5. Worksheet: Alpha Diversity

Jonathan Enriquez Madrid; Z620: Quantitative Biodiversity, Indiana University

24 January, 2023

OVERVIEW

In this exercise, we will explore aspects of local or site-specific diversity, also known as alpha (α) diversity. First we will quantify two of the fundamental components of (α) diversity: **richness** and **evenness**. From there, we will then discuss ways to integrate richness and evenness, which will include univariate metrics of diversity along with an investigation of the **species abundance distribution (SAD)**.

Directions:

1. In the Markdown version of this document in your cloned repo, change "Student Name" on line 3 (above) to your name.
2. Complete as much of the worksheet as possible during class.
3. Use the handout as a guide; it contains a more complete description of data sets along with the proper scripting needed to carry out the exercise.
4. Answer questions in the worksheet. Space for your answer is provided in this document and indicated by the ">" character. If you need a second paragraph be sure to start the first line with ">". You should notice that the answer is highlighted in green by RStudio (color may vary if you changed the editor theme).
5. Before you leave the classroom, **push** this file to your GitHub repo.
6. For the assignment portion of the worksheet, follow the directions at the bottom of this file.
7. When you are done, **Knit** the text and code into a PDF file.
8. After Knitting, submit the completed exercise by creating a **pull request** via GitHub. Your pull request should include this file `AlphaDiversity_Worskheet.Rmd` and the PDF output of `Knitr` (`AlphaDiversity_Worskheet.pdf`).

1) R SETUP

In the R code chunk below, please provide the code to: 1) Clear your R environment, 2) Print your current working directory, 3) Set your working directory to your `5.AlphaDiversity` folder, and 4) Load the `vegan` R package (be sure to install first if you haven't already).

```r
rm(list = ls())
getwd()
```

```
## [1] "C:/Users/jonat/GitHub/QB2023_Enriquez_Madrid/2.Worksheets/5.AlphaDiversity"
```

```r
setwd("C:/Users/jonat/GitHub/QB2023_Enriquez_Madrid/2.Worksheets/5.AlphaDiversity")
#install.packages("vegan")
require("vegan")
```

```
## Loading required package: vegan
```

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

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

```
## This is vegan 2.6-4
```

2) LOADING DATA

In the R code chunk below, do the following: 1) Load the BCI dataset, and 2) Display the structure of the dataset (if the structure is long, use the `max.level = 0` argument to show the basic information).

```
data(BCI)
```

3) SPECIES RICHNESS

Species richness (S) refers to the number of species in a system or the number of species observed in a sample.

Observed richness

In the R code chunk below, do the following:

1. Write a function called `S.obs` to calculate observed richness
2. Use your function to determine the number of species in `site1` of the BCI data set, and
3. Compare the output of your function to the output of the `specnumber()` function in `vegan`.

```
S.obs <- function(x = ""){
  rowSums(x > 0) * 1
}#function for finding richness at sites. The number of diff. species at a sight.
#S.obs(BCI, m[1,]), wrong way of doing it.
S.obs(BCI[1,]) #93 species
```

```
##  1
## 93
```

```
specnumber(BCI[1,])#93 species
```

```
##  1
## 93
```

```
S.obs(BCI[1:4,])#93,84,90,94
```

```
##  1  2  3  4
## 93 84 90 94
```

Question 1: Does `specnumber()` from `vegan` return the same value for observed richness in `site1` as our function `S.obs`? What is the species richness of the first four sites (i.e., rows) of the BCI matrix?

> ***Answer 1***: The 'specnumber' from 'vegan' does return the same value for observed richness in site1 as our 'S.obs' function. Both give the value of 93. In addition, the species richness for row 2 is 84, row 3 is 90, and row 4 is 94.

Coverage: How well did you sample your site?

In the R code chunk below, do the following:

1. Write a function to calculate Good's Coverage, and
2. Use that function to calculate coverage for all sites in the BCI matrix.

```r
C <- function(x = ""){1 - (rowSums(x == 1)/rowSums(x))}#Function for Good's Coverage
C(BCI)#measures the percent of coverage.
```

```
##         1         2         3         4         5         6         7         8
## 0.9308036 0.9287356 0.9200864 0.9468504 0.9287129 0.9174757 0.9326923 0.9443155
##         9        10        11        12        13        14        15        16
## 0.9095355 0.9275362 0.9152120 0.9071038 0.9242054 0.9132420 0.9350649 0.9267735
##        17        18        19        20        21        22        23        24
## 0.8950131 0.9193084 0.8891455 0.9114219 0.8946078 0.9066986 0.8705882 0.9030612
##        25        26        27        28        29        30        31        32
## 0.9095023 0.9115479 0.9088729 0.9198966 0.8983516 0.9221053 0.9382423 0.9411765
##        33        34        35        36        37        38        39        40
## 0.9220183 0.9239374 0.9267887 0.9186047 0.9379310 0.9306488 0.9268868 0.9386503
##        41        42        43        44        45        46        47        48
## 0.8880597 0.9299517 0.9140049 0.9168704 0.9234234 0.9348837 0.8847059 0.9228916
##        49        50
## 0.9086651 0.9143519
```

