5. Worksheet: Alpha Diversity

Jaeyoung Yoo; Z620: Quantitative Biodiversity, Indiana University

28 January, 2025

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 `Week-2/` folder folder, and 4) Load the `vegan` R package (be sure to install first if you have not already).

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

```
## [1] "/cloud/project/QB2025_Yoo/Week2-Alpha"
```

```
setwd("/cloud/project/QB2025_Yoo/Week2-Alpha")
#install.packages("vegan")
library("vegan")
```

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

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

```
## This is vegan 2.6-8
```

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)
str(BCI, max.level = 0)
```

```
## 'data.frame':    50 obs. of  225 variables:
##  - attr(*, "original.names")= chr [1:225] "Abarema.macradenium" "Acacia.melanoceras" "Acalypha.divers
```

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
}
x1 = BCI [1, ]
S.obs(x1)
```

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

```
specnumber(x1)
```

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

```
x = BCI [1:4, ]
S.obs(x)
```

```
##  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***: Yes, 'S.obs' and 'specnumber()' return the same value. The species richness of the first four sites are 93, 84, 90, and 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.

```
C <- function(x = ""){
  1 - rowSums(x == 1) / rowSums(x)
```

```
}

C(BCI)
```

```
##         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
```

```
CBCI <- C(BCI)
1-CBCI[1]
```

```
##          1
## 0.06919643
```

```
max(CBCI)
```

```
## [1] 0.9468504
```

```
min(CBCI)
```

```
## [1] 0.8705882
```

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: 0 to 1

Answer 2b: Good's coverage is 0, which means all taxa was represented by singletons.

Answer 2c: 1-Good's coverage, which is 0.06919643. About 7% of taxa in `site1` was represented by singletons.

Answer 2d: The site that has lowest coverage is site 23, with the value of 0.8705882. The site has highest portion of singletons. The site that has highest coverage is site 4, with the value of 0.9468504. The site has lowest portion of singletons.

Estimated richness

In the R code chunk below, do the following:

1. Load the microbial dataset (located in the `Week-2/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,]

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

S.chao1(soilbac1)
```

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

```
C(soilbac1)
```

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

```
C(x1) #coverage of site1
```

```
##         1
## 0.9308036
```

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: 13310 sequences

Answer 3b: 2628.514

Answer 3c: The coverage of soilbac 1 (0.6479471) was less than that of site1 (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(soilbac1)
```

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

```
S.chao1(x1)
```

```
##        1
## 119.6944
```

```
S.chao2("T1_1",soilbac1)
```

```
## T1_1
##  Inf
```

```
S.chao2(1,x1)
```

```
##   1
## Inf
```

```
S.ace(soilbac1)
```

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

```
S.ace(x1)
```

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

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***: Chao estimators is calculated using singletons and doubletons, while ACE is calculated by rare species which has 10 or fewer counts. Both estimators give consistent results that soilbac1 has greater estimated richness. I would choose to use ACE for soilbac1, and Chao for site1 because soilbac1 has many species with 11 or more counts, while site 1 has fewer counts for overall species.

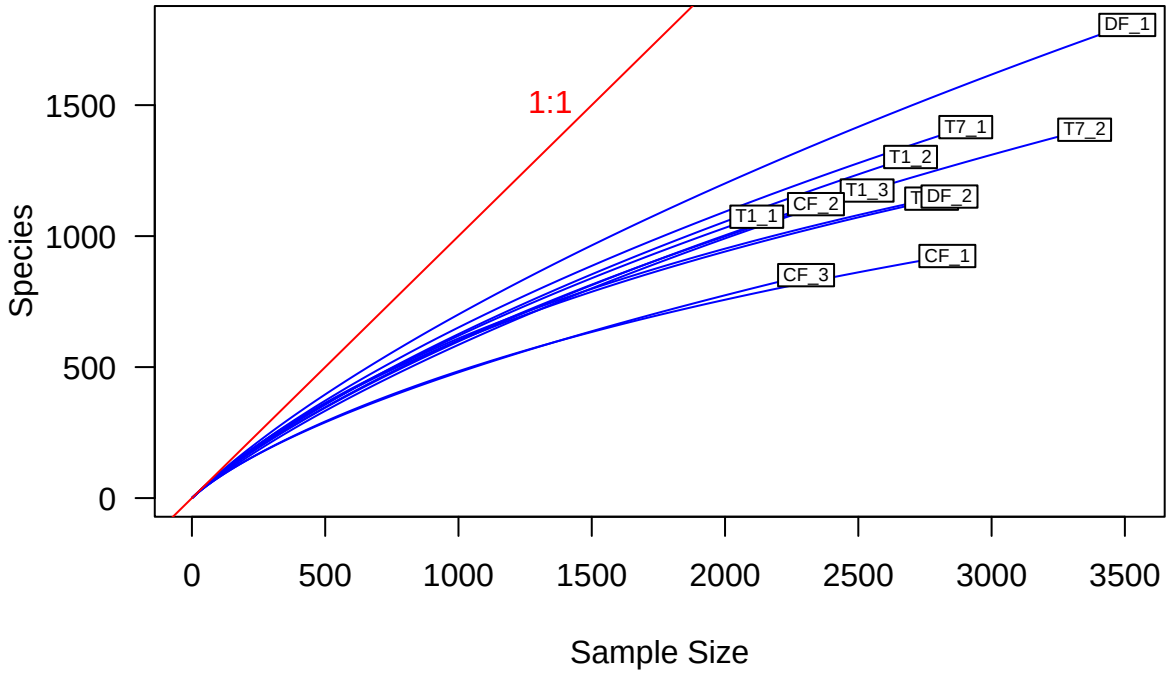
Rarefaction

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)
min.N <- min(rowSums(soilbac.t))
S.rarefy <- rarefy(x = soilbac.t, sample = min.N, se =TRUE)
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

