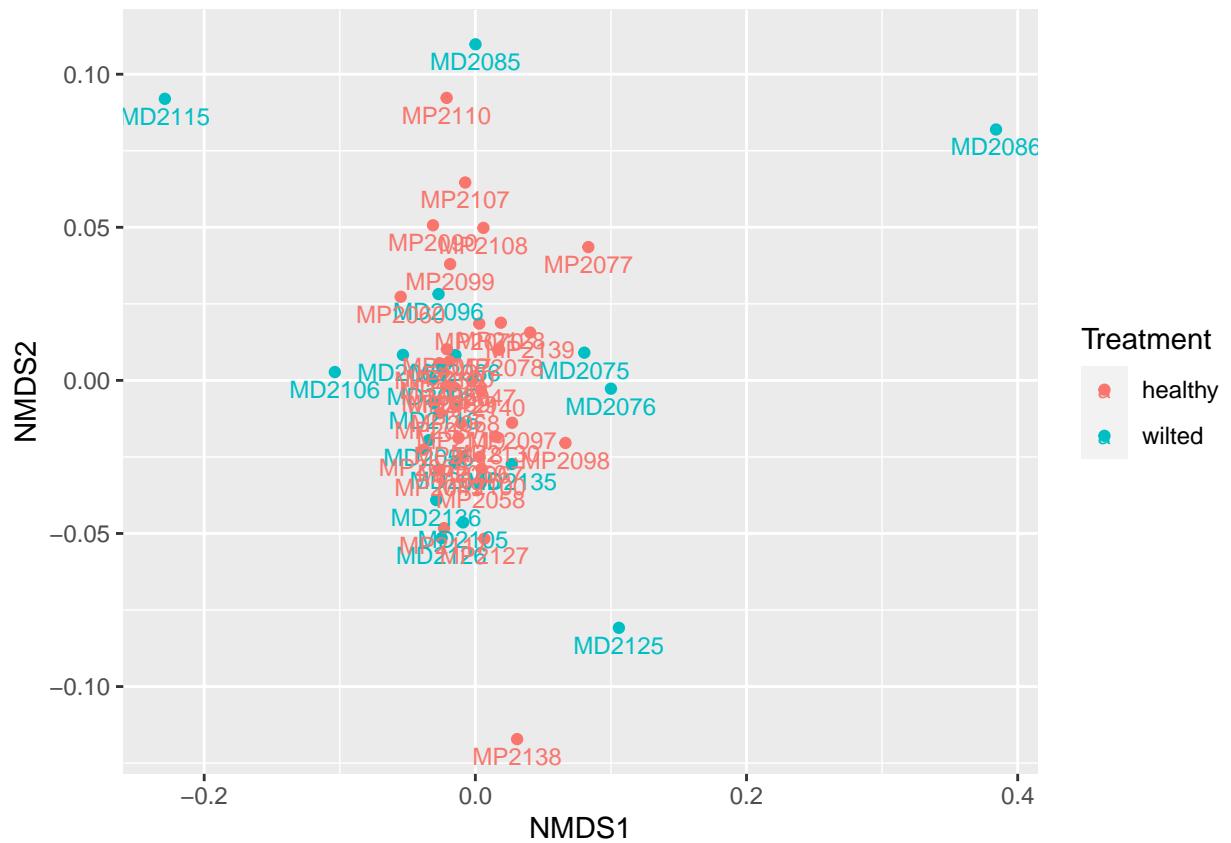
## ... Similar to previous best
## Run 11 stress 0.123631
## Run 12 stress 0.1107093
## ... Procrustes: rmse 0.0002703302 max resid 0.001515246
## ... Similar to previous best
## Run 13 stress 0.1119822
## Run 14 stress 0.1107097
## ... Procrustes: rmse 0.0003814028 max resid 0.002171307
## ... Similar to previous best
## Run 15 stress 0.1107093
## ... Procrustes: rmse 0.0008624006 max resid 0.004911933
## ... Similar to previous best
## Run 16 stress 0.1236308
## Run 17 stress 0.1127217
## Run 18 stress 0.1223835
## Run 19 stress 0.1108623
## ... Procrustes: rmse 0.0075558301 max resid 0.04031273
## Run 20 stress 0.1235023
## *** Best solution repeated 8 times

```

```

plot_ordination(physeq = percentages_Eukaryota_Genus, ordination = meta_ord_Eukaryota_Genus, color = "Treatment"
geom_text(mapping = aes(label = colnames(fresa_kraken_file@otu_table@Data)), size = 3, vjust = 1.5)

```



Veremos aqui solo el reino Bacteriano