Question 2: Answer the following questions about coverage:

a. What is the range of values that can be generated by Good's Coverage?
b. What would we conclude from Good's Coverage if n_i equaled N?
c. What portion of taxa in `site1` was represented by singletons?
d. Make some observations about coverage at the BCI plots.

> ***Answer 2a***: The range of values that can be generated by Good's Coverage is 0 to 1, where 0 is no coverage and 1 is total coverage, and numbers between 0 and 1 show the proportion of coverage between 0 and 1. A large value of Good's Coverage means you have a small probability of finding new species in your sample.

> ***Answer 2b***: If the number of singletons is equal to the number of total individuals in the sample, this means there is low coverage and the probability of finding a new species in the sample is high. Say the number of singletons is 10 and the number of total individuals in the sample is 10. 10 divided by 10 is 1, and if we subtract 1 by 1 we get a Good's Coverage value of 0, indicading no covarage.

Answer 2c:The proportion of taxa in site 1 that is represented by singletons is 0.0691964. The value Good's Coverage spits out shows the proportion of taxa that is sampled more than once, by subtracting this value by 1 we get the proportion of individuals sampled only once.

Answer 2d: Coverage at the BCI plots seems to be very good, with the smallest value of coverage being 0.8705882. This is close to a value of 1 which indicates complete coverage.

Estimated richness

In the R code chunk below, do the following:

1. Load the microbial dataset (located in the `5.AlphaDiversity/data` folder),
2. Transform and transpose the data as needed (see handout),
3. Create a new vector (`soilbac1`) by indexing the bacterial OTU abundances of any site in the dataset,
4. Calculate the observed richness at that particular site, and
5. Calculate coverage of that site

```
soilbac <- read.table("data/soilbac.txt", sep = "\t", header = TRUE, row.names = 1)
soilbac.t <- as.data.frame(t(soilbac))
soilbac1 <- soilbac.t[1,] #New vector created for T1_1 site
S.obs(soilbac1) #Observed richness at T1_1 is 1074.
```

```
## T1_1
## 1074
```

```
C(soilbac1) #Good's Coverage value of 0.647941 (not great coverage)
```

```
##      T1_1
## 0.6479471
```

Question 3: Answer the following questions about the soil bacterial dataset.

a. How many sequences did we recover from the sample `soilbac1`, i.e. N?
b. What is the observed richness of `soilbac1`?
c. How does coverage compare between the BCI sample (`site1`) and the KBS sample (`soilbac1`)?

Answer 3a: I do not know how to add the number of sequences from each column together, which is how we get the number of the sequences we recovered from sample soilbac1.

Answer 3b: The observed richness of soilbac1 is 1074 different OTUs.

Answer 3c: Coverage for the KBS sample (0.647941) is lower than coverage for site 1 in the BCI sample (0.9308036).

Richness estimators

In the R code chunk below, do the following:

1. Write a function to calculate **Chao1**,
2. Write a function to calculate **Chao2**,
3. Write a function to calculate **ACE**, and
4. Use these functions to estimate richness at `site1` and `soilbac1`.

```
S.chao1 <- function(x = ""){S.obs(x) + (sum(x == 1)^2) / (2 * sum(x == 2))}

S.chao2 <- function(site = "", SbyS = ""){
  SbyS = as.data.frame(SbyS)
  x = SbyS[site,]
  SbyS.pa <- (SbyS > 0)* 1
  Q1 = sum(colSums(SbyS.pa) ==1)
  Q2 = sum(colSums(SbyS.pa) ==2)
  S.chao2 = S.obs(x) + (Q1^2)/(2 * Q2)
  return(S.chao2)
}

S.ace <- function(x = "", thresh = 10){
  x <- x[x>0]
  S.abund <- length(which(x > thresh))
  S.rare <- length(which(x <= thresh))
  singlt <- length(which(x == 1))
  N.rare <- sum(x[which(x <= thresh)])
  C.ace <- 1 - (singlt / N.rare)
  i <- c(1:thresh)
  count <- function(i, y){
    length(y[y == i])
  }
  a.1 <- sapply(i, count, x)
  f.1 <- (i * (i - 1)) * a.1
  G.ace <- (S.rare/C.ace)*(sum(f.1)/(N.rare*(N.rare-1)))
  S.ace <- S.abund + (S.rare/C.ace) + (singlt/C.ace) * max(G.ace, 0)
  return(S.ace)
}

S.chao1(BCI) #at site 1, value of 1855.365.
```

```
##        1        2        3        4        5        6        7        8
## 1855.365 1846.365 1852.365 1856.365 1863.365 1847.365 1844.365 1850.365
##        9       10       11       12       13       14       15       16
## 1852.365 1856.365 1849.365 1846.365 1855.365 1860.365 1855.365 1855.365
##       17       18       19       20       21       22       23       24
## 1855.365 1851.365 1871.365 1862.365 1861.365 1853.365 1861.365 1857.365
##       25       26       27       28       29       30       31       32
## 1867.365 1853.365 1861.365 1847.365 1848.365 1859.365 1839.365 1850.365
##       33       34       35       36       37       38       39       40
## 1848.365 1854.365 1845.365 1854.365 1850.365 1844.365 1846.365 1842.365
##       41       42       43       44       45       46       47       48
## 1864.365 1849.365 1848.365 1843.365 1843.365 1848.365 1864.365 1853.365
##       49       50
## 1853.365 1855.365
```

```
S.chao2("1",BCI) #at site 1 value of 104.6053.
```

```
##        1
## 104.6053
```

```
S.ace(BCI[1,])#159.3404
```

```
## [1] 159.3404
```

```
S.chao1(soilbac1) #at T1_1, value of 2628.514.
```

```
##     T1_1
## 2628.514
```

```
S.chao2("T1_1",soilbac.t) #at T1_1, value of 21055.39.
```

```
##     T1_1
## 21055.39
```

```
S.ace(soilbac1)#4465.883
```

```
## [1] 4465.983
```

Question 4: What is the difference between ACE and the Chao estimators? Do the estimators give consistent results? Which one would you choose to use and why?

> ***Answer 4***:The difference between ACE and Chao estimators is ACE is used to find the abundance of rare species. ACE uses a threshold of 10 individuals to classify a taxa as rare. Chao1 is used to check for richness at a single site, while Chao2 is used to examine richness across multiple sites. The estimators do not give consistent results. Depending on whether one uses chao1 or choa2, one will get different values. As for ACE, the code did not read the data, but I would assume that the values would not be consistent with that of chao1 and choa2 as ACE is used to find the abundance of rare species. The estimator one chooses is dependent on the question one is asking. If one is looking for the richness at a single site, one uses chao1. If one is looking for richness across sites one uses chao2. If one is looking for abundance of rare taxa one uses ACE.

Rarefaction

In the R code chunk below, please do the following:

1. Calculate observed richness for all samples in `soilbac`,
2. Determine the size of the smallest sample,

3. Use the `rarefy()` function to rarefy each sample to this level,
4. Plot the rarefaction results, and
5. Add the 1:1 line and label.

```
soilbac.S <- S.obs(soilbac.t)
soilbac.S #observed richness for all samples in soilbac.t
```

```
## T1_1 T1_2 T1_3 T7_1 T7_2 T7_3 DF_1 DF_2 CF_1 CF_2 CF_3
## 1074 1302 1174 1416 1406 1143 1806 1151  924 1122  851
```

