BIOM_Example

BIOM Example with Phyloseq Graphics

This notebook shows how to import the BIOM tables in your code and use them within the phyloseq package for taxonomical analysis and visualization.

```r
#After cloning the github repo with 'git clone https://github.com/LivGen/LMAT.git'
rm(list=ls())
#setwd("LMAT/Supplementary_Tools")
load("BIOM_concatenated.file.RData")
```

Lets check that all the tables got loaded with the right dimensions

```r
ls()
```

```
## [1] "OTU_RA"    "OTU_Reads" "Specie"    "Tax_Full"
```

```r
print(dim(OTU_Reads))
```

```
## [1] 13 25
```

```r
print(dim(OTU_RA))
```

```
## [1] 13 25
```

```r
print(dim(Specie))
```

```
## [1] 13  2
```

Now lets look at the headers of a couple of the data frames to get a sense of what information it has.

```r
head(OTU_Reads)
```

```
##        sntc1_S47 sntc2_S58 sntc3_S68 sntc4_S78 sntc5_S48 sntc6_S59
## 9606         309      1120       310       321       537       413
## 11269       1565      1883      2125      1691      1269      1201
## 186540       838       881      1992      1142      1514       943
## 186538       266       344       442       297       269       270
## 565995       361       408       340       357       195       232
## 186541        35       104        63        78        67        36
##        sntc7_S69 sntc8_S79 zptc1_S10 zptc2_S20 zs2-1_S41 zs2-2_S42
## 9606         592       355      1993      2261       610       453
## 11269       1131      1443       649       965      1019      1427
## 186540      1044       753       627       800       556       671
## 186538       288       255     14698     24825       227       307
## 565995       266       367       229       422       181       280
## 186541        44        63       110       106        72        80
##        zs2-3_S43 zs3-1_S31 zs3-2_S32 zs3-3_S33 zs4-1_S21 zs4-2_S22
## 9606         486       784      1220       766       658      1040
## 11269       1238      1258      1555      1210       827      1673
## 186540       499       756      1227       868       465      1241
## 186538       282       213       193       259       176       272
## 565995       177       202       347       201       253       316
## 186541        52        27        83        27        67        65
##        zs4-3_S23 zs5-1_S11 zs5-2_S12 zs5-3_S13 zs6-1_S1 zs6-2_S2 zs6-3_S3
## 9606        1509      1615       995       466     3453     9468     1581
## 11269       1462      1503      2048      1532     1859     1099     1314
## 186540      1800       718       782       763     1232      679      714
## 186538       292      4328      8584      2098    24057    16481    19106
## 565995       382       295       411       308      278      296      380
## 186541        45        40       103        26       48       67       66
```

```
head(Tax_Full)
```

```
##        superkingdom kingdom   phylum    class           order      family
## 9606      Eukaryota Metazoa Chordata Mammalia        Primates   Hominidae
## 11269       Viruses                           Mononegavirales Filoviridae
## 186540      Viruses                           Mononegavirales Filoviridae
## 186538      Viruses                           Mononegavirales Filoviridae
## 565995      Viruses                           Mononegavirales Filoviridae
## 186541      Viruses                           Mononegavirales Filoviridae
##                 genus                specie
## 9606             Homo          Homo sapiens
## 11269    Marburgvirus  Marburg marburgvirus
## 186540     Ebolavirus      Sudan ebolavirus
## 186538     Ebolavirus      Zaire ebolavirus
## 565995     Ebolavirus Bundibugyo ebolavirus
## 186541     Ebolavirus Tai Forest ebolavirus
```

Awesome! It seems like we sucessfully retrieve the sample information for each taxonomical identification. Of course there are other interesting information in the other fastsummaries such as counts per genus and counts unlabelled. But we will focus on reads at specie level.