```
merge_Bacteria<-subset_taxa(fresa_kraken_fil,Kingdom=="Bacteria")
```

sacamos las abundancias relativas

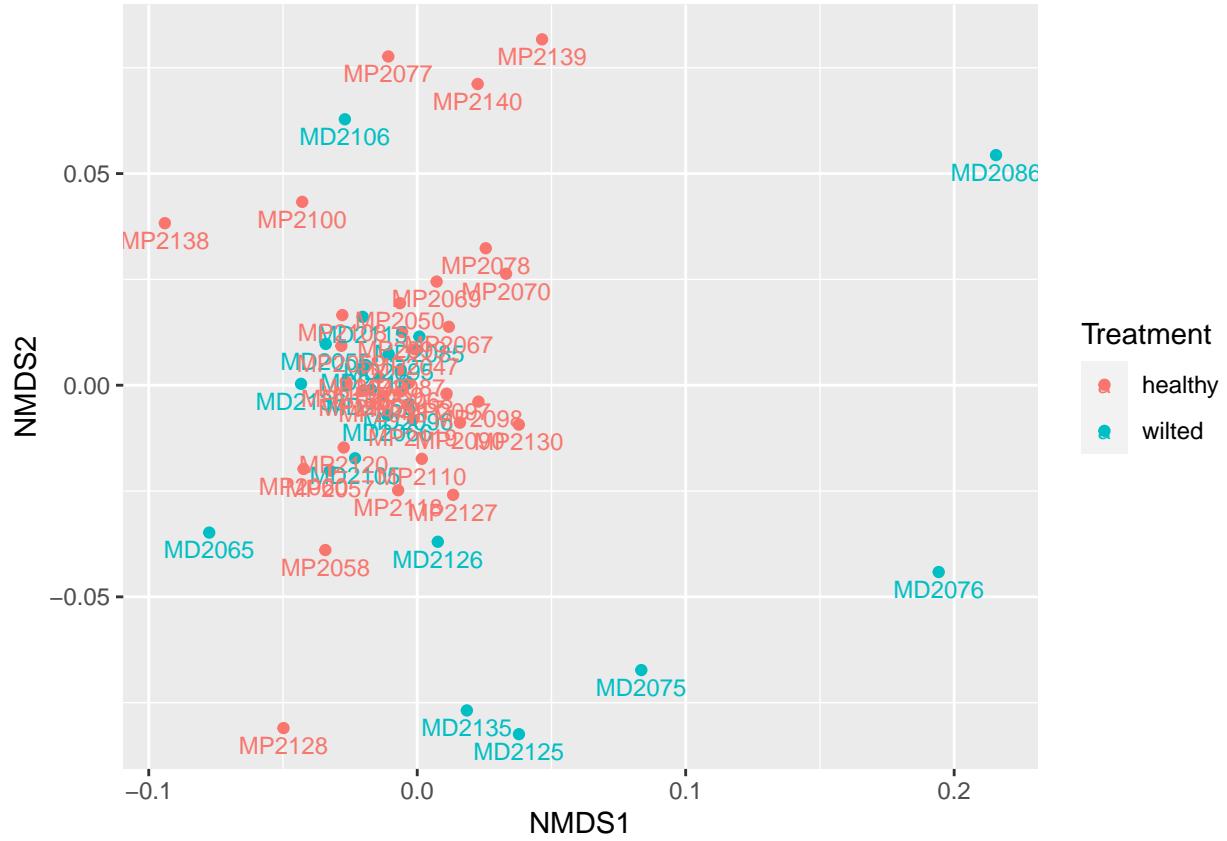
```
percentages_Bacteria <- transform_sample_counts(merge_Bacteria, function(x) x*100 / sum(x) )
percentages_Bacteria_df <- psmelt(percentages_Bacteria)
```

beta diversidad de Bacteria

```
meta_ord_Bacteria <- ordinate(physeq = percentages_Bacteria, method = "NMDS", distance = "bray")
```

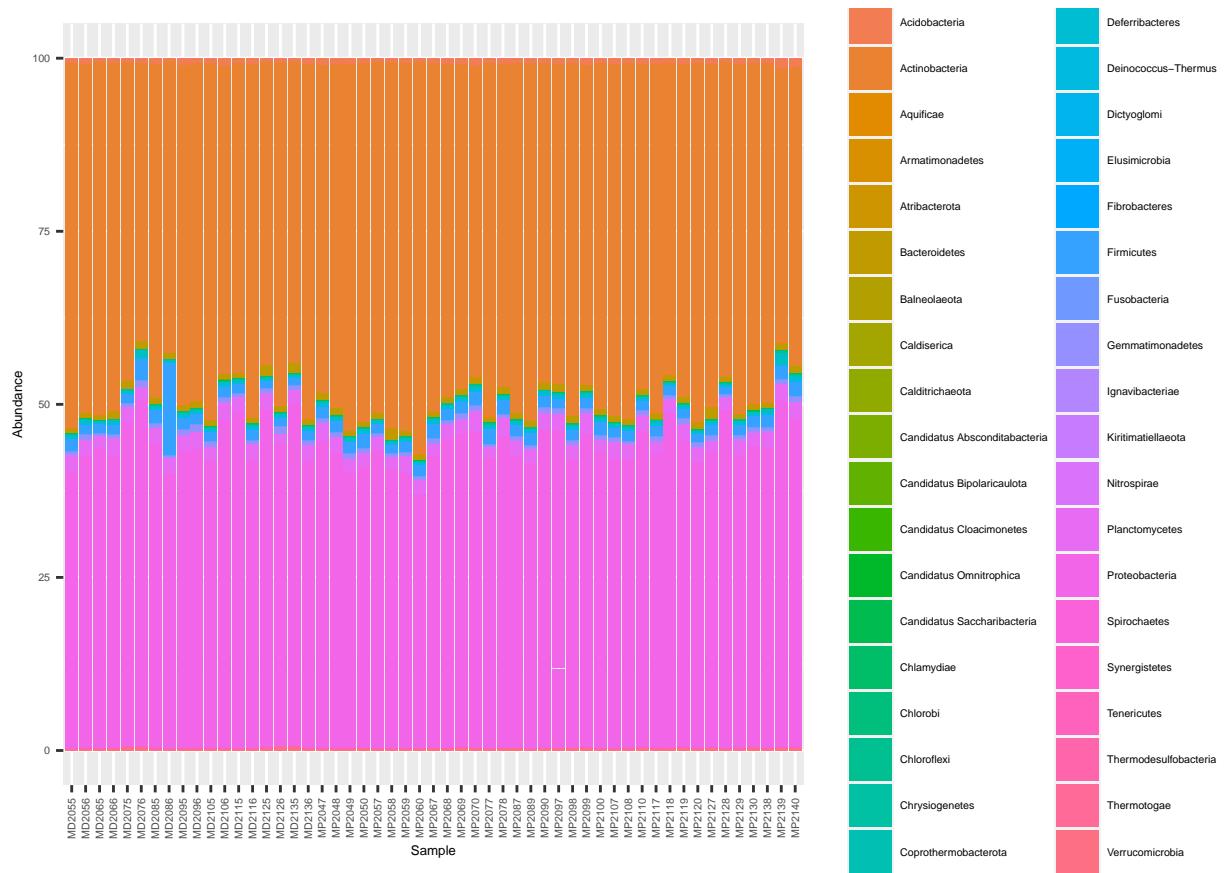
```
## Wisconsin double standardization
## Run 0 stress 0.1680232
## Run 1 stress 0.1706871
## Run 2 stress 0.1661875
## ... New best solution
## ... Procrustes: rmse 0.05345565 max resid 0.2639486
## Run 3 stress 0.165798
## ... New best solution
## ... Procrustes: rmse 0.07350935 max resid 0.3685006
## Run 4 stress 0.1855277
## Run 5 stress 0.1714973
## Run 6 stress 0.2186955
## Run 7 stress 0.1657578
## ... New best solution
## ... Procrustes: rmse 0.003916201 max resid 0.02076304
## Run 8 stress 0.1795456
## Run 9 stress 0.1869755
## Run 10 stress 0.1694656
## Run 11 stress 0.170957
## Run 12 stress 0.1631208
## ... New best solution
## ... Procrustes: rmse 0.06372081 max resid 0.3458528
## Run 13 stress 0.1662643
## Run 14 stress 0.1662608
## Run 15 stress 0.1657274
## Run 16 stress 0.1945724
## Run 17 stress 0.1666047
## Run 18 stress 0.1660681
## Run 19 stress 0.170165
## Run 20 stress 0.1689046
## *** Best solution was not repeated -- monoMDS stopping criteria:
##      2: no. of iterations >= maxit
##      18: stress ratio > sratmax
```