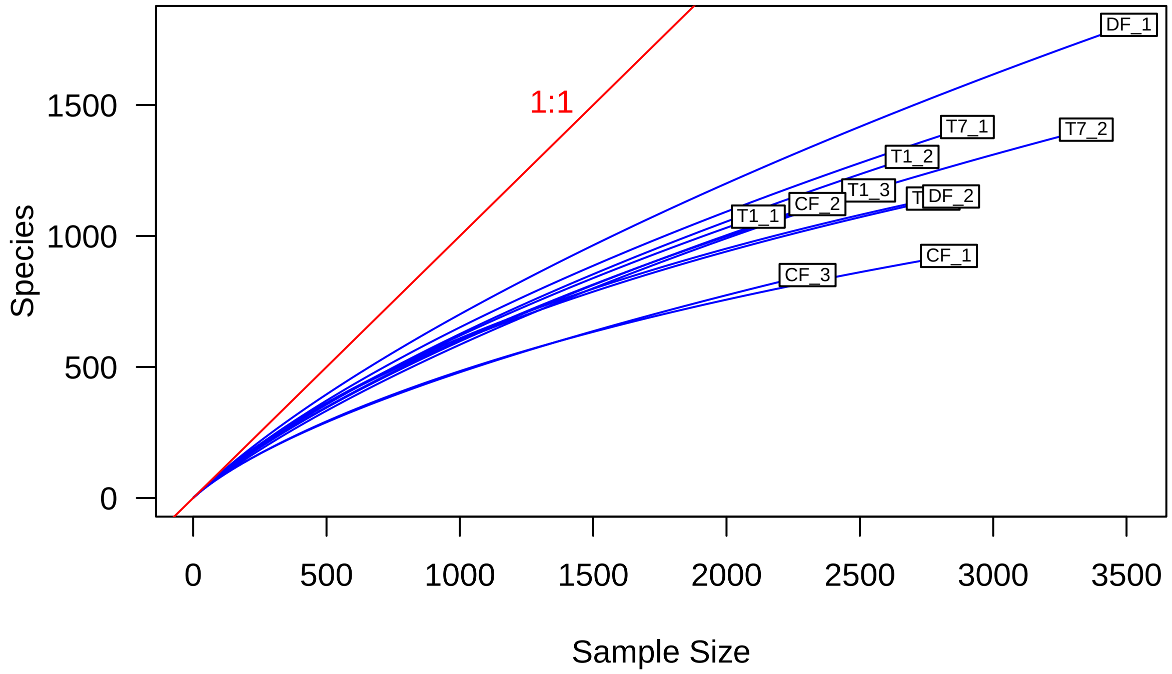
```
min.N <- min(rowSums(soilbac.t))
min.N #Size of smallest sample is 2119.
```

```
## [1] 2119
```

```
S.rarefy <- rarefy(x = soilbac.t, sample = min.N, se = TRUE)
S.rarefy#Rarefy so that all samples have 2119 sample size.
```

```
##    T1_1       T1_2        T1_3       T7_1       T7_2       T7_3       DF_1
## S  1074 1099.69226 1033.618984 1138.85781 1039.66098 984.552923 1254.09586
## se    0    9.92876    8.668344   11.10399   12.38929   9.376365   13.53094
##          DF_2       CF_1        CF_2       CF_3
## S  973.839396 783.458905 1045.431707 804.746297
## se   9.782744   9.075373    6.673692   5.623012
## attr(,"Subsample")
## [1] 2119
```

```
rarecurve(x = soilbac.t, step = 20, col = "blue", cex = 0.6, las = 1)
abline(0, 1, col = 'red')
text(1500, 1500, "1:1", pos = 2, col = 'red')
```



4) SPECIES EVNENNESS

Here, we consider how abundance varies among species, that is, **species evenness**.

Visualizing evenness: the rank abundance curve (RAC)

One of the most common ways to visualize evenness is in a **rank-abundance curve** (sometime referred to as a rank-abundance distribution or Whittaker plot). An RAC can be constructed by ranking species from the most abundant to the least abundant without respect to species labels (and hence no worries about 'ties' in abundance).

In the R code chunk below, do the following:

1. Write a function to construct a RAC,
2. Be sure your function removes species that have zero abundances,
3. Order the vector (RAC) from greatest (most abundant) to least (least abundant), and
4. Return the ranked vector

```r
RAC <- function(x = ""){
  x.ab = x[x > 0]
  x.ab.ranked = x.ab[order(x.ab, decreasing = TRUE)]
  as.data.frame(lapply(x.ab.ranked, unlist))
  return(x.ab.ranked)
}
```

Now, let us examine the RAC for `site1` of the BCI data set.

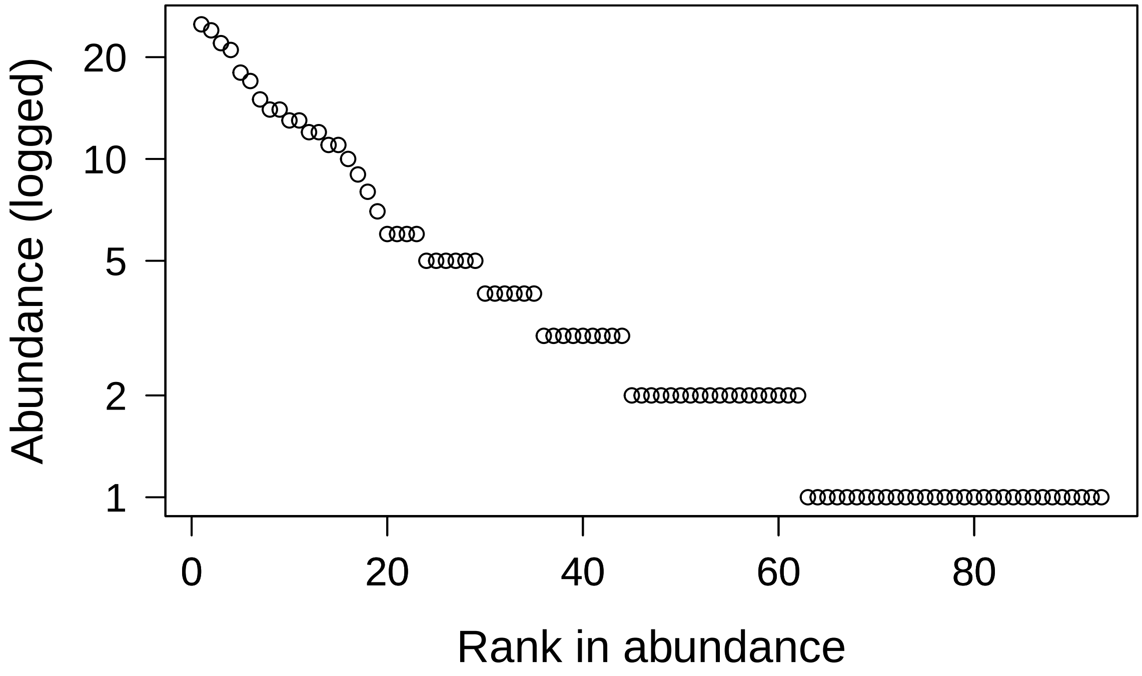
In the R code chunk below, do the following:

1. Create a sequence of ranks and plot the RAC with natural-log-transformed abundances,
2. Label the x-axis "Rank in abundance" and the y-axis "log(abundance)"

```
plot.new()

site1 <- BCI[1,]

rac <- RAC(x = site1)
ranks <- as.vector(seq(1, length(rac)))
opar <- par(no.readonly = TRUE)
par(mar = c(5.1, 5.1, 4.1, 2.1))
plot(ranks, log(rac), type = 'p', axis = F,
     xlab = "Rank in abundance", ylab = "Abundance (logged)",
     las = 1, cex.lab = 1.4, cex.axis = 1.25, yaxt = "n")
box()
axis(1, labels = T, cex.axis = 1.25)
axis(2, las = 1, cex.axis = 1.25,
     labels = c(1, 2, 5, 10, 20), at = log(c(1, 2, 5, 10, 20)))
```



Question 5: What effect does visualizing species abundance data on a log-scaled axis have on how we interpret evenness in the RAC?

> ***Answer 5***:Being able to visualize species abundance data allows us to see how some species are more present than others. This shows us that species evenness is low among these species.

Now that we have visualized unevennes, it is time to quantify it using Simpson's evenness ($E_{1/D}$) and Smith and Wilson's evenness index (E_{var}).

Simpson's evenness ($E_{1/D}$)

In the R code chunk below, do the following:

1. Write the function to calculate $E_{1/D}$, and
2. Calculate $E_{1/D}$ for `site1`.

```
SimpE <- function(x = ""){
  S <- S.obs(x)
  x = as.data.frame(x)
  D <- diversity(x, "inv")
  E <- (D)/S
  return(E)
}
site1 <- BCI[1,]
SimpE(site1)#Moderately even value of 0.4238232 (max evenness is a value of 1)
```

```
##         1
## 0.4238232
```

Smith and Wilson's evenness index (E_{var})

In the R code chunk below, please do the following:

1. Write the function to calculate E_{var},
2. Calculate E_{var} for `site1`, and
3. Compare $E_{1/D}$ and E_{var}.

```
Evar <- function(x){
  x <- as.vector(x[x > 0])
  1 - (2/pi) * atan(var(log(x)))
}
Evar(site1) #evennes value of 0.5067211
```

```
## [1] 0.5067211
```

Question 6: Compare estimates of evenness for `site1` of BCI using $E_{1/D}$ and E_{var}. Do they agree? If so, why? If not, why? What can you infer from the results.

> ***Answer 6***:The estimates of evenness for site1 of the BCI data from Simpson's Evenness and Smith & Wilson's Evenness Index do not agree. They are somewhat similar, with Simpson's Evenness value being 0.4238232, and Smith & Wilson's Evenness Index being 0.5067211. The reason for this difference in values being that Smith & Wilson's Evenness Index takes into account the bias towards the most abundant species by using natural log. Simpson's Evenness falls to the bias of the most abundant species. From these estimates of evenness we can infer that there is moderate evenness at site1 of the BCI data.

5) INTEGRATING RICHNESS AND EVENNESS: DIVERSITY METRICS

So far, we have introduced two primary aspects of diversity, i.e., richness and evenness. Here, we will use popular indices to estimate diversity, which explicitly incorporate richness and evenness We will write our own diversity functions and compare them against the functions in `vegan`.

