MB590 Microbiome Analysis - Alpha Diversity

Christine Hawkes

2/9/2022

Contents

References:

Data reference: Lozupone & Knight (2007) PNAS 104:11436-11440 doi.org/10.1073/pnas.0611525104

Setup

Load/install packages

```
library(tidyverse)
library(phyloseq)
library(vegan)
library(ggplot2)

# install.packages("remotes")
# remotes::install_github("twbattaglia/btools")
library(btools)

# install.packages("ggpubr")
library(ggpubr)
```

Load and explore data

```
# call preloaded data with the phyloseq package - ps object with otu table, taxa table, sample data, &
data(GlobalPatterns)

# explore the data
phyloseq::ntaxa(GlobalPatterns)
```

```
## [1] 19216
```

```
phyloseq::nsamples(GlobalPatterns)
```

```
## [1] 26
```

```
phyloseq::sample_names(GlobalPatterns)
```

```
##  [1] "CL3"      "CC1"      "SV1"      "M31Fcsw"  "M11Fcsw"  "M31Plmr"
##  [7] "M11Plmr"  "F21Plmr"  "M31Tong"  "M11Tong"  "LMEpi24M" "SLEpi20M"
## [13] "AQC1cm"   "AQC4cm"   "AQC7cm"   "NP2"      "NP3"      "NP5"
## [19] "TRRsed1"  "TRRsed2"  "TRRsed3"  "TS28"     "TS29"     "Even1"
## [25] "Even2"    "Even3"
```

```
phyloseq::sample_data(GlobalPatterns)
```

```
##          X.SampleID  Primer Final_Barcode Barcode_truncated_plus_T
## CL3             CL3 ILBC_01        AACGCA                   TGCGTT
## CC1             CC1 ILBC_02        AACTCG                   CGAGTT
## SV1             SV1 ILBC_03        AACTGT                   ACAGTT
## M31Fcsw     M31Fcsw ILBC_04        AAGAGA                   TCTCTT
## M11Fcsw     M11Fcsw ILBC_05        AAGCTG                   CAGCTT
## M31Plmr     M31Plmr ILBC_07        AATCGT                   ACGATT
```

```
## M11Plmr     M11Plmr ILBC_08        ACACAC                  GTGTGT
## F21Plmr     F21Plmr ILBC_09        ACACAT                  ATGTGT
## M31Tong     M31Tong ILBC_10        ACACGA                  TCGTGT
## M11Tong     M11Tong ILBC_11        ACACGG                  CCGTGT
## LMEpi24M   LMEpi24M ILBC_13        ACACTG                  CAGTGT
## SLEpi20M   SLEpi20M ILBC_15        ACAGAG                  CTCTGT
## AQC1cm       AQC1cm ILBC_16        ACAGCA                  TGCTGT
## AQC4cm       AQC4cm ILBC_17        ACAGCT                  AGCTGT
## AQC7cm       AQC7cm ILBC_18        ACAGTG                  CACTGT
## NP2             NP2 ILBC_19        ACAGTT                  AACTGT
## NP3             NP3 ILBC_20        ACATCA                  TGATGT
## NP5             NP5 ILBC_21        ACATGA                  TCATGT
## TRRsed1     TRRsed1 ILBC_22        ACATGT                  ACATGT
## TRRsed2     TRRsed2 ILBC_23        ACATTC                  GAATGT
## TRRsed3     TRRsed3 ILBC_24        ACCACA                  TGTGGT
## TS28           TS28 ILBC_25        ACCAGA                  TCTGGT
## TS29           TS29 ILBC_26        ACCAGC                  GCTGGT
## Even1         Even1 ILBC_27        ACCGCA                  TGCGGT
## Even2         Even2 ILBC_28        ACCTCG                  CGAGGT
## Even3         Even3 ILBC_29        ACCTGT                  ACAGGT
##          Barcode_full_length         SampleType
## CL3              CTAGCGTGCGT               Soil
## CC1              CATCGACGAGT               Soil
## SV1              GTACGCACAGT               Soil
## M31Fcsw          TCGACATCTCT              Feces
## M11Fcsw          CGACTGCAGCT              Feces
## M31Plmr          CGAGTCACGAT               Skin
## M11Plmr          GCCATAGTGTG               Skin
## F21Plmr          GTAGACATGTG               Skin
## M31Tong          TGTGGCTCGTG             Tongue
## M11Tong          TAGACACCGTG             Tongue
## LMEpi24M         CATGAACAGTG         Freshwater
## SLEpi20M         AGCCGACTCTG         Freshwater
## AQC1cm           GACCACTGCTG Freshwater (creek)
## AQC4cm           CAAGCTAGCTG Freshwater (creek)
## AQC7cm           ATGAAGCACTG Freshwater (creek)
## NP2              TCGCGCAACTG              Ocean
## NP3              GCTAAGTGATG              Ocean
## NP5              GAACGATCATG              Ocean
## TRRsed1          CACGTGACATG Sediment (estuary)
## TRRsed2          TGCGCTGAATG Sediment (estuary)
## TRRsed3          GATGTATGTGG Sediment (estuary)
## TS28             GCATCGTCTGG              Feces
## TS29             CTAGTCGCTGG              Feces
## Even1            TGACTCTGCGG               Mock
## Even2            TCTGATCGAGG               Mock
## Even3            AGAGAGACAGG               Mock
##                                          Description
## CL3          Calhoun South Carolina Pine soil, pH 4.9
## CC1          Cedar Creek Minnesota, grassland, pH 6.1
## SV1        Sevilleta new Mexico, desert scrub, pH 8.3
## M31Fcsw       M3, Day 1, fecal swab, whole body study
## M11Fcsw      M1, Day 1, fecal swab, whole body study
## M31Plmr       M3, Day 1, right palm, whole body study
```

```
## M11Plmr      M1, Day 1, right palm, whole body study
## F21Plmr     F1, Day 1,  right palm, whole body study
## M31Tong          M3, Day 1, tongue, whole body study
## M11Tong          M1, Day 1, tongue, whole body study
## LMEpi24M Lake Mendota Minnesota, 24 meter epilimnion
## SLEpi20M Sparkling Lake Wisconsin, 20 meter eplimnion
## AQC1cm                   Allequash Creek, 0-1cm depth
## AQC4cm                  Allequash Creek, 3-4 cm depth
## AQC7cm                  Allequash Creek, 6-7 cm depth
## NP2            Newport Pier, CA surface water, Time 1
## NP3            Newport Pier, CA surface water, Time 2
## NP5            Newport Pier, CA surface water, Time 3
## TRRsed1                Tijuana River Reserve, depth 1
## TRRsed2                Tijuana River Reserve, depth 2
## TRRsed3                Tijuana River Reserve, depth 2
## TS28                                         Twin #1
## TS29                                         Twin #2
## Even1                                          Even1
## Even2                                          Even2
## Even3                                          Even3
```

```
phyloseq::rank_names(GlobalPatterns)
```

```
## [1] "Kingdom" "Phylum"  "Class"   "Order"   "Family"  "Genus"   "Species"
```

```
phyloseq::sample_variables(GlobalPatterns)
```

```
## [1] "X.SampleID"               "Primer"
## [3] "Final_Barcode"            "Barcode_truncated_plus_T"
## [5] "Barcode_full_length"      "SampleType"
## [7] "Description"
```

```
phyloseq::sample_sums(GlobalPatterns)
```

```
##      CL3      CC1      SV1  M31Fcsw  M11Fcsw  M31Plmr  M11Plmr  F21Plmr
##   864077  1135457   697509  1543451  2076476   718943   433894   186297
##  M31Tong  M11Tong LMEpi24M SLEpi20M   AQC1cm   AQC4cm   AQC7cm      NP2
##  2000402   100187  2117592  1217312  1167748  2357181  1699293   523634
##      NP3      NP5  TRRsed1  TRRsed2  TRRsed3     TS28     TS29    Even1
##  1478965  1652754    58688   493126   279704   937466  1211071  1216137
##    Even2    Even3
##   971073  1078241
```