```
plot_ordination(physeq = percentages_Bacteria, ordination = meta_ord_Bacteria, color = "Treatment") +
  geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@Data)), size = 3, vjust = 1.5)
```



Bacterias por Phylum

```
ggplot(data= percentages_Bacteria_df, aes(x=Sample, y=Abundance, fill=Phylum))+  
  geom_bar(aes(), stat="identity", position="stack") +  
  theme(text = element_text(size = 5),  
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))
```



```
merge_Bacteria_Phylum<-tax_glm(merge_Bacteria, taxrank="Phylum")
```

sacamos las abundancias relativas

```
percentages_Bacteria_Phylum <- transform_sample_counts(merge_Bacteria_Phylum, function(x) x*100 / sum(x))
percentages_Bacteria_Phylum_df <- psmelt(percentages_Bacteria_Phylum)
meta_ord_Bacteria_Phylum <- ordinate(physeq = percentages_Bacteria_Phylum, method = "NMDS", distance =
```

```
## Square root transformation
## Wisconsin double standardization
## Run 0 stress 0.1520004
## Run 1 stress 0.1736455
## Run 2 stress 0.1572809
## Run 3 stress 0.1582412
## Run 4 stress 0.1614675
## Run 5 stress 0.1513788
## ... New best solution
## ... Procrustes: rmse 0.01100183 max resid 0.04914669
## Run 6 stress 0.1600298
## Run 7 stress 0.1660033
## Run 8 stress 0.1654843
## Run 9 stress 0.1600302
## Run 10 stress 0.1525511
## Run 11 stress 0.152551
## Run 12 stress 0.160173
```

```

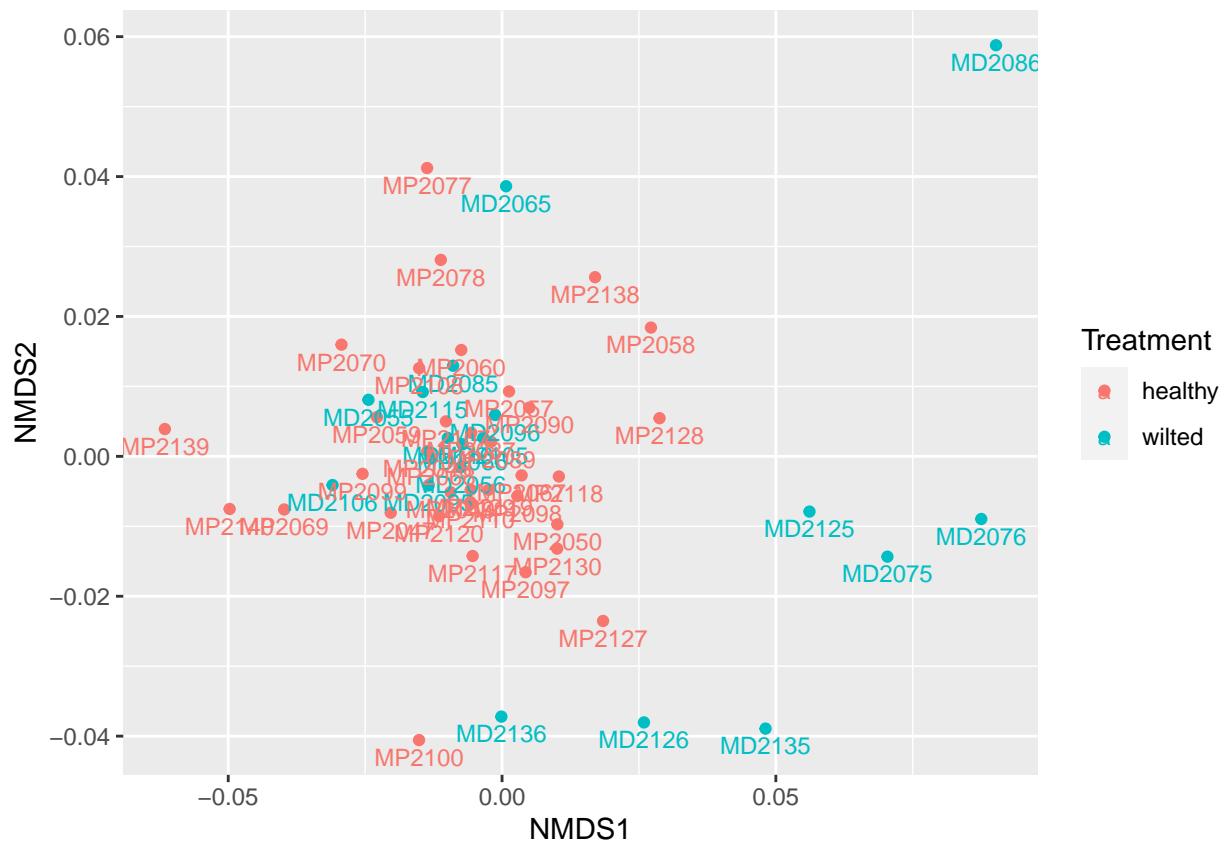
## Run 13 stress 0.1664769
## Run 14 stress 0.1640624
## Run 15 stress 0.1652946
## Run 16 stress 0.1552657
## Run 17 stress 0.1684183
## Run 18 stress 0.168103
## Run 19 stress 0.1694669
## Run 20 stress 0.1572811
## *** Best solution was not repeated -- monoMDS stopping criteria:
##     1: no. of iterations >= maxit
##     19: stress ratio > sratmax

```