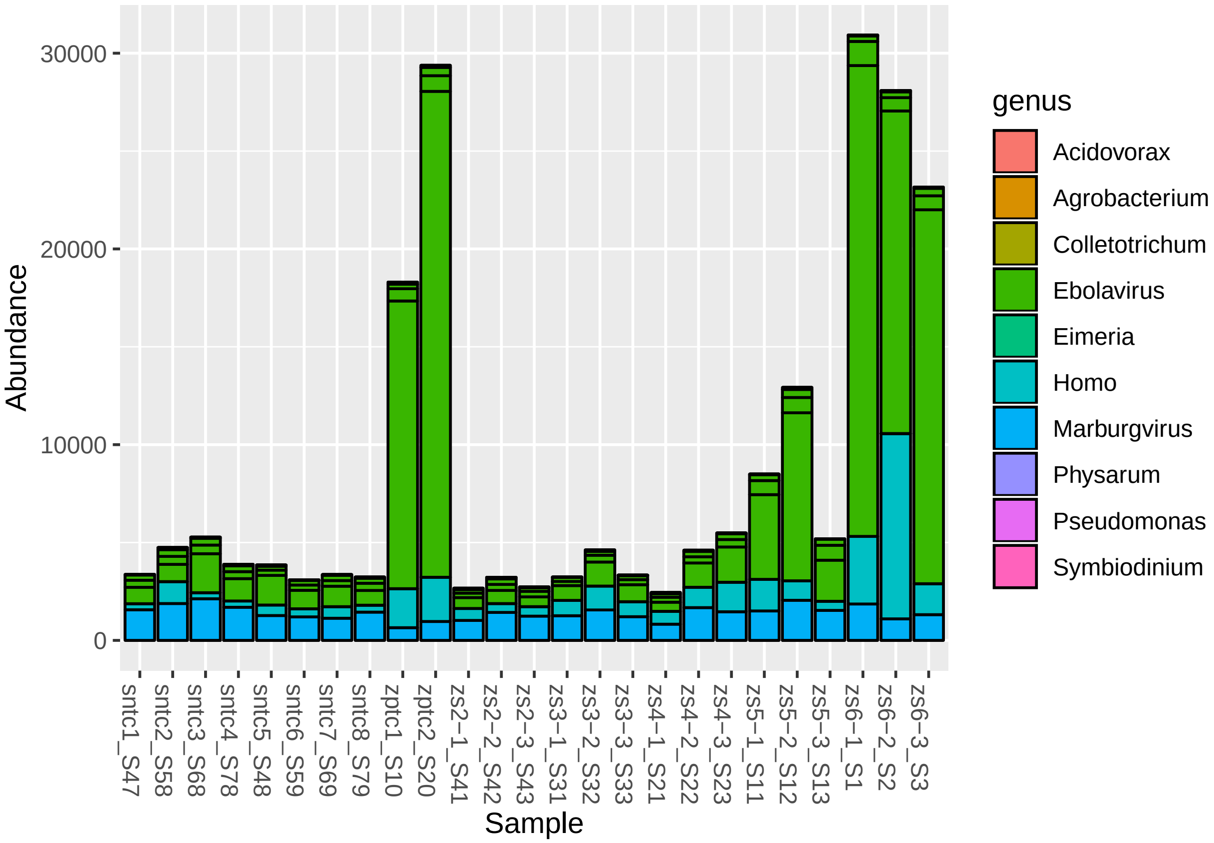
In this sample test, we have 13 taxonomical identifications at the specie level among all the 25 samples. Now lets explore this data withs some of the features in phyloseq. If you encounter installation errors please see this page: https://joey711.github.io/phyloseq/install.html

Some Vizualizations

Multiple merging, analysis and vizualizations of the data can be seen in the Tutorials section of the above mentioned page. Lets see the abundance representation per genus in each of the samples, in the *plot_bar()* method the parameter "fill=" helps to make that subsetting of the data.

```
#We have to create a phyloseq object
otu=otu_table(as.matrix(OTU_Reads),taxa_are_rows=T)
taxa=tax_table(as.matrix(Tax_Full))
physeq=phyloseq(otu,taxa)


#Lets see the abundance representation per genus in each of the samples per
plot_bar(physeq,fill="genus")
```