Shannon's diversity (a.k.a., Shannon's entropy)

In the R code chunk below, please do the following:

1. Provide the code for calculating H' (Shannon's diversity),
2. Compare this estimate with the output of `vegan`'s diversity function using method = "shannon".

```
ShanH <- function(x = ""){
  H = 0
  for (n_i in x){
    if (n_i > 0){
      p = n_i / sum(x)
      H = H - p*log(p)
    }
  }
  return(H)
}

ShanH(site1) #value of 4.018412.
```

```
## [1] 4.018412
```

```
diversity(site1, index = "shannon")#value of 4.018412. This function is from the vegan package.
```

```
## [1] 4.018412
```

Simpson's diversity (or dominance)

In the R code chunk below, please do the following:

1. Provide the code for calculating D (Simpson's diversity),
2. Calculate both the inverse (1/D) and 1 - D,
3. Compare this estimate with the output of `vegan's` diversity function using method = "simp".

```r
SimpD <- function(x = ""){
  D = 0
  N = sum(x)
  for (n_i in x){
    D = D + (n_i^2)/(N^2)
    }
  return(D)}
SimpD(site1) #0.0253707
```

```
## [1] 0.0253707
```

```r
D.inv <- 1/SimpD(site1)
D.inv #value of 39.41555
```

```
## [1] 39.41555
```

```r
D.sub <- 1-SimpD(site1)
D.sub #value of 0.9746293
```

```
## [1] 0.9746293
```

```r
diversity(site1, "inv") #value of 39.41555. From vegan package.
```

```
## [1] 39.41555
```

```r
diversity(site1, "simp") #value of 0.9746293. From vegan package.
```

```
## [1] 0.9746293
```

Fisher's α

In the R code chunk below, please do the following:

1. Provide the code for calculating Fisher's α,
2. Calculate Fisher's α for `site1` of BCI.

```r
rac <- as.vector(site1[site1 > 0])
invd <- diversity(rac, "inv")
invd #value of 39.41555 (same as D.inv value)
```

```
## [1] 39.41555
```

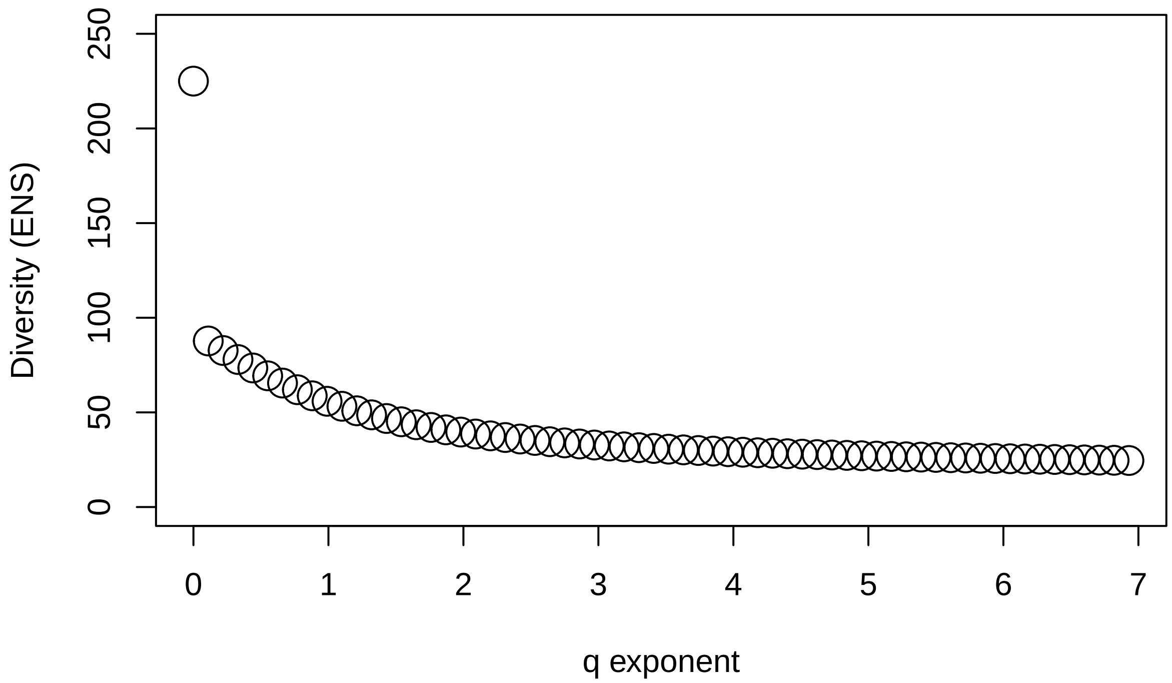
```r
Fisher <- fisher.alpha(rac)
Fisher #Value of 35.67297.
```

```
## [1] 35.67297
```

```
#Hill Numbers

profile <- function(C) {
  cbind(seq(0, 7, by = 0.11),
        unlist(lapply(seq(0, 7, by = 0.11),function(q) sum(apply(C, 1, function(x)
          (x/sum(x))^q))^(1/(1-q)))))
}

S1_profile <- profile(site1)
set.seed(42)
plot(S1_profile[,1], S1_profile[,2], ylim=c(0,250), cex = 2,
     xlab = "q exponent", ylab = "Diversity (ENS)")
```



Question 7: How is Fisher's $\boldsymbol{\alpha}$ different from $E_{H'}$ and E_{var}? What does Fisher's $\boldsymbol{\alpha}$ take into account that $E_{H'}$ and E_{var} do not?