Subset Data in Phyloseq

After subsetting, consider deleting the original from your Environment if your laptop has limited memory (use `rm` to remove files).

Delete mocks

```
# make a copy of the original ps object
ps_gp <- GlobalPatterns
ps_gp
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 19216 taxa and 26 samples ]
## sample_data() Sample Data:       [ 26 samples by 7 sample variables ]
## tax_table()   Taxonomy Table:    [ 19216 taxa by 7 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 19216 tips and 19215 internal nodes ]
```

```
# delete Mock samples (Sample type = Mock or Sample names are Even1-3)
ps_gp <-phyloseq::subset_samples(ps_gp, SampleType != "Mock")
ps_gp
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 19216 taxa and 23 samples ]
## sample_data() Sample Data:       [ 23 samples by 7 sample variables ]
## tax_table()   Taxonomy Table:    [ 19216 taxa by 7 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 19216 tips and 19215 internal nodes ]
```

```
phyloseq::sample_names(ps_gp) #check that "Even" samples are gone
```

```
##  [1] "CL3"      "CC1"      "SV1"      "M31Fcsw"  "M11Fcsw"  "M31Plmr"
##  [7] "M11Plmr"  "F21Plmr"  "M31Tong"  "M11Tong"  "LMEpi24M" "SLEpi20M"
## [13] "AQC1cm"   "AQC4cm"   "AQC7cm"   "NP2"      "NP3"      "NP5"
## [19] "TRRsed1"  "TRRsed2"  "TRRsed3"  "TS28"     "TS29"
```

```
phyloseq::nsamples(ps_gp) #check that there are now only 23 samples
```

```
## [1] 23
```

Subset to Bacteria only

```
phyloseq::get_taxa_unique(ps_gp, taxonomic.rank="Kingdom") # check Kingdoms names
```

```
## [1] "Archaea"  "Bacteria"
```

```
phyloseq::ntaxa(ps_gp)
```

```
## [1] 19216
```

```
ps_gp_bact <- phyloseq::subset_taxa(ps_gp, Kingdom=="Bacteria")
phyloseq::ntaxa(ps_gp_bact)
```

```
## [1] 19008
```

Remove singleton taxa and samples with zero sums

```
ps_gp_bact <- phyloseq::prune_taxa(taxa_sums(ps_gp_bact) > 1, ps_gp_bact)
phyloseq::ntaxa(ps_gp_bact)
```

```
## [1] 16432
```

```
phyloseq::sample_sums(ps_gp_bact)
```

```
##      CL3      CC1      SV1  M31Fcsw  M11Fcsw  M31Plmr  M11Plmr  F21Plmr
##   862627  1134016   668539  1543312  2076317   718817   433710   186119
##  M31Tong  M11Tong LMEpi24M SLEpi20M   AQC1cm   AQC4cm   AQC7cm      NP2
##  2000055   100150  2117394  1217135  1163223  2332373  1671138   521777
##      NP3      NP5  TRRsed1  TRRsed2  TRRsed3     TS28     TS29
##  1435669  1618666    57792   484536   265325   935780  1209289
```

```
ps_gp_bact <- phyloseq::prune_samples(sample_sums(ps_gp_bact)>0, ps_gp_bact)
phyloseq::nsamples(ps_gp_bact)
```

```
## [1] 23
```