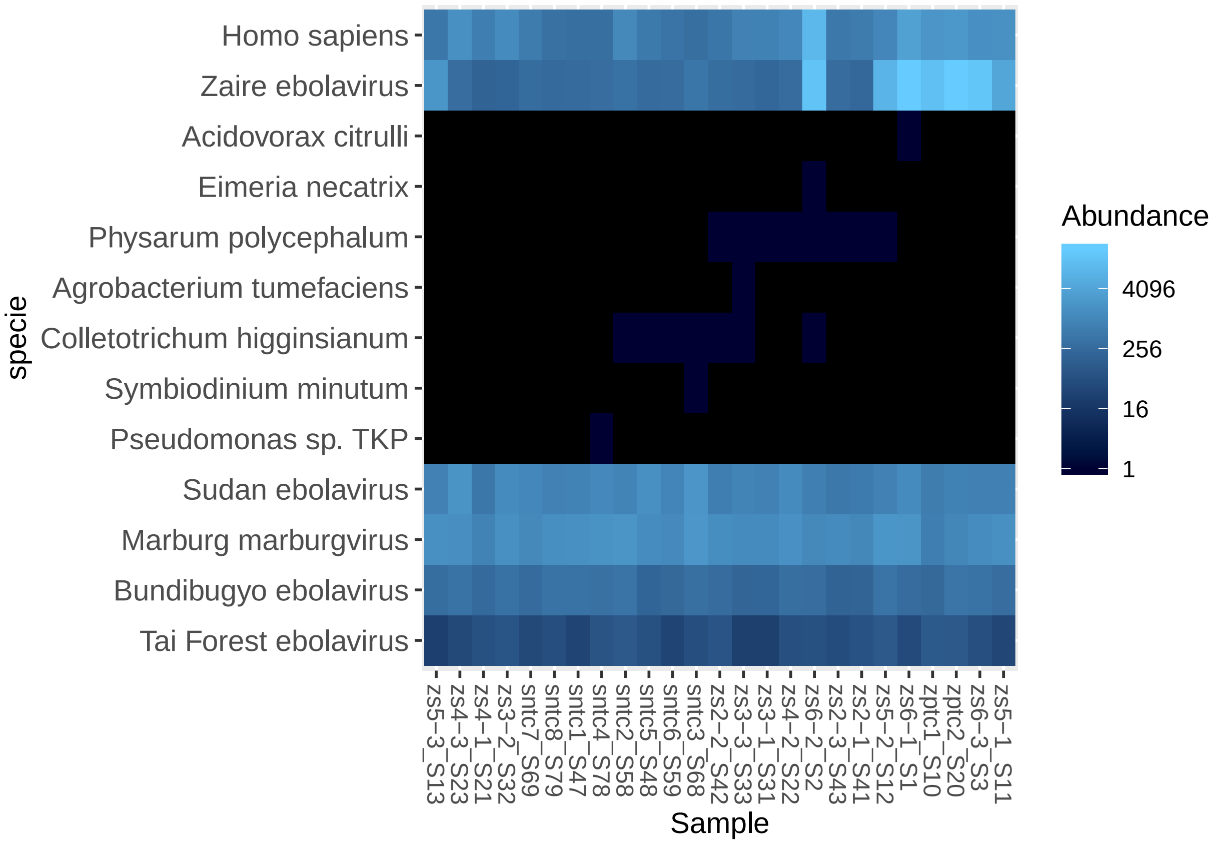
Above we can see that majority of the composition of our dataset is based on Human and Ebolavirus reads, this can give us the insight that these samples are of human individuals infected with some type of virus.