```

plot_ordination(physeq = percentages_Bacteria_Phylum, ordination = meta_ord_Bacteria_Phylum, color = "Treatment"
geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@Data)), size = 3, vjust = 1.5)

```



```

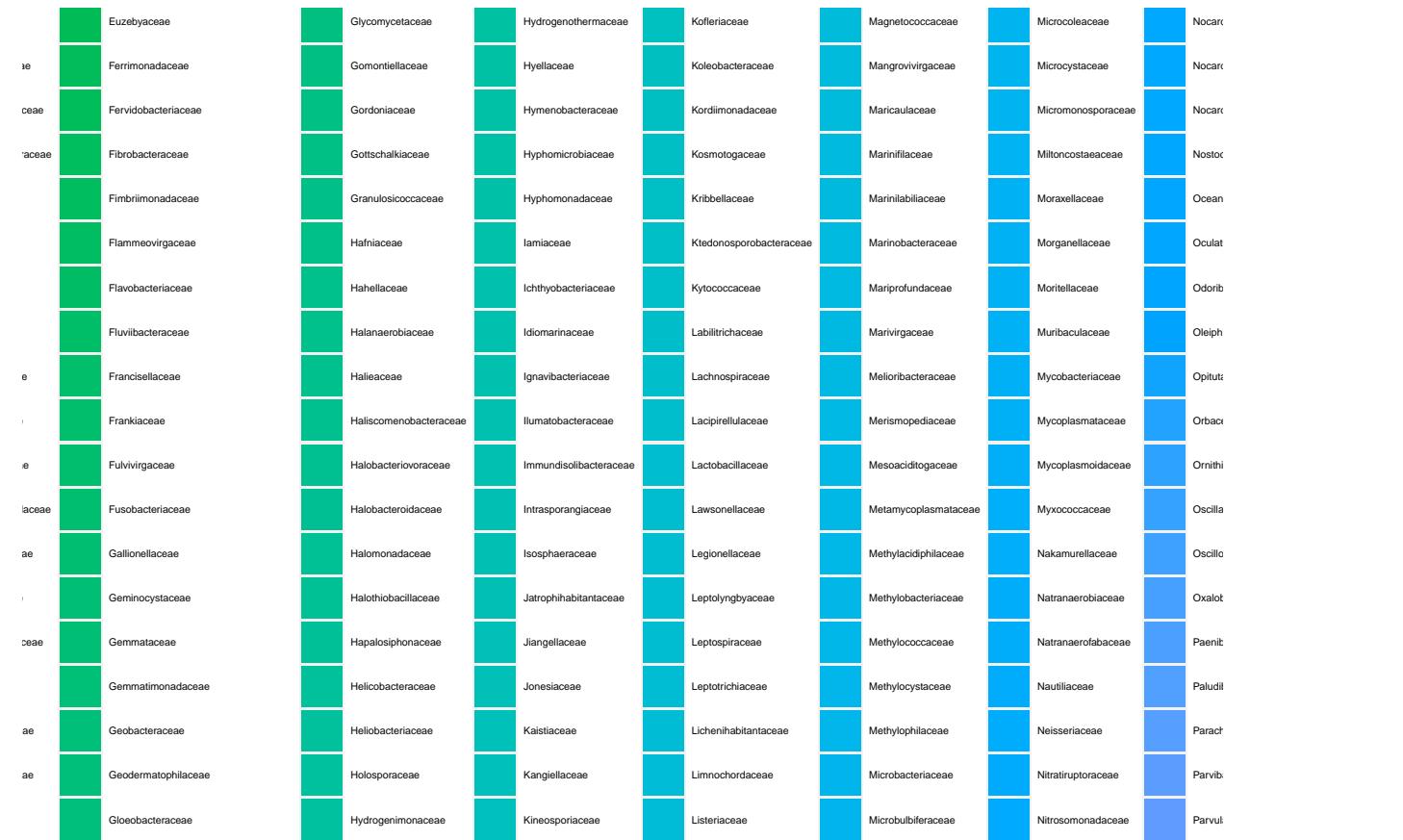
## Bacterias por Family

```

```

ggplot(data= percentages_Bacteria_df, aes(x=Sample, y=Abundance, fill=Family))+
  geom_bar(aes(), stat="identity", position="stack") +
  theme(text = element_text(size = 5),
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))

```



```
merge_Bacteria_Family<-tax_glom(merge_Bacteria,taxrank="Family")
```

sacamos las abundancias relativas