Limit data to top 100 taxa based on abundance

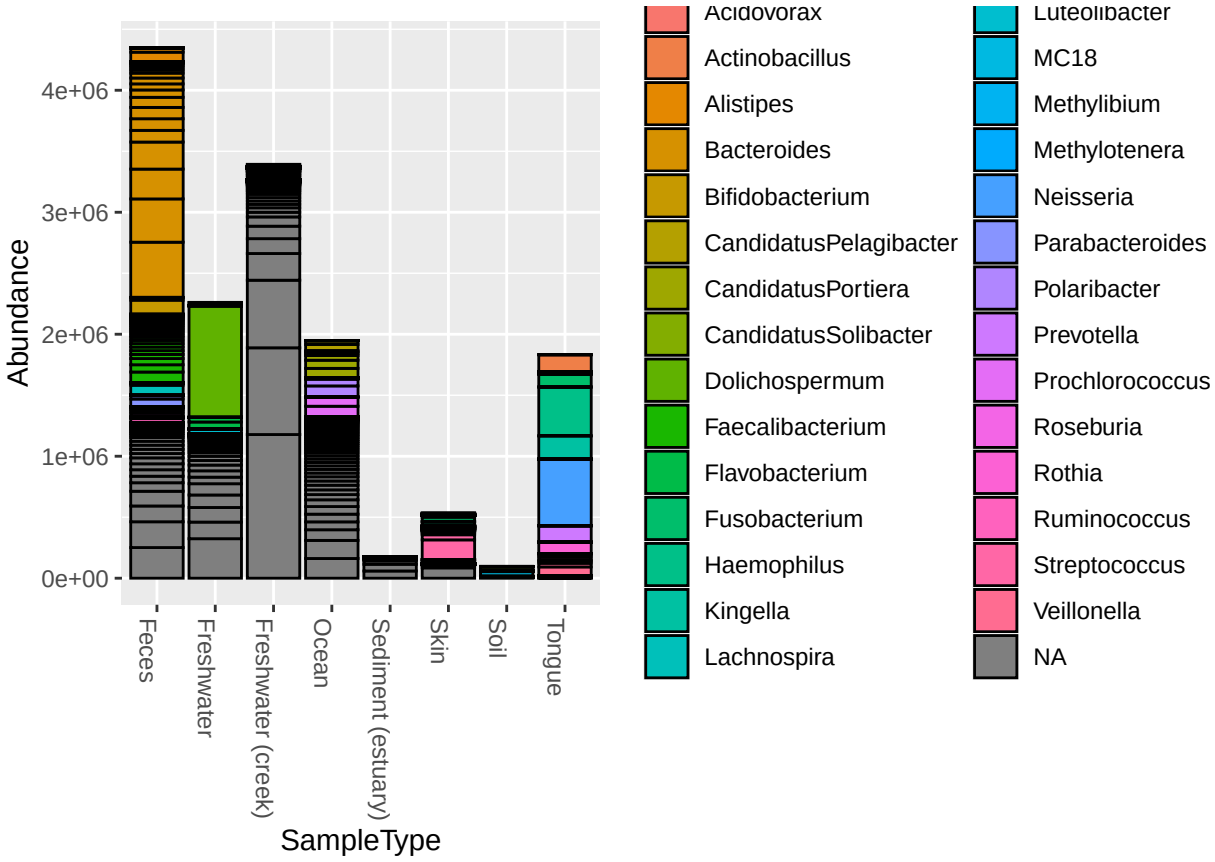
```
# if your laptop is very slow, try trimming further to top 100 taxa
# note that this subset will not accurately represent overall GP diversity
top100 <-names(sort(phyloseq::taxa_sums(ps_gp_bact), decreasing=TRUE)) [1:100]
ps_gp_top100 <- phyloseq::prune_taxa(top100, ps_gp_bact)
phyloseq::ntaxa(ps_gp_top100)
```

```
## [1] 100
```

```
phyloseq::sample_sums(ps_gp_top100) #check that there are no zero samples
```

```
##      CL3      CC1      SV1  M31Fcsw  M11Fcsw  M31Plmr  M11Plmr  F21Plmr
##    44764    48214     3975  1061411  1747789   347461   123750    62271
##  M31Tong  M11Tong LMEpi24M SLEpi20M   AQC1cm   AQC4cm   AQC7cm      NP2
##  1757942    75387  1563626   697776   797234  1610632   984999   280223
##      NP3      NP5  TRRsed1  TRRsed2  TRRsed3     TS28     TS29
##   783531   885273     8482   127231    40525   596747   941428
```

```
phyloseq::plot_bar(ps_gp_top100, "SampleType", fill = "Genus")
```