Heatmap

Now lets look at intensities of abundance of reads using a heatmap using a plot_heatmap()

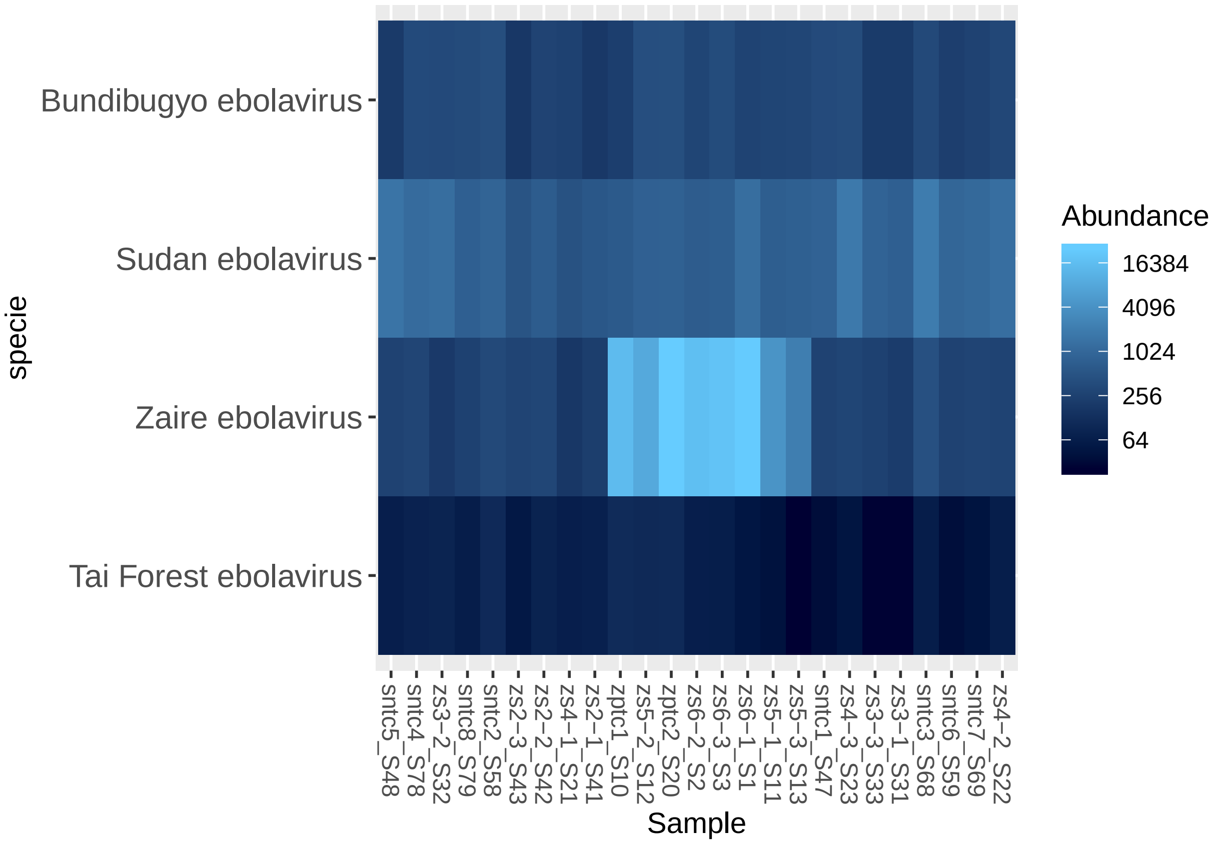
```
plot_heatmap(physeq,taxa.label = "specie")
```

```
## Warning: Transformation introduced infinite values in discrete y-axis
```



Non ebolavirus species have a low abundance in these samples [DARK color]. We can prune these reads out using subset_taxa() to only have “Ebolavirus reads” or prune_samples() if we want our sample at least certain amount of reads.

```
physeq1=subset_taxa(physeq,genus=="Ebolavirus")
plot_heatmap(physeq1,taxa.label = "specie")
```



Much better, here we can see that samples“zptc and zs6” have the highest abundance of Zaire ebolavirus.