```
percentages_Bacteria_Family <- transform_sample_counts(merge_Bacteria_Family, function(x) x*100 / sum(x))
percentages_Bacteria_Family_df <- psmelt(percentages_Bacteria_Family)
meta ord Bacteria Family <- ordinate(phylseg = percentages_Bacteria_Family, method = "NMDS", distance =
```

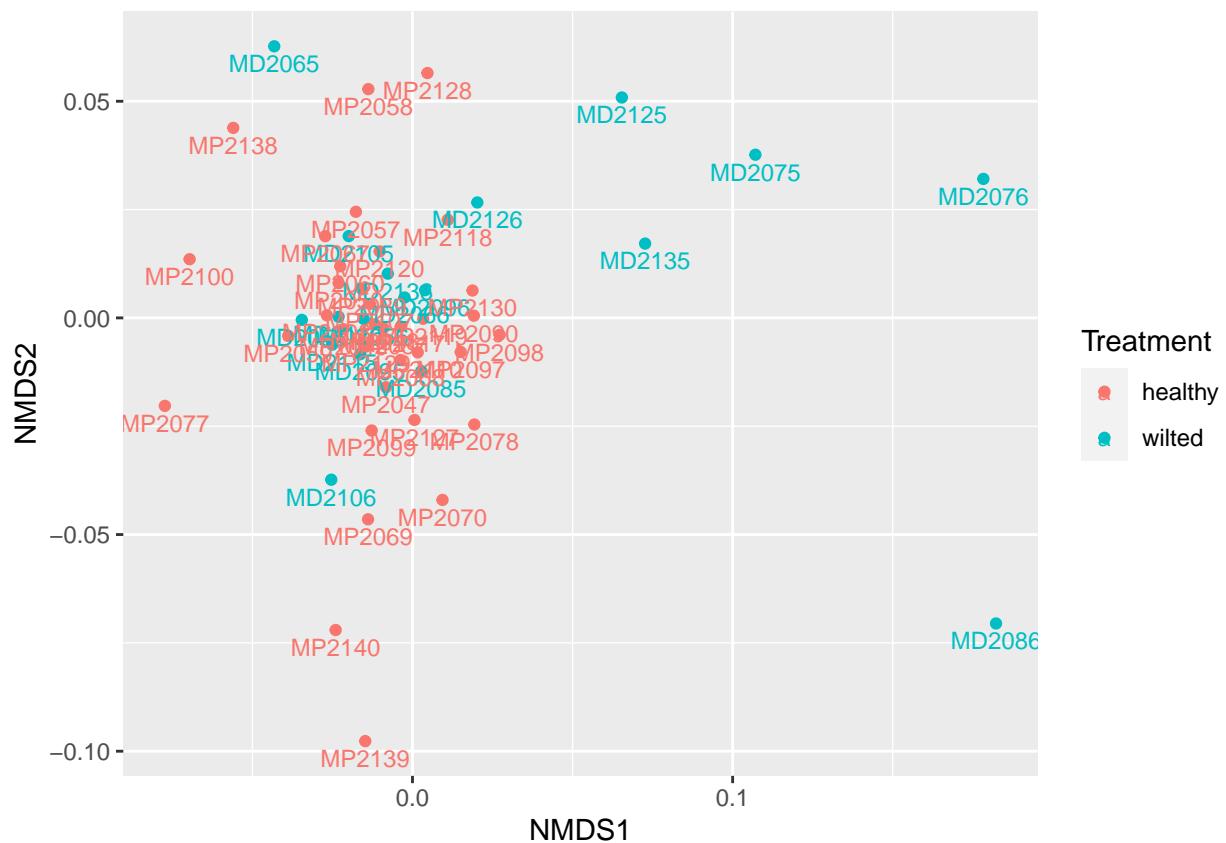
```
## Wisconsin double standardization
## Run 0 stress 0.1412073
## Run 1 stress 0.1400266
## ... New best solution
## ... Procrustes: rmse 0.02624436 max resid 0.1398476
## Run 2 stress 0.1522458
## Run 3 stress 0.1872704
## Run 4 stress 0.1487706
## Run 5 stress 0.1391561
## ... New best solution
## ... Procrustes: rmse 0.01336643 max resid 0.06805884
## Run 6 stress 0.1423484
## Run 7 stress 0.1390459
## ... New best solution
## ... Procrustes: rmse 0.003383345 max resid 0.01453331
```

```

## Run 8 stress 0.1521185
## Run 9 stress 0.1396541
## Run 10 stress 0.1857861
## Run 11 stress 0.1665445
## Run 12 stress 0.169478
## Run 13 stress 0.1412064
## Run 14 stress 0.1412064
## Run 15 stress 0.1558122
## Run 16 stress 0.1418146
## Run 17 stress 0.1369863
## ... New best solution
## ... Procrustes: rmse 0.06277167 max resid 0.3766657
## Run 18 stress 0.1400259
## Run 19 stress 0.1370376
## ... Procrustes: rmse 0.002435054 max resid 0.01204808
## Run 20 stress 0.1519216
## *** Best solution was not repeated -- monoMDS stopping criteria:
##     1: no. of iterations >= maxit
##     19: stress ratio > sratmax