Alpha Diversity

Observed richness and diversity indices

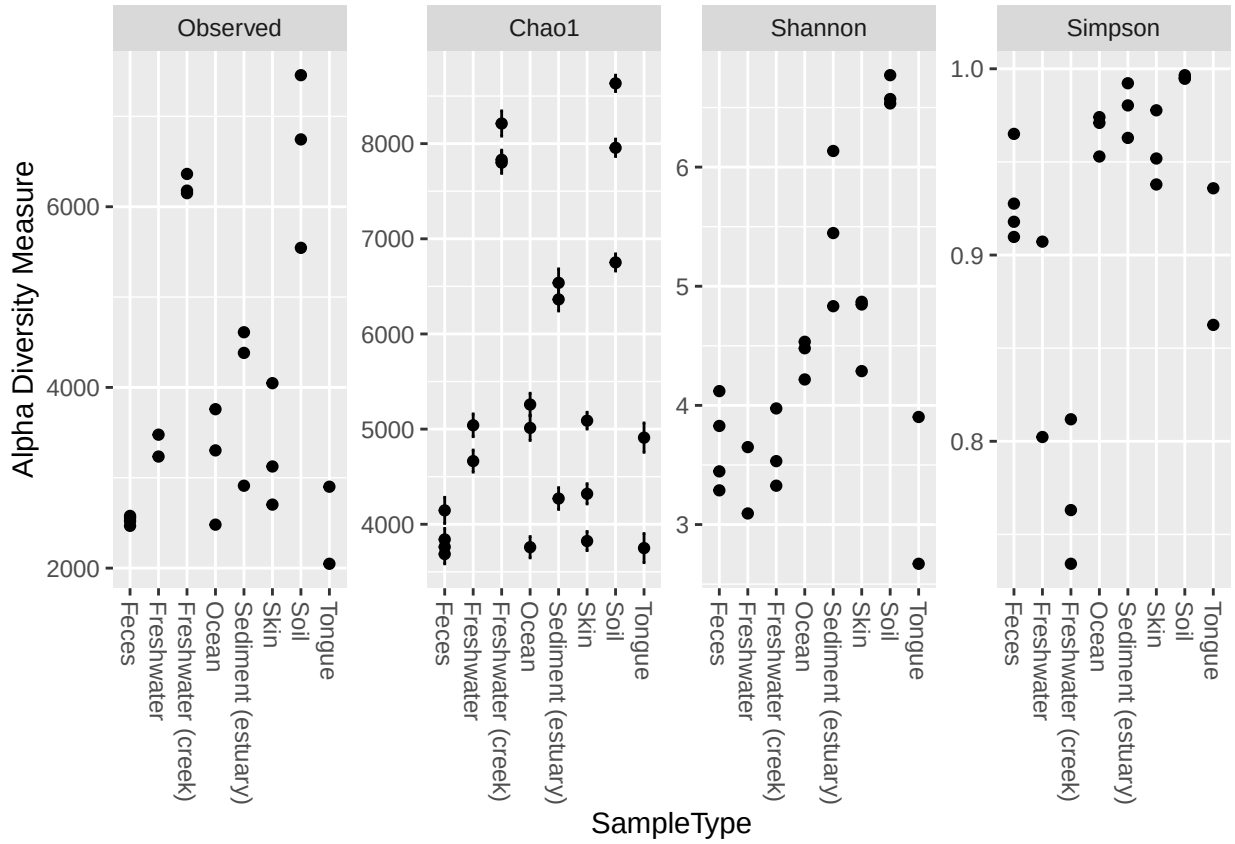
```
# richness calcs are typically made on untrimmed data
alpha <- phyloseq::estimate_richness(ps_gp_bact, split=TRUE, measures=c("Observed", "Chao1", "Shannon",
alpha
```

```
##          Observed    Chao1  se.chao1  Shannon   Simpson
## CL3          6745 7956.368  91.17202 6.571246 0.9946391
## CC1          7456 8633.446  85.92656 6.771401 0.9951997
## SV1          5545 6751.597  90.04809 6.534320 0.9965679
## M31Fcsw      2555 3686.989 102.23463 3.827398 0.9275858
## M11Fcsw      2468 3761.882 116.51558 3.286831 0.9097244
## M31Plmr      3125 4320.168 106.09692 4.287632 0.9378896
## M11Plmr      4047 5088.833  88.06245 4.846838 0.9518325
## F21Plmr      2703 3823.854 100.22273 4.869254 0.9777085
## M31Tong      2901 4910.383 154.07868 2.670510 0.8624907
## M11Tong      2048 3749.797 152.70903 3.902651 0.9358453
## LMEpi24M     3477 5040.126 119.34349 3.092977 0.8022909
## SLEpi20M     3235 4663.840 115.76231 3.649837 0.9071917
## AQC1cm       6150 8211.414 132.45634 3.532024 0.7630573
## AQC4cm       6363 7830.001 101.26721 3.325816 0.7342254
## AQC7cm       6178 7800.500 112.64809 3.974387 0.8118201
## NP2          2481 3758.918 111.64152 4.217592 0.9529057
## NP3          3759 5258.402 118.47435 4.479315 0.9709596
## NP5          3303 5012.631 131.72665 4.533863 0.9740461
## TRRsed1      2912 4270.507 114.21665 6.135146 0.9922386
## TRRsed2      4611 6362.958 121.41534 4.832479 0.9628864
## TRRsed3      4382 6537.209 146.58032 5.446387 0.9803352
## TS28         2578 4145.831 137.30472 4.119753 0.9650524
## TS29         2521 3840.760 115.10451 3.445989 0.9178579
```

```
# results can be exported as .csv file for later use
# replace ~ with your file path
# write.csv(gp_alpha, "~/GP_AlphaDiv_all.csv")
```

Plot richness metrics by sample type

```
# plot untrimmed richness data using built in plotting function
phyloseq::plot_richness(ps_gp_bact, x="SampleType", measures=c("Observed", "Chao1", "Shannon", "Simpson
```



Faith's phylogenetic diversity (PD)

Calculate PD and SR

```
# using btools today, which is a wrapper for picante
# that directly calculates PD from a ps object
# otherwise, use picante::pd on ASV and TREE accessed from the ps object
pd <- btools::estimate_pd(ps_gp_bact)
pd
```

```
##                PD   SR
## CL3      241.3382 6745
## CC1      253.9796 7456
## SV1      201.3032 5545
## M31Fcsw  113.3431 2555
## M11Fcsw  115.5378 2468
## M31Plmr  132.3571 3125
## M11Plmr  155.3777 4047
## F21Plmr  119.5629 2703
## M31Tong  139.6491 2901
## M11Tong  109.1294 2048
## LMEpi24M 151.6039 3477
## SLEpi20M 143.5940 3235
## AQC1cm   238.6941 6150
## AQC4cm   242.6837 6363
## AQC7cm   234.9570 6178
## NP2      122.2333 2481
## NP3      161.2465 3759
## NP5      149.5964 3303
## TRRsed1  137.0418 2912
## TRRsed2  187.6274 4611
## TRRsed3  187.8474 4382
## TS28     119.8659 2578
## TS29     116.2812 2521
```