Hierarchical Clustering

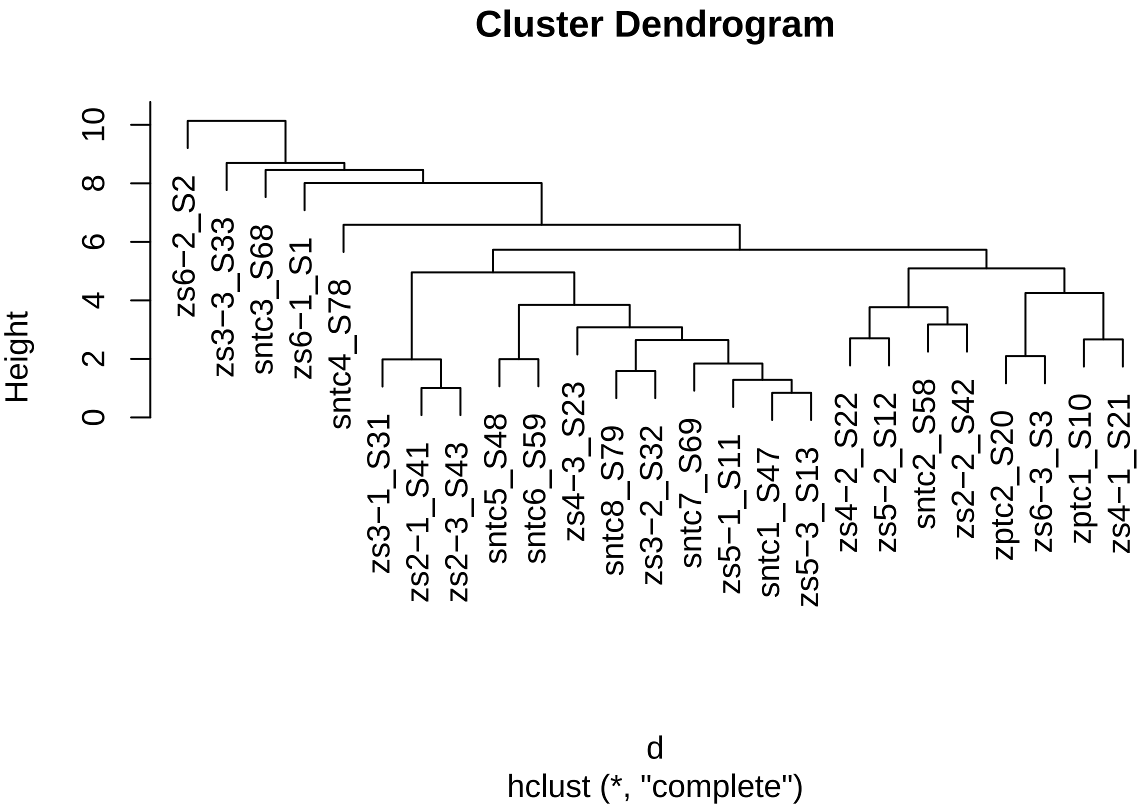
Although I have not provided any type of information about this dataset or the relationships of each of the samples, we can start by applying a Hierarchical Clustering algorithm to get an insight on how these samples may be related to one another, perhaps they are cases/control as we may asume by the above heatmap.

I have chosen to implement complete linkage, to account for distances in between clusters by looking at the furthest member of the cluster.

```r
#Lets call the HC package
#library(hclust)

#Lets use the above OTU_Reads matrx

data<-scale(t(as.matrix(OTU_Reads)))
d<-dist(data,method="euclidean")
hc=hclust(d,method="complete")
plot(hc)
```



Above we can see some relationships between samples, one relationship that comes out is of sample zs6-3 and zptc2, which had an abundance of zaire ebolavirus.