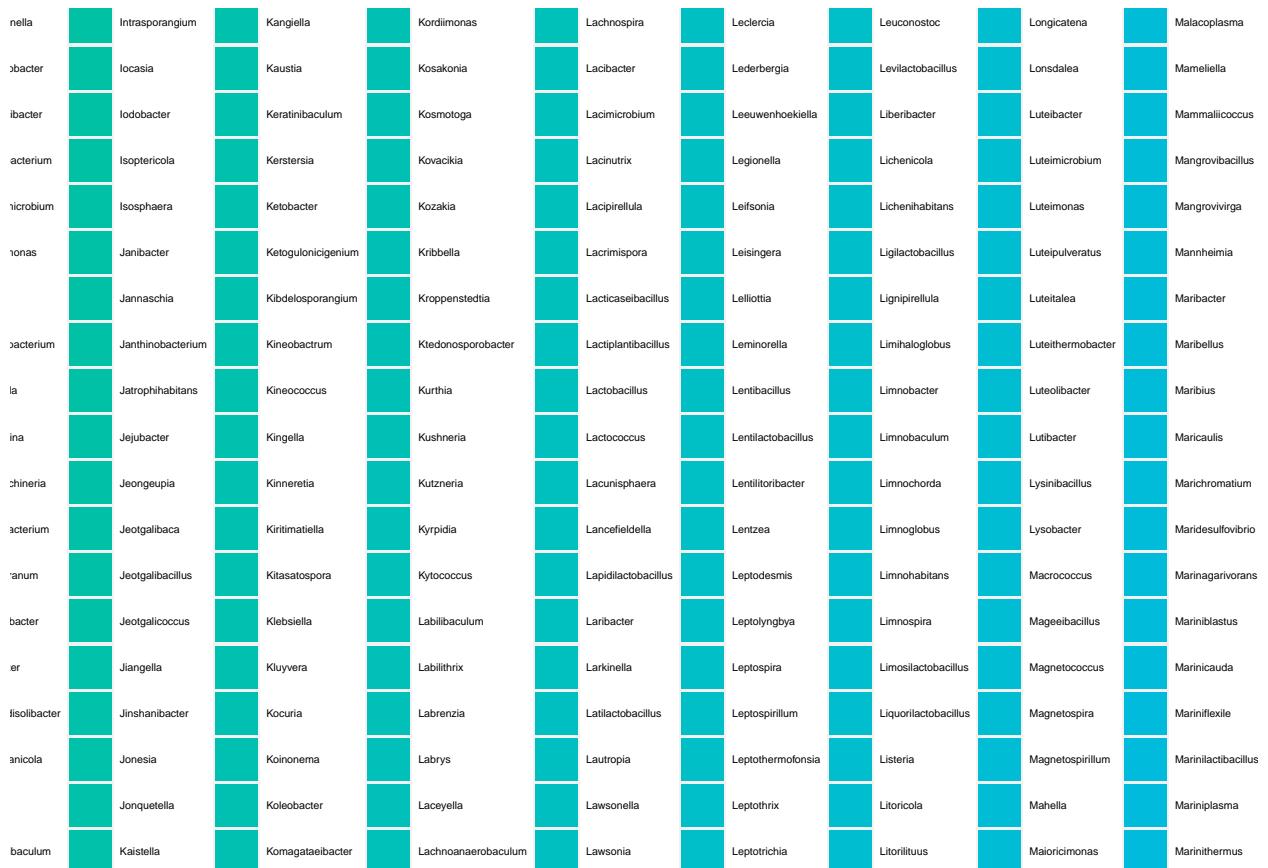
plot_ordination(physeq = percentages_Bacteria_Family, ordination = meta_ord_Bacteria_Family, color = "Treatment"
  geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@Data)), size = 3, vjust = 1.5)

```



```
## Bacterias por Genero
```

```
ggplot(data= percentages_Bacteria_df, aes(x=Sample, y=Abundance, fill=Genus))+  
  geom_bar(aes(), stat="identity", position="stack") +  
  theme(text = element_text(size = 5),  
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))
```



```
merge_Bacteria_Genus<-tax_glom(merge_Bacteria,taxrank="Genus")
```

sacamos las abundancias relativas

```
percentages_Bacteria_Genus <- transform_sample_counts(merge_Bacteria_Genus, function(x) x*100 / sum(x))  
percentages_Bacteria_Genus_df <- psmelt(percentages_Bacteria_Genus)  
meta_ord_Bacteria_Genus <- ordinate(physeq = percentages_Bacteria_Genus, method = "NMDS", distance = "bray")
```

```
## Wisconsin double standardization  
## Run 0 stress 0.1467945  
## Run 1 stress 0.1432842  
## ... New best solution  
## ... Procrustes: rmse 0.08877191 max resid 0.3282517  
## Run 2 stress 0.1438881  
## Run 3 stress 0.150125  
## Run 4 stress 0.1496766  
## Run 5 stress 0.1536577  
## Run 6 stress 0.147675  
## Run 7 stress 0.1512368
```

```

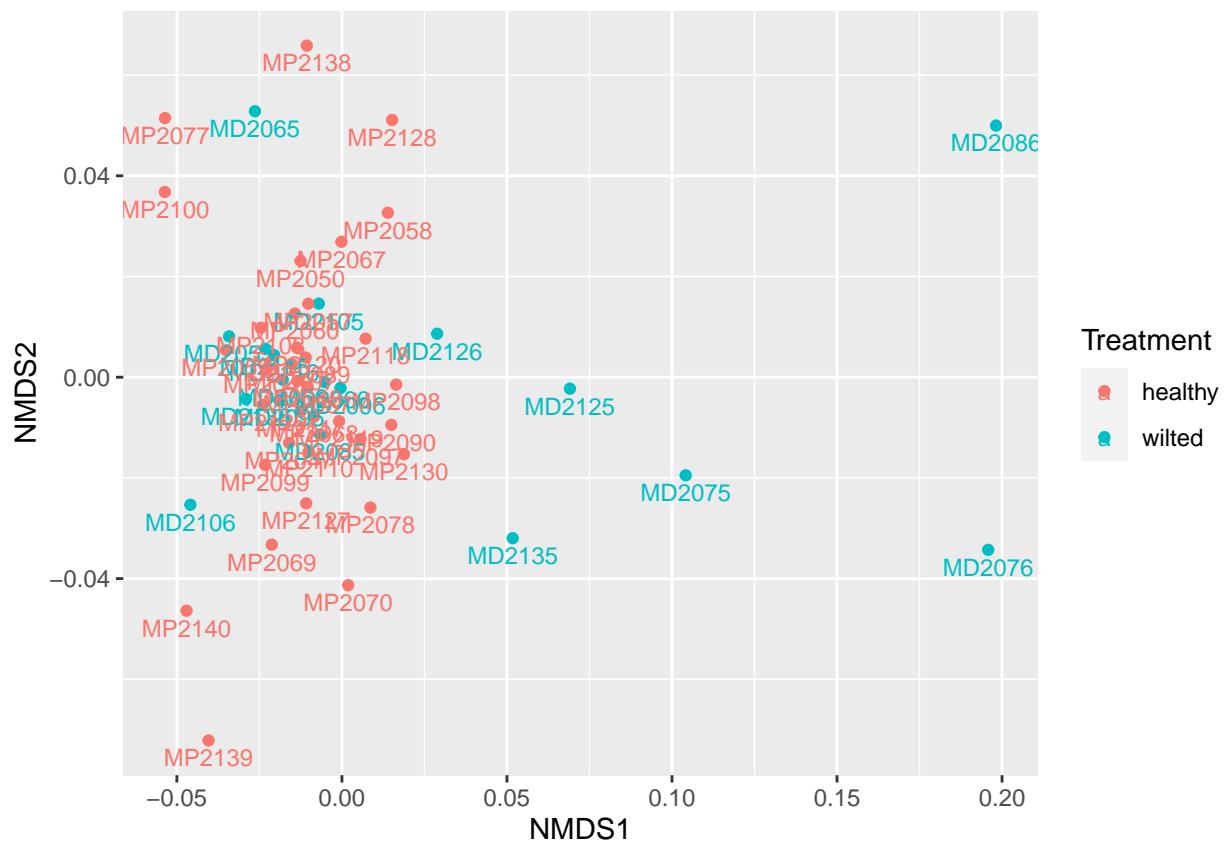
## Run 8 stress 0.1438189
## Run 9 stress 0.1454651
## Run 10 stress 0.1476414
## Run 11 stress 0.1432833
## ... New best solution
## ... Procrustes: rmse 0.0004003181 max resid 0.001692555
## ... Similar to previous best
## Run 12 stress 0.1477409
## Run 13 stress 0.1484467
## Run 14 stress 0.1454652
## Run 15 stress 0.1492423
## Run 16 stress 0.1438727
## Run 17 stress 0.1546134
## Run 18 stress 0.1454648
## Run 19 stress 0.1492682
## Run 20 stress 0.1476398
## *** Best solution repeated 1 times

```