Plot PD and SR by sample type

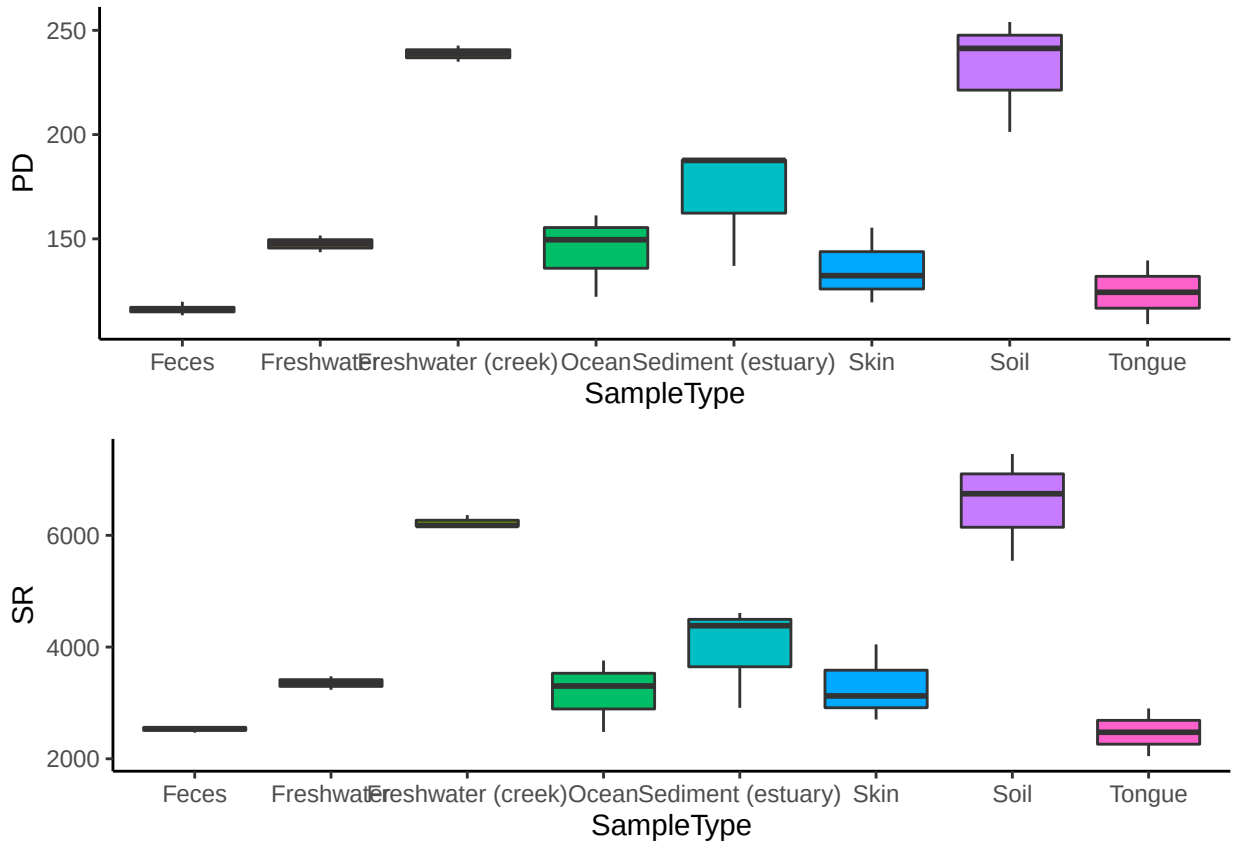
```
# access the sample data from the ps object
SAM <- phyloseq::sample_data(ps_gp_bact)

# merge pd into SAM
SAM2 <- merge(SAM, pd, by="row.names", all=TRUE)
colnames(SAM2)
```

```
##  [1] "Row.names"                "X.SampleID"
##  [3] "Primer"                   "Final_Barcode"
##  [5] "Barcode_truncated_plus_T" "Barcode_full_length"
##  [7] "SampleType"               "Description"
##  [9] "PD"                       "SR"
```

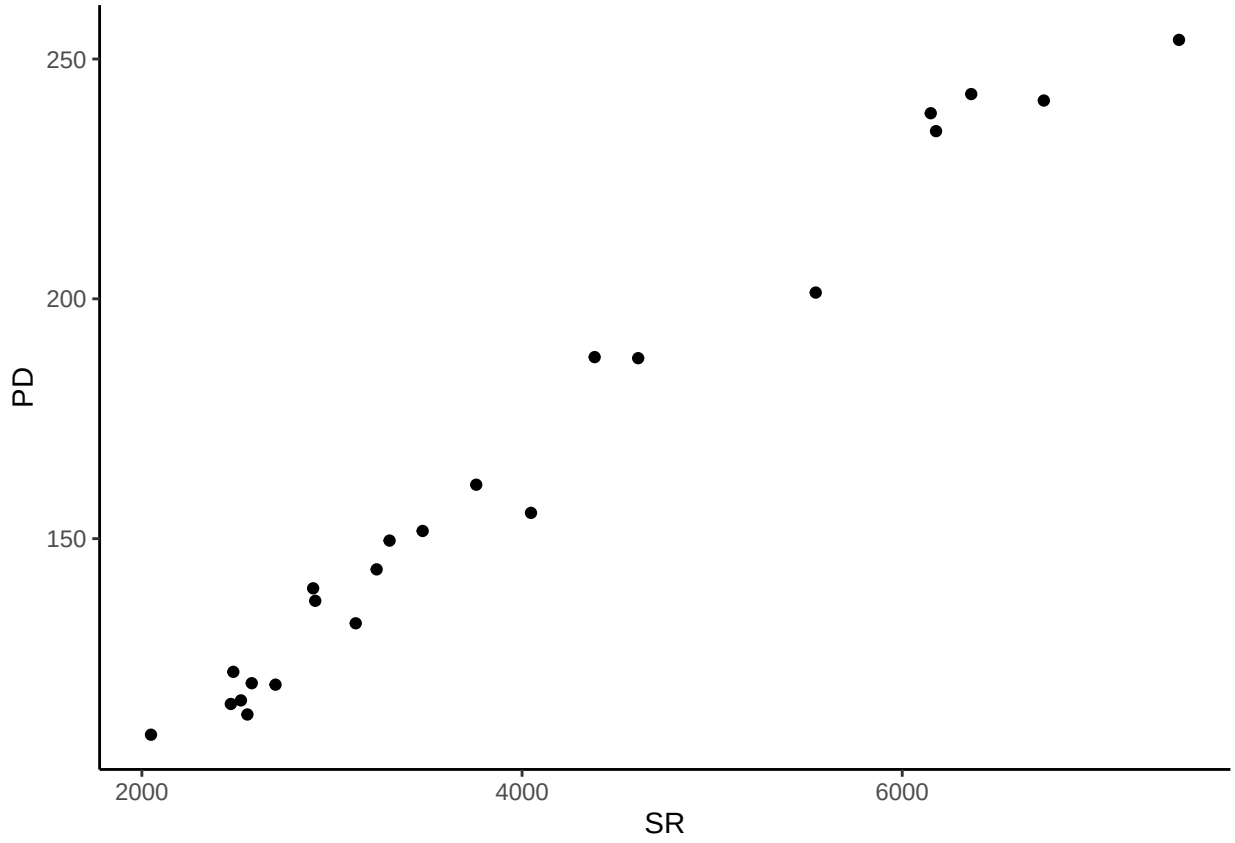
```
# examine PD by SampleType
p1 <- ggplot(data=SAM2, (aes(x=SampleType, y=PD, fill=SampleType))) +
  geom_boxplot() +
  theme_classic()
p2 <- ggplot(data=SAM2, (aes(x=SampleType, y=SR, fill=SampleType))) +
  geom_boxplot() +
  theme_classic()

ggpubr::ggarrange(p1, p2, nrow=2, legend = FALSE)
```



