

# QIIME\_to\_R(phyloseq, microbiome)

Primero se necesitan cargar las librerías que vamos a utilizar. Es posible usar las librerías de dplyr y tidy, si no es posible instalar tidyverse

```
library(tidyverse)
library(phyloseq)
library(qiime2R)
library(vegan)
library(microbiome)
library(ggplot2)
library(cowplot)
```

Vegan, phyloseq y microbiome se pueden considerar como extensiones de las mismas.

Son muy útiles para tratar con datos de provenientes de la secuenciación de amplicones, tablas de otus, tablas de taxonomía y sus respectivos metadatos. El primer paso es importar las salida de qiime2 a un archivo tipo phyloseq, su estructura es una lista que contiene diferentes tablas con la información de cada muestra usada en qiime2.

Para esto se necesitan cuatro archivos provenientes de qiime:

La tabla de ASVs (table.qza generalmente),

El archivo de taxonomía

El meta data usado

EL árbol generado por qiime2 entre las muestras.

La forma mas fácil de importar los datos es con la librería qiime2R y el siguiente comando:

PARA ESTE COMANDO LOS ARCHIVOS TIENEN ESE NOMBRE, RECUERDA NO SOLO COPIAR Y PEGAR.

```
#setwd("")
#DEFINIMOS EL DIRECTORIO DE TRABAJO
Physeq_QIIME2 <- qza_to_phyloseq(features = "table.qza", tree =
"rootedtree.qza", metadata = "metadata.tsv", taxonomy = "taxonomy_blast.qza")
#En este caso estos archivos tiene ese nombre, RECUERDA ESCRIBIR EL NOMBRE DE
CADA ARCHIVO CON EL QUE TU LOS GUARDASTE SALIENDO DE QIIME2
```

La taxonomía de la base de datos de silva-ARB tiene una estructura que definida por lo que hay que manualmente declarar los niveles taxonómicos y cambiar la forma en que esta escrita la taxonomía .

```
tax <- data.frame(phyloseq::tax_table(Physeq_QIIME2)[, 1]) %>%
  mutate(Kingdom = stringr::str_replace_all(Kingdom, "D_\\d__", ""))
#Definimos la categorias para cada nivel taxonomico
tax <- tax %>% tidyr::separate(Kingdom, c("Kingdom", "Phylum", "Class", "Order",
"Family", "Genus", "Species"), sep = ";")
#Agregamos estos cambios al archivo Phyloseq
tax_matriz <- as.matrix(tax)
rownames(tax_matriz) <- phyloseq::taxa_names(Physeq_QIIME2)
phyloseq::tax_table(Physeq_QIIME2) <- tax_matriz
```

Con esto se tiene el archivo phyloseq de nuestras muestras procesadas por QIIME2.

También es posible generar manualmente el archivo phyloseq.

```
#Se importa cada parte del phyloseq manualmente
metadata<-read.table("metadata.tsv",
header=T, sep="\t",comment="", row.names=1)
metadata<-metadata[-1,]
tableASV<-read_qza("table.qza")
taxonomy<-read_qza("taxonomy_blast.qza")
taxtable<-taxonomy$data %>% as_tibble() %>% separate(Taxon, sep=";",
c("Kingdom", "Phylum", "Class", "Order", "Family", "Genus", "Species"))
taxtable<- taxtable %>% arrange(Feature.ID)
rooted_tree<-read_qza("rooted-tree.qza")
#Se arma la lista
physeq_QIIME2 <- phyloseq(
  otu_table(tableASV$data, taxa_are_rows = T),
  phy_tree(tree$data),
  tax_table(as.data.frame(taxtable) %>% select(-Confidence) %>%
column_to_rownames("Feature.ID") %>%
as.matrix()), sample_data(metadata) %>% as.data.frame() %>%
column_to_rownames("X.SampleID"))
```

## Curvas de rarefacción y métricas de alfa diversidad

```
#para genera curvas de rarefaccion se usa vegan
#tarda un poco dependiendo del numero de muestras y secuencias.

rarecurve(t(otu_table(Physeq_QIIME2)), step=50, cex=0.5,xlab = "Esfuerzo de
muestreo", ylab = "ASVs",)
#Se puede cambiar el tipo de linea y color de esta
```