```
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)
}
```

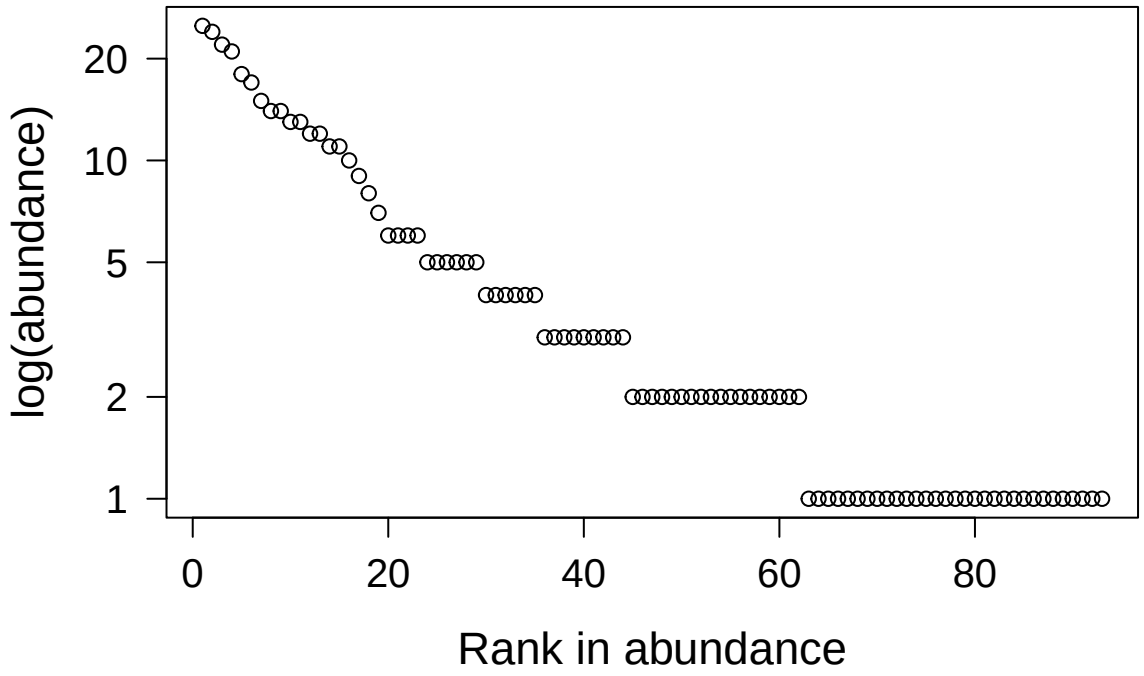
Now, let us examine the RAC for `site1` of the BCI data set.

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)"

```
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", axes = F,
     xlab = "Rank in abundance", ylab = "log(abundance)",
     las = 1, cex.lab = 1.4, cex.axis = 1.25)
box()
axis(side = 1, labels = T, cex.axis = 1.25)
axis(side = 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***: The log-scaled axis make the differences in abundance are less pronounced and visually reinforces the interpretation of evenness.

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)
}

SimpE(site1)
```

```
##         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)
```

```
## [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***: Both Simpson's evenness (0.4238232), and Smith and Wilson's evenness (0.5067211) agrees to each other's result and shows that the site1's evenness is moderately even.

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)
```

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

```
diversity(site1, index = "shannon")
```

```
## [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".