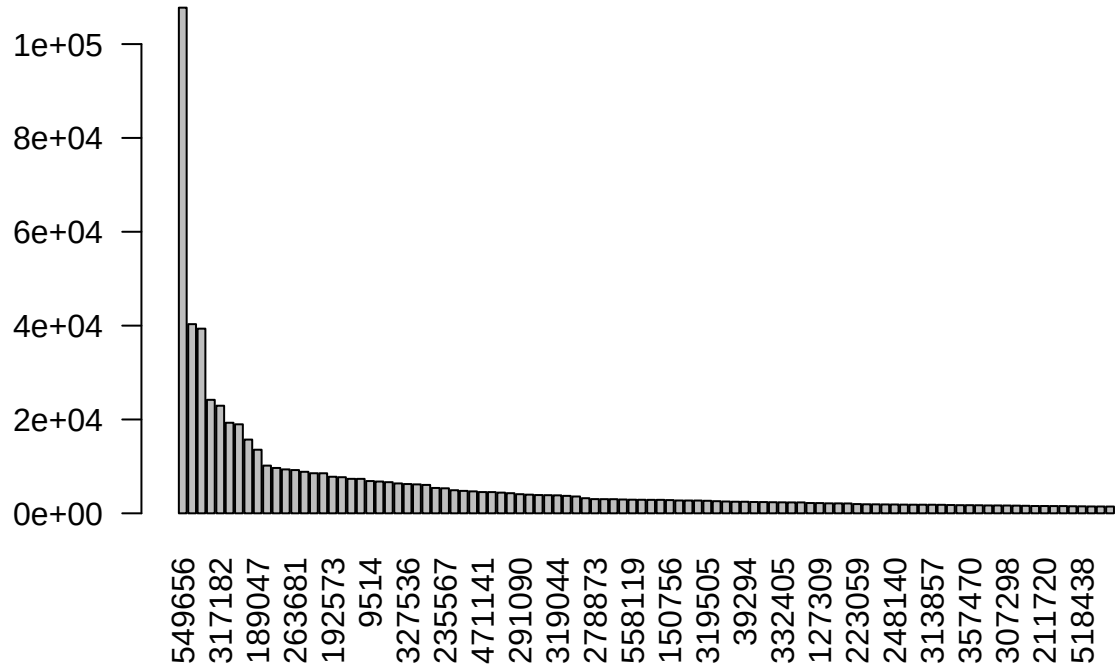
Compare PD and SR

```
# examine relationship between PD and SR
ggplot(data=SAM2, mapping=aes(x=SR, y=PD)) +
  geom_point() +
  theme_classic()
```



Rank-abundance curve

```
# limit rank-abundance curve to top 100
N <- 100
barplot(sort(taxa_sums(ps_gp_bact), TRUE) [1:N]/nsamples(ps_gp_bact), las=2)
```



Species accumulation and sampling curves

https://cran.r-project.org/web/packages/vegan/vegan.pdf

Export data from phyloseq to use in vegan

```
# export OTU table from ps object for use in vegan (replace with top100 if needed)
votu <- otu_table(ps_gp_bact)
votu <- t(votu) #transpose otutable so rows are samples
votu <- as.data.frame(votu) #convert to dataframe
# str(votu) #check that observations are numeric
anyNA(votu) #check for NA in data
```