Shannon Entropy

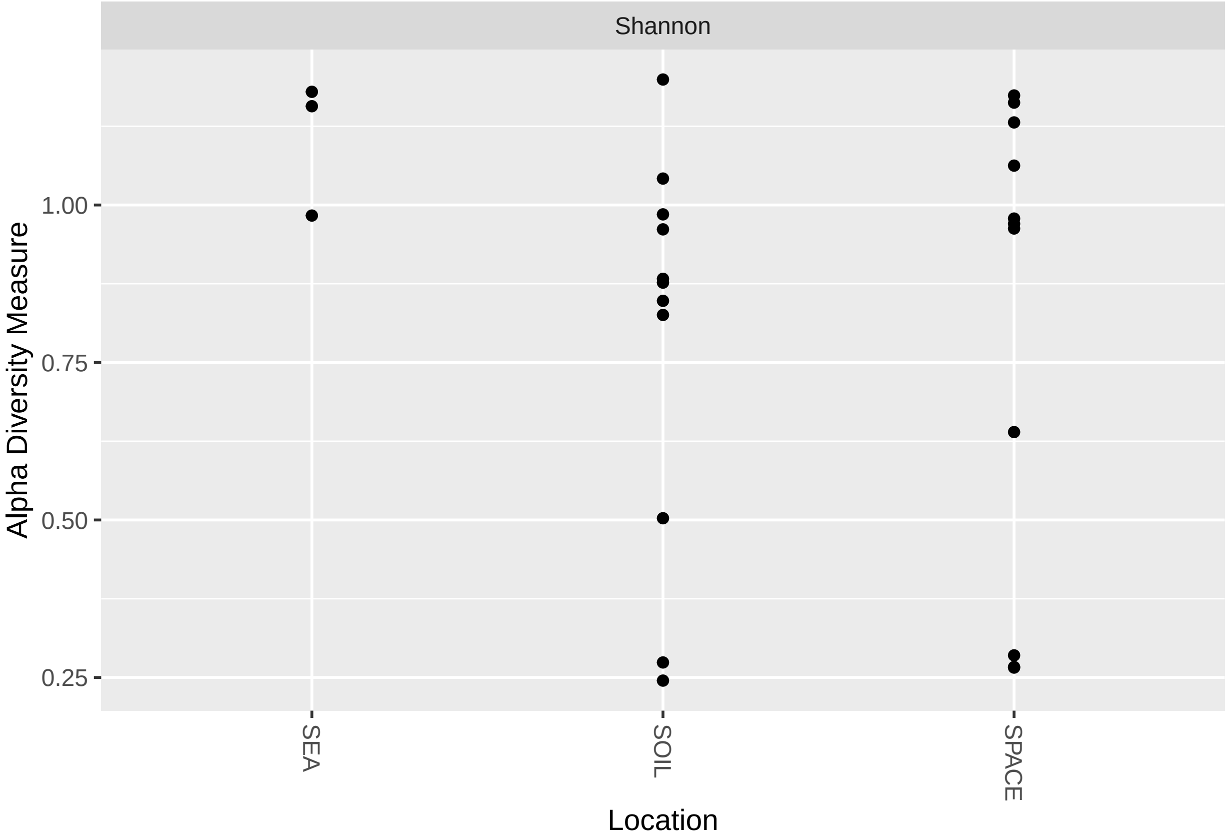
There are many more vizualizations that can be done with phyloseq, you can add sample information, to denote if your samples come from a different origin. Lets calculate the Shannon Entropy as diversity measure, for the sample type. The more diverse the more entropy.

```
#Lets create some origins/sample type for our 13 samples.
types=c("SOIL","SPACE","SEA")
sample_type=sample(types,length(sample_names(physeq1)),replace=T)

#Lets create a data frame to store this information
source_sample=data.frame(Location=sample_type,row.names=sample_names(physeq1))

#Add this extra information to the phyloseq object
sample_data(physeq1)=source_sample
plot_richness(physeq1,x="Location",measures = "Shannon")
```

```
## Warning in estimate_richness(physeq, split = TRUE, measures = measures): The data you have provided
## 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.
```



This concludes the example.