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

D.inv <- 1 / SimpD(site1)
D.sub <- 1 - SimpD(site1)

D.inv
```

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

```
D.sub
```

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

```
diversity(site1, "inv")
```

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

```
diversity(site1, "simp")
```

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

Fisher's α

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

1. Provide the code for calculating Fisher's $\boldsymbol{\alpha}$,
2. Calculate Fisher's $\boldsymbol{\alpha}$ for `site1` of BCI.

```
rac <- as.vector(site1[site1 > 0])
invD <- diversity(rac, "inv")
invD
```

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

```
Fisher <- fisher.alpha(rac)
Fisher
```

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

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 $\boldsymbol{\alpha}$ is much greater than other indices. It take into account incorporating richness and evenness, and sampling error.

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.

```
hill <- function(p, q) {
    return((sum(p^q))^(1 / (1 - q)))
  }
p <- site1/sum(site1)
qs <- c(0, 2)

Hills <- sapply(qs, function(q) hill(p, q))

Hills
```

```
## [1] 225.00000  39.41555
```

```
exp(ShanH(p))
```

```
## [1] 55.6127
```

> ***Answer 8a***: Hill numbers for q exponent 0, 1 and 2 is 225.00000, 55.6127, 39.41555.
>
> ***Answer 8b***: Every species are treated same when q is 0 regardless of rareness. However, by q increases, the influence of rare species decreases.

##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

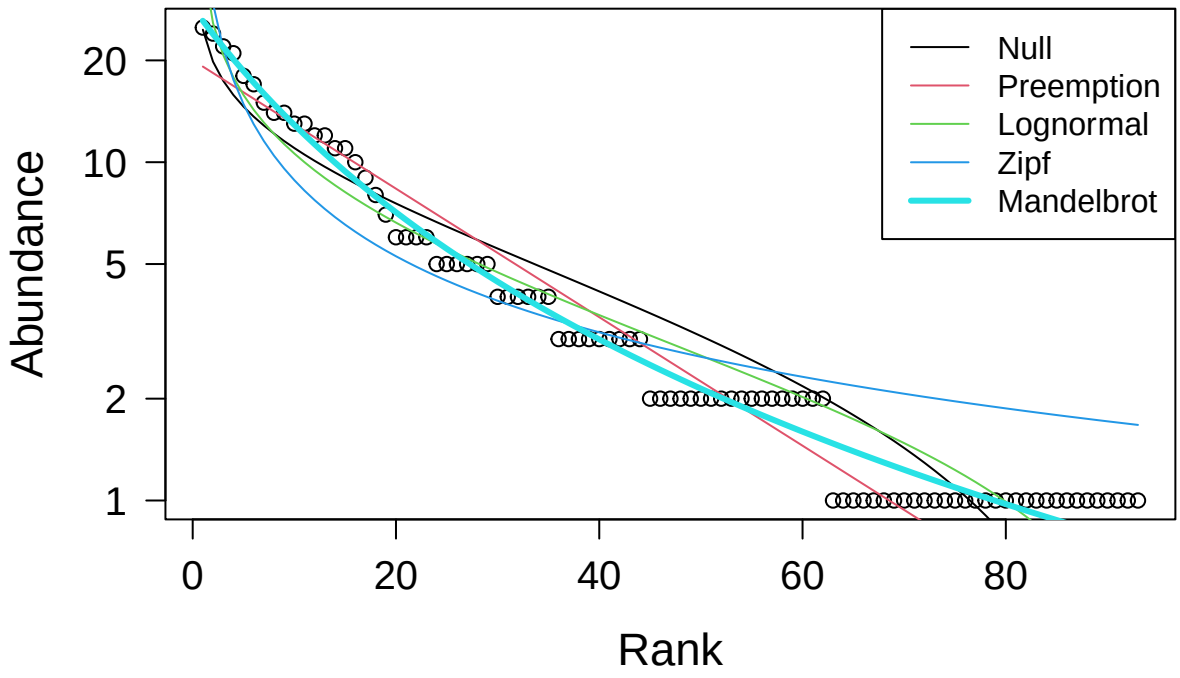
The RAC is a simple data structure that is both a vector of abundances. It is also a row in the site-by-species matrix (minus the zeros, i.e., absences).

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)
plot(RACresults, las = 1, cex.lab = 1.4, cex.axis = 1.25)
```



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***: Mandelbrot model best fits.
>
> ***Answer 9b***: No, we cannot make any inferences about them.

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 assume that N is proportional to the total resources.
>
> ***Answer 10b***: It is because the model assumes geometric decline in abundance with rank, so the model is a straight line in logarithmic scale.

Question 11: 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 because too many parameters make the model unstable and higly sensitive to small changes in the input dataset, which leads to poor predictions.