Para generar las clásicas graficas de barras lo mejor es colapsar todas las muestras a un nivel taxonómico deseado, filtrar por abundancia y generar abundancias relativas antes de graficar.

```
#Por ejemplo a nivel de Orden
ORDER_PHYSEQ <- Physeq_QIIME2 %>% aggregate_taxa(level = "Order") %>%
core(detection = 3 ,prevalence = .1) %>% transform(transform = "compositional")

grafica <- plot_composition(ORDER_PHYSEQ,plot.type = "barplot")
#Quitamos la legenda para observar la grafica.
grafica + theme(legend.position = "none")
#Modificamos los colores
```

```

color = grDevices::colors()[grep('colores', grDevices::colors(), invert = T)]
set.seed(120)
color_b=c(sample(color, 54), "gray")
grafica + theme(legend.position = "none") + scale_fill_manual(values = color_b)

#Graficamos las legendas aparte

grafica <- grafica + scale_fill_manual(values = color_b)
legenda <- cowplot::get_legend(grafica)
plot(legenda)

#Tambien se pueden generar heatmaps.

plot_composition(ORDER_PHYSEQ, plot.type = "heatmap", sample.sort = "neatmap",
  otu.sort = "neatmap")

#Todas las imagenes pueden ser modificadas con parametreos de ggplot2

grafica + xlabel("") + ggtilte("") + geom_bar() + coord_polar() #etc

#DIVERSIDAD ALFA
#Calcular las metricas (todas)
tabla_alfa <- alpha(Physeq_QIIME2, index = "all")
table_alfa
#Indices de dominancia
dominance(Physeq_QIIME2, index = "all")
#OTUS DOMINANTES
dominant(Physeq_QIIME2)
plot_richness(Physeq_QIIME2, measures = c("Simpson", "Shannon")) +
geom_point(size=6)

```

## Diversidad Beta y "ordination"

La diversidad beta cuantifica la similitud o disimilitud entre las composiciones de microorganismos entre las muestras. En este caso se usara utilizando la métrica de diversidad beta de Bray-Curtia.

```

#Calcular las divergencias entre las muestra de suelo, sedimento y cultivo.
Cultivos <- divergence(subset_samples(Physeq_QIIME2, Type == "Culture"),
  apply(abundances(subset_samples(Physeq_QIIME2, Type ==
"Culture")), 1, median))
Sedimentos <- divergence(subset_samples(Physeq_QIIME2, Type == "Sediment"),
  apply(abundances(subset_samples(Physeq_QIIME2, Type ==
"Sediment")), 1, median))
Suelo <- divergence(subset_samples(cPhyseq_QIIME2, Type == "Soil"),
  apply(abundances(subset_samples(Physeq_QIIME2, Type ==
"Soil")), 1, median))

#Graficamos la tabla en donde el grupo con un valor mas grande tendra mas
heterogenidad.
boxplot(list(Sed = Sedimentos, Cult = Cultivos, Soil = Suelo)

```

La ordenación es un método para visualizar conjuntos de datos y encontrar agrupamientos, de modo que muestras similares estén cerca de otros en un plano de dos dimensiones. Estas relaciones se caracterizan numéricamente o gráficamente. Existen muchas técnicas de ordenación (PCA, NMDS, CCA ,DCA, etc.) así como diversas métricas para calcular las distancias.

En este caso generaremos dos gráficas de ordenación entre nuestras muestras :

```
#Primero transformamos y filtramos nuestros datos
Physeq_relcore <- core(Physeq_QIIME2, detection = .3, prevalence = 0.1) %>%
transform(transform = "compositional")
#Genramos la ordenacion con MDS
first_ordi <- ordinate(Physeq_relcore, "MDS", "bray")
#Graficamos tanto muestras, como taxones
plot_ordination(Physeq_relcore, first_ordi, color = "Type") +geom_point(size =
3)
plot_ordination(Physeq_relcore, first_ordi, color = "Phylum", type = "taxa")
+geom_point(size = 3)
#En este caso usaremos una metrica de distancias que toma en cuenta la filogenia
entre las ASVs, con esta graficamos tanto Phylas y muestras.
pseq.NMDS <- ordinate(Physeq_relcore, "NMDS", "wunifrac")
plot_ordination(Physeq_relcore, pseq.NMDS,
                type = "split", shape = "Type",
                color = "Phylum") + geom_point(size=3)
```