> ***Answer 7***: Fisher's alpha is different from Shannon's diversity in that Fisher's alpha is an estimate of diversity rather than a calculated metric of diversity like Shannon's Diversity. In addition, Fisher's alpha is different from Smith & Wilson's evenness index in that Fisher's alpha is an estimate of diversity which includes both richness and evenness. Smith & Wilsnon's evenness index only measures evenness. Fisher's alpha also takes into account sampling error, something neither Shannon's diversity or Smith & Wilsnon's evenness index take into account.
> ## 6) HILL NUMBERS

Remember that we have learned about the advantages of Hill Numbers to measure and compare diversity among samples. We also learned to explore the effects of rare species in a community by examining diversity for a series of exponents q.

Question 8: Using `site1` of BCI and `vegan` package, a) calculate Hill numbers for q exponent 0, 1 and 2 (richness, exponential Shannon's entropy, and inverse Simpson's diversity). b) Interpret the effect of rare species in your community based on the response of diversity to increasing exponent q.

> ***Answer 8a***:I created a graph that measured diversity in response to q exponent. Hill number for q exponent 0 seems to be around 225, Hill number for q exponent 1 seems to be around 58, and Hill number for q exponent 2 seems to be around 46.
> ***Answer 8b***: Diversity decreases as the the q exponent increases.

##7) MOVING BEYOND UNIVARIATE METRICS OF α DIVERSITY

The diversity metrics that we just learned about attempt to integrate richness and evenness into a single, univariate metric. Although useful, information is invariably lost in this process. If we go back to the rank-abundance curve, we can retrieve additional information – and in some cases – make inferences about the processes influencing the structure of an ecological system.

Species abundance models

The RAC is a simple data structure that is both a vector of abundances. It is also a row in the site-by-species matrix (minus the zeros, i.e., absences).

Predicting the form of the RAC is the first test that any biodiversity theory must pass and there are no less than 20 models that have attempted to explain the uneven form of the RAC across ecological systems.

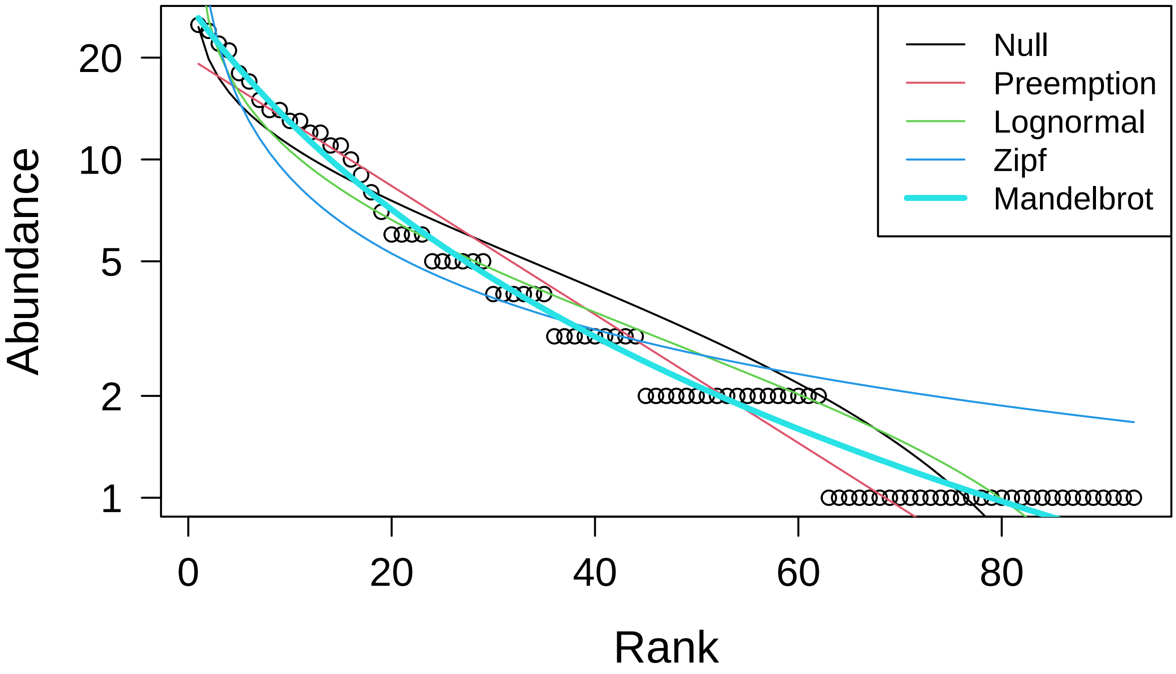
In the R code chunk below, please do the following:

1. Use the `radfit()` function in the `vegan` package to fit the predictions of various species abundance models to the RAC of `site1` in BCI,
2. Display the results of the `radfit()` function, and
3. Plot the results of the `radfit()` function using the code provided in the handout.

```
RACresults <- radfit(site1)
RACresults #Mandelbrot model has the lowest AIC and BIC values (low AIC and BIC values correspond to a
```

```
##
## RAD models, family poisson
## No. of species 93, total abundance 448
##
##            par1      par2     par3    Deviance AIC      BIC
## Null                                   39.5261 315.4362 315.4362
## Preemption  0.042797                   21.8939 299.8041 302.3367
## Lognormal   1.0687    1.0186           25.1528 305.0629 310.1281
## Zipf        0.11033  -0.74705          61.0465 340.9567 346.0219
## Mandelbrot  100.52   -2.312    24.084   4.2271 286.1372 293.7350
```

```
plot.new()
plot(RACresults, las = 1, cex.lab = 1.4, cex.axis = 1.25)#Namdelbrot model fits data the best.
```



Question 9: Answer the following questions about the rank abundance curves: a) Based on the output of `radfit()` and plotting above, discuss which model best fits our rank-abundance curve for `site1`? b) Can we make any inferences about the forces, processes, and/or mechanisms influencing the structure of our system, e.g., an ecological community?

> ***Answer 9a***: The model that best fits our rank-abundance curve for site1 is the Mandelbrot model. This model has the lowest AIC and BIC values.
> ***Answer 9b***: The model that has the most parameters seems to fit the data the best in this instance. This tells us that, although the parameters are not described, they do influence how the model fits the data, showing us that forces, processes, and/or mechanisms do influence the structure of our system.

Question 10: Answer the following questions about the preemption model: a. What does the preemption model assume about the relationship between total abundance (N) and total resources that can be preempted? b. Why does the niche preemption model look like a straight line in the RAD plot?

> ***Answer 10a***: The preemption model assumes that relationship between total abundance and total resources is 1 to 1. The preemption model is a straight line, decreasing in abundance as rank increases.
> ***Answer 10b***: I assume that the niche preemption model looks like a straight line because the ratio of abundance and rank is 1 to 1.

Question 10: Why is it important to account for the number of parameters a model uses when judging how well it explains a given set of data?

Answer 11: It is important to account for the number of parameters in a model as one does not want a over-parameterized model, as it might be difficult to discern which parameter is most influential in the data. AIC and BIC assign penalties to parameters, with more parameters leading to more penalties.

SYNTHESIS

1. As stated by Magurran (2004) the $D = \sum p_i^2$ derivation of Simpson's Diversity only applies to communities of infinite size. For anything but an infinitely large community, Simpson's Diversity index is calculated as $D = \sum \frac{n_i(n_i-1)}{N(N-1)}$. Assuming a finite community, calculate Simpson's D, 1 - D, and Simpson's inverse (i.e. 1/D) for `site 1` of the BCI site-by-species matrix.
2. Along with the rank-abundance curve (RAC), another way to visualize the distribution of abundance among species is with a histogram (a.k.a., frequency distribution) that shows the frequency of different abundance classes. For example, in a given sample, there may be 10 species represented by a single individual, 8 species with two individuals, 4 species with three individuals, and so on. In fact, the rank-abundance curve and the frequency distribution are the two most common ways to visualize the species-abundance distribution (SAD) and to test species abundance models and biodiversity theories. To address this homework question, use the R function **hist()** to plot the frequency distribution for `site 1` of the BCI site-by-species matrix, and describe the general pattern you see.
3. We asked you to find a biodiversity dataset with your partner. This data could be one of your own or it could be something that you obtained from the literature. Load that dataset. How many sites are there? How many species are there in the entire site-by-species matrix? Any other interesting observations based on what you learned this week?

SUBMITTING YOUR ASSIGNMENT

Use Knitr to create a PDF of your completed 5.AlphaDiversity_Worksheet.Rmd document, push it to GitHub, and create a pull request. Please make sure your updated repo include both the pdf and RMarkdown files.

Unless otherwise noted, this assignment is due on **Wednesday, January 25th, 2023 at 12:00 PM (noon)**.