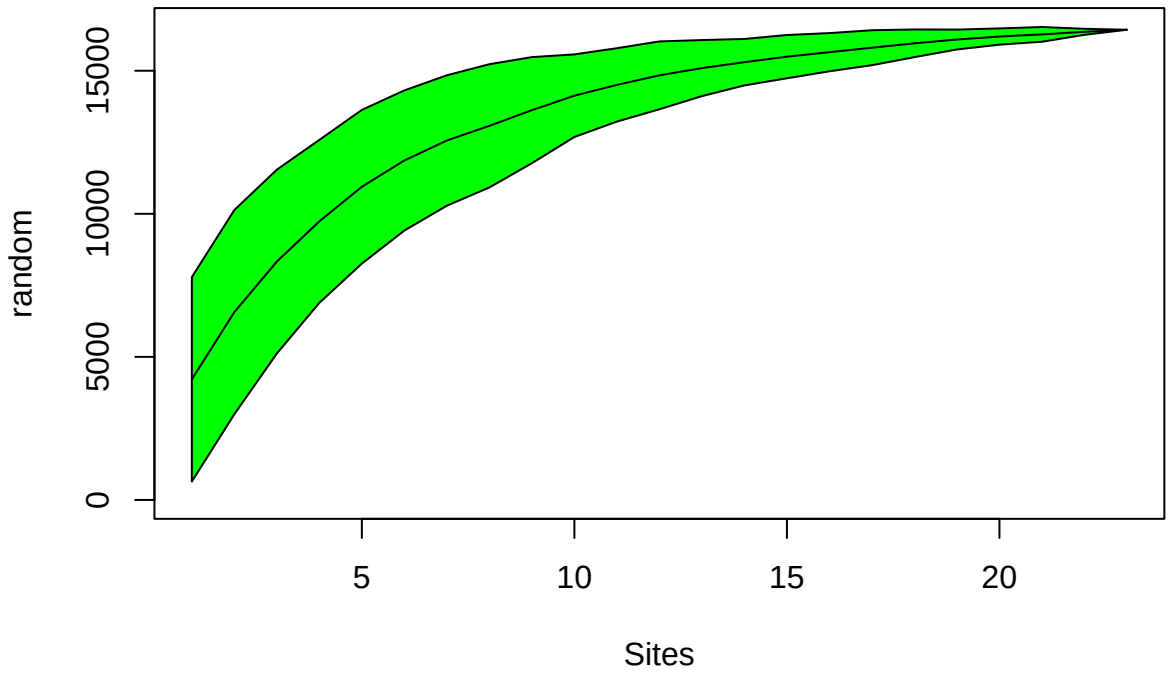
```
## [1] FALSE
```

```
#get sample data from ps object for vegan
vdata <- sample_data(ps_gp_bact)
str(vdata) #examine classifications
```

```
## 'data.frame':    23 obs. of  7 variables:
## Formal class 'sample_data' [package "phyloseq"] with 4 slots
##   ..@ .Data    :List of 7
##   .. ..$ : Factor w/ 23 levels "AQC1cm","AQC4cm",..: 5 4 18 11 8 12 9 6 13 10 ...
##   .. ..$ : Factor w/ 23 levels "ILBC_01","ILBC_02",..: 1 2 3 4 5 6 7 8 9 10 ...
##   .. ..$ : Factor w/ 23 levels "AACGCA","AACTCG",..: 1 2 3 4 5 6 7 8 9 10 ...
##   .. ..$ : Factor w/ 23 levels "AACTGT","ACAGTT",..: 21 11 2 18 8 4 15 6 17 10 ...
##   .. ..$ : Factor w/ 23 levels "AGCCGACTCTG",..: 9 5 17 20 7 8 15 18 23 19 ...
##   .. ..$ : Factor w/ 8 levels "Feces","Freshwater",..: 7 7 7 1 1 6 6 6 8 8 ...
##   .. ..$ : Factor w/ 22 levels "Allequash Creek, 0-1cm depth",..: 4 5 17 11 8 12 9 6 13 10 ...
##   ..@ names    : chr  "X.SampleID" "Primer" "Final_Barcode" "Barcode_truncated_plus_T" ...
##   ..@ row.names: chr  "CL3" "CC1" "SV1" "M31Fcsw" ...
##   ..@ .S3Class : chr "data.frame"
```

Make species accumulation curve (community level)

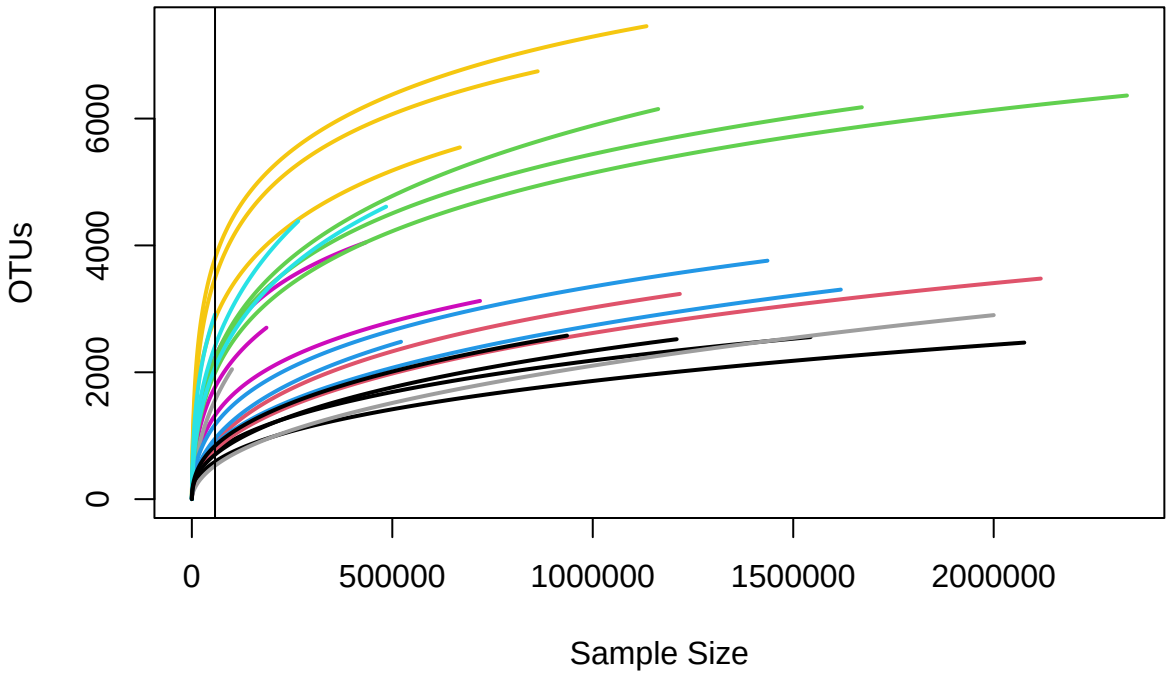
```
sac <- vegan::specaccum(votu, "random")
# classic method is random, but can also use "exact" (sample-based), "collector" (in order), etc.
plot(sac, ci.type="polygon", ci.col="green")
```