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.

```
SimpD_f <- function(x = ""){
  D = 0
  N = sum(x)
  for (n_i in x){
    D = D + (n_i*(n_i - 1)) / (N*(N-1))
  }
  return(D)
}

D_f <- SimpD_f(site1)
D_f.inv <- 1 / SimpD_f(site1)
D_f.sub <- 1 - SimpD_f(site1)

D_f
```

```
## [1] 0.02319032
```

```
D_f.inv
```

```
## [1] 43.12145
```

```
D_f.sub
```

```
## [1] 0.9768097
```

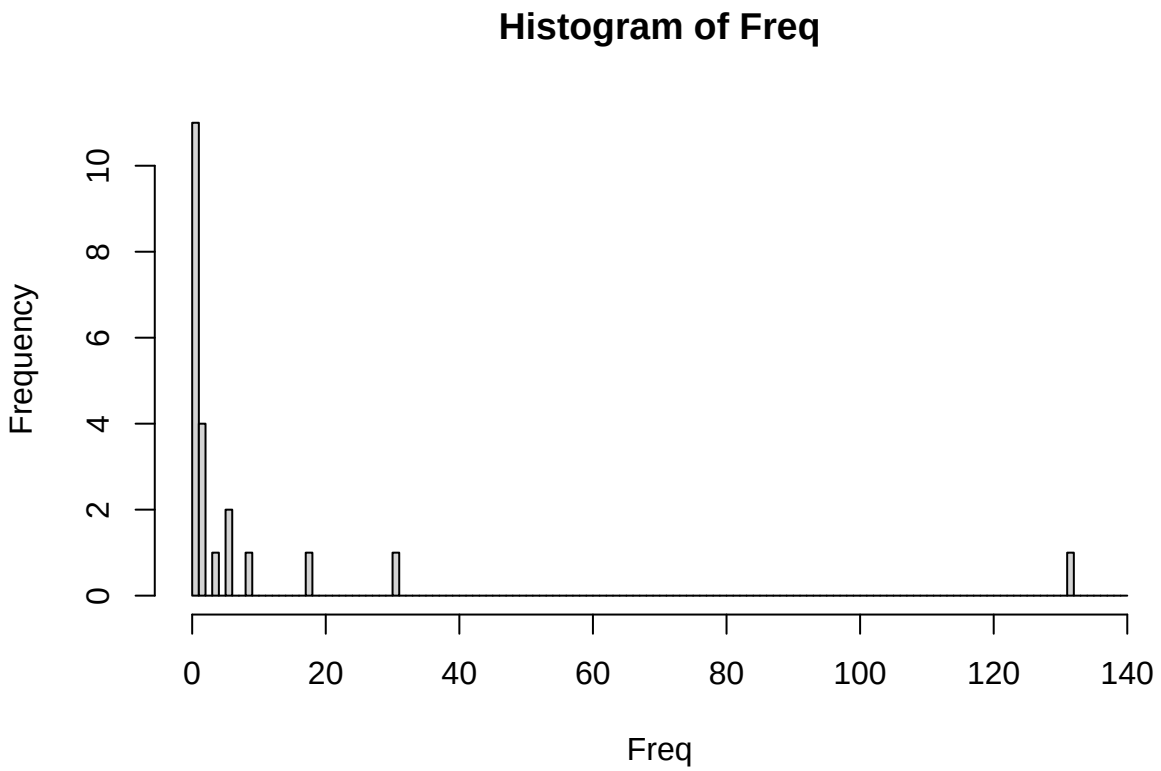
Answer: Simpson's D, 1 - D, and Simpson's inverse fir site 1 is 0.02319032, 0.9768097 and 43.12145.

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.

```
Freq <- table(t(site1))
Freq
```

```
##
##   0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  17  18  21  22
## 132  31  18   9   6   6   4   1   1   1   1   2   2   2   2   1   1   1   1   1
##  24  25
##   1   1
```

```
brk <- seq(0, 140, by = 1)
hist(Freq, breaks = brk)
```



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?

```
library(readr)
LDW <- read_csv("LDW_combcensus.csv")
```

```
## New names:
## Rows: 32271 Columns: 33
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (8): Latin, Family, Region, Genus, Species, Authority, IDLevel, myc_type dbl
## (23): ...1, No, Quadrat, PX, PY, TreeID, StemID, DBH1, DBH2, DBH3, X.1, ... lgl
## (2): syn, subsp
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```
Freq_LDW <- as.data.frame(table(LDW$Latin))
Freq_LDW <- as.data.frame(t(Freq_LDW))

colnames(Freq_LDW) <- Freq_LDW[1, ]
Freq_LDW <- Freq_LDW[-1, ]
```

Answer: There are 26 species of trees in the dataset. The dataset has the information on associated mycorrhizal fungi (AM vs. ECM), so the research question related to this variable will be interesting topic.

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 29th, 2025 at 12:00 PM (noon)**.