```

plot_ordination(physeq = percentages_Bacteria_Genus, ordination = meta_ord_Bacteria_Genus, color = "Treatment"
  geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@Data))), size = 3, vjust = 1.5)

```



tomando diferentes porcentajes de abundancia

```

percentages_Bacteria_Genus_df$Genus[percentages_Bacteria_Genus_df$Abundance < 10.0] <- "Genus < 0.5% abundance"
percentages_Bacteria_Genus_df$Genus <- as.factor(percentages_Bacteria_Genus_df$Genus)

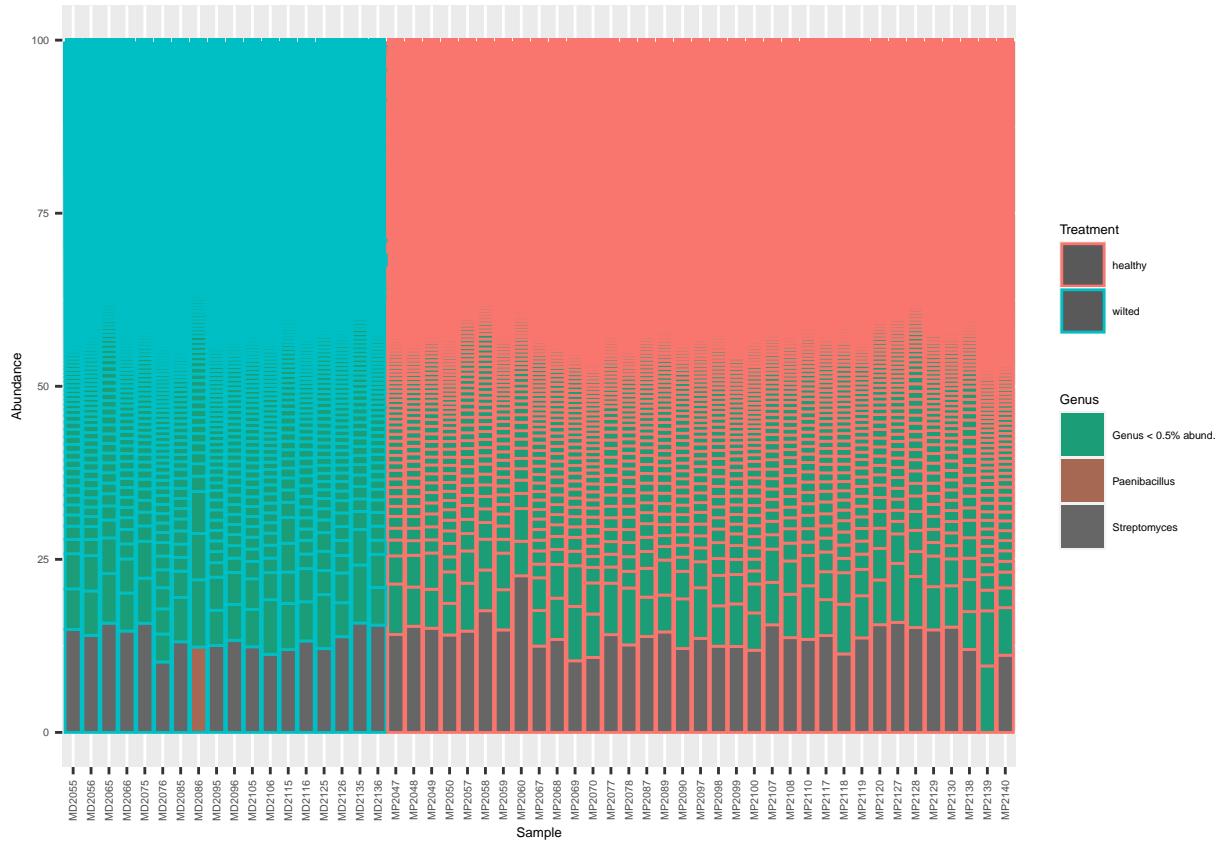
```

```

genus_colors_rel <- colorRampPalette(brewer.pal(8, "Dark2")) (length(levels(percentages_Bacteria_Genus_d
relative_plot <- ggplot(data=percentages_Bacteria_Genus_df, aes(x=Sample, y=Abundance, fill=Genus ,colo
  geom_bar(aes(), stat="identity", position="stack") +
  scale_fill_manual(values = genus_colors_rel) +
  theme(text = element_text(size = 5),
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))