Make rarefaction curve (sample level)

```
# step = step size for sample sizes in rarefaction
# typical to start w/ step size of 100 but using 1000 here for speed
# if too slow on your computer, can switch to top100 taxa file or run at home

# run the next two code lines together
# abline adds vertical line at fewest number of sequences in any sample
vegan::rarecurve(votu, step=1000, col=vdata$SampleType, lwd=2, ylab="OTUs", label=F)
abline(v=(min(rowSums(votu))))
```



Coding Exercises

Please submit as a knitted html markdown to GitHub due on 2/16

1. Use phyloseq to zoom in on richness of specific phyla in the data:

- use phyloseq::get_taxa_unique to examine taxonomic Phyla in the ps_gp_bact data
  - https://rdrr.io/bioc/phyloseq/man/get_taxa_unique.html
  - how many phyla are present?
- use phyloseq::subset_taxa to select a single phylum (e.g., Actinobacteria)
  - https://rdrr.io/bioc/phyloseq/man/subset_taxa-methods.html
- use phyloseq::plot_bar to examine abundance of that phylum by sample type
  - https://rdrr.io/bioc/phyloseq/man/plot_bar.html
  - where are these most abundant?

2. Use phyloseq to examine genus-level richness:

- use the phyloseq::tax_glom function to agglomerate at the Genus level
  - https://rdrr.io/bioc/phyloseq/man/tax_glom.html
- rerun richness calcs and rank-abundance curves
- how have these changed and why?

3. Phyloseq acts as a wrapper for vegan for many of its community metrics Use vegan::diversity to calculate untrimmed data Shannon's H, Simpson's D:

- https://www.rdocumentation.org/packages/vegan/versions/2.4-2/topics/diversity
- check that calculations are the same
- plot diversity as histograms using the hist function

4. Use vegan::radfit to determine the best model fit for rank-abundance curves (lowest AIC value) and plot

- https://www.rdocumentation.org/packages/vegan/versions/2.4-2/topics/radfit
- which model is the best fit?

Session Info

```
sessionInfo()
```

```
## R version 4.1.2 (2021-11-01)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19042)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] ggpubr_0.4.0    btools_0.0.1    vegan_2.5-7     lattice_0.20-45
##  [5] permute_0.9-7   phyloseq_1.38.0 forcats_0.5.1   stringr_1.4.0
##  [9] dplyr_1.0.7     purrr_0.3.4     readr_2.1.1     tidyr_1.1.4
## [13] tibble_3.1.6    ggplot2_3.3.5   tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-155           bitops_1.0-7           fs_1.5.2
##  [4] lubridate_1.8.0        httr_1.4.2             GenomeInfoDb_1.30.0
##  [7] tools_4.1.2            backports_1.4.1        utf8_1.2.2
## [10] R6_2.5.1               mgcv_1.8-38            DBI_1.1.2
## [13] BiocGenerics_0.40.0    colorspace_2.0-2       rhdf5filters_1.6.0
## [16] ade4_1.7-18            withr_2.4.3            tidyselect_1.1.1
## [19] compiler_4.1.2         cli_3.1.1              rvest_1.0.2
## [22] Biobase_2.54.0         xml2_1.3.3             labeling_0.4.2
## [25] scales_1.1.1           digest_0.6.29          rmarkdown_2.11
## [28] XVector_0.34.0         pkgconfig_2.0.3        htmltools_0.5.2
## [31] highr_0.9              dbplyr_2.1.1           fastmap_1.1.0
## [34] rlang_0.4.12           readxl_1.3.1           rstudioapi_0.13
## [37] farver_2.1.0           generics_0.1.2         jsonlite_1.7.3
## [40] car_3.0-12             RCurl_1.98-1.5         magrittr_2.0.1
## [43] GenomeInfoDbData_1.2.7 biomformat_1.22.0      Matrix_1.4-0
## [46] Rcpp_1.0.8             munsell_0.5.0          S4Vectors_0.32.3
## [49] Rhdf5lib_1.16.0        fansi_0.5.0            abind_1.4-5
## [52] ape_5.6-1              lifecycle_1.0.1        stringi_1.7.6
## [55] yaml_2.2.1             carData_3.0-5          MASS_7.3-54
## [58] zlibbioc_1.40.0        rhdf5_2.38.0           plyr_1.8.6
## [61] grid_4.1.2             parallel_4.1.2         crayon_1.4.2
## [64] cowplot_1.1.1          splines_4.1.2          Biostrings_2.62.0
## [67] haven_2.4.3            multtest_2.50.0        hms_1.1.1
## [70] knitr_1.37             pillar_1.7.0           igraph_1.2.11
## [73] ggsignif_0.6.3         reshape2_1.4.4         codetools_0.2-18
```

```
## [76] stats4_4.1.2          picante_1.8.2         reprex_2.0.1
## [79] glue_1.6.0            evaluate_0.14         data.table_1.14.2
## [82] modelr_0.1.8          vctrs_0.3.8           tzdb_0.2.0
## [85] foreach_1.5.1         cellranger_1.1.0      gtable_0.3.0
## [88] assertthat_0.2.1      xfun_0.29             broom_0.7.11
## [91] rstatix_0.7.0         survival_3.2-13       iterators_1.0.13
## [94] IRanges_2.28.0        cluster_2.1.2         ellipsis_0.3.2
```