relative_plot

```

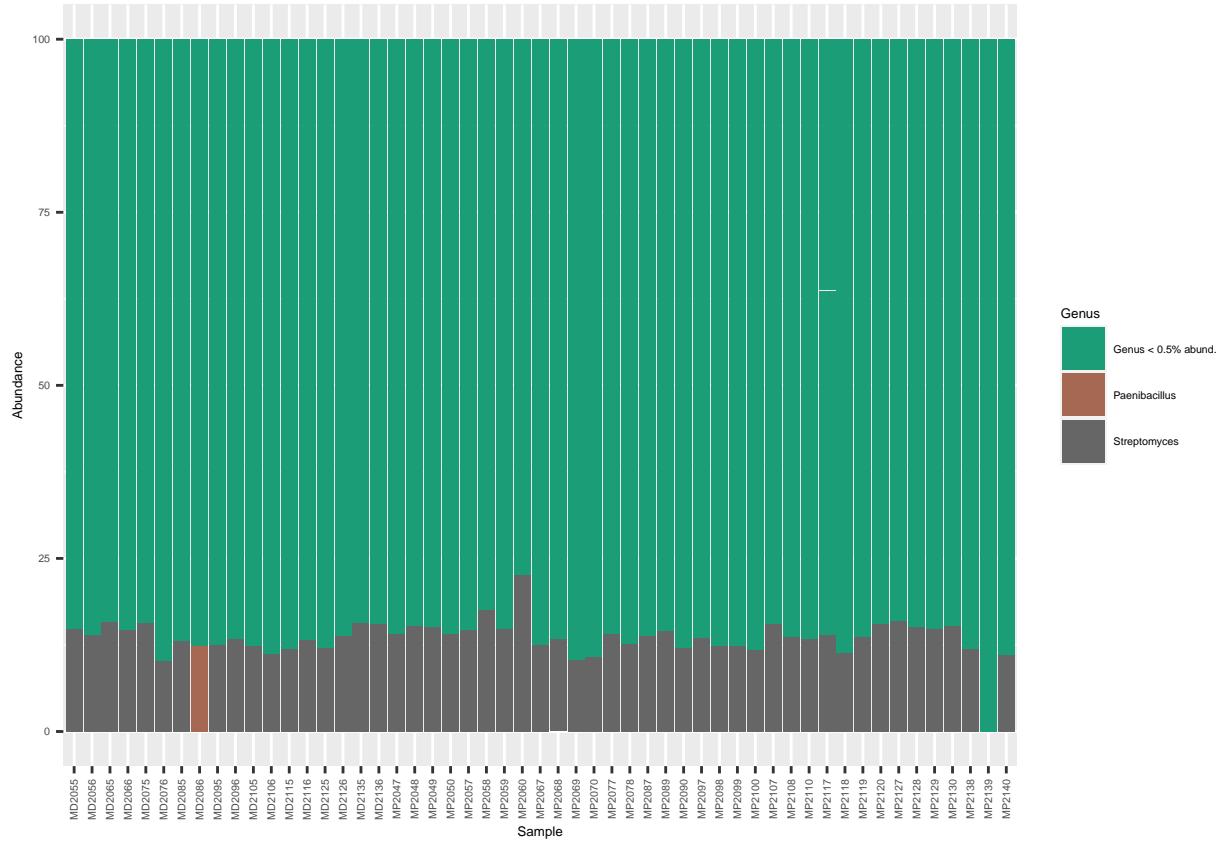


```

relative_plot <- ggplot(data=percentages_Bacteria_Genus_df, aes(x=Sample, y=Abundance, fill=Genus)) +
  geom_bar(aes(), stat="identity", position="stack") +
  scale_fill_manual(values = genus_colors_rel) +
  theme(text = element_text(size = 5),
        axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5))

relative_plot

```



sacando la beta diversidad con aglomerado de 10%

```
meta_ord_Bacteria_Genus <- ordinate(physeq = percentages_Bacteria_Genus, method = "NMDS", distance = "b
```

```
## Wisconsin double standardization
## Run 0 stress 0.1467945
## Run 1 stress 0.1481479
## Run 2 stress 0.1467449
## ... New best solution
## ... Procrustes: rmse 0.008104864 max resid 0.03191964
## Run 3 stress 0.1437855
## ... New best solution
## ... Procrustes: rmse 0.090568 max resid 0.3320095
## Run 4 stress 0.144925
## Run 5 stress 0.1432841
## ... New best solution
## ... Procrustes: rmse 0.01390617 max resid 0.08652475
## Run 6 stress 0.149082
## Run 7 stress 0.1458785
## Run 8 stress 0.1432835
## ... New best solution
## ... Procrustes: rmse 0.0009225133 max resid 0.003814088
## ... Similar to previous best
## Run 9 stress 0.1438148
## Run 10 stress 0.1461845
## Run 11 stress 0.1504894
```

```

## Run 12 stress 0.1495483
## Run 13 stress 0.1438165
## Run 14 stress 0.1500527
## Run 15 stress 0.1437859
## Run 16 stress 0.150125
## Run 17 stress 0.1490419
## Run 18 stress 0.1505679
## Run 19 stress 0.1438156
## Run 20 stress 0.1505678
## *** Best solution repeated 1 times

```

```

plot_ordination(physeq = percentages_Bacteria_Genus, ordination = meta_ord_Bacteria_Genus, color = "Treatment"
geom_text(mapping = aes(label = colnames(fresa_kraken_fil@otu_table@Data)), size = 3, vjust = 1.5)

```

