7. Worksheet: Diversity Synthesis

Anna Lennon

19 February, 2025

OVERVIEW

In this worksheet, you will conduct exercises that reinforce fundamental concepts of biodiversity. First, you will construct a site-by-species matrix by sampling confectionery taxa from a source community. Second, you will make a preference-profile matrix, reflecting each student's favorite confectionery taxa. With this primary data structure, you will then answer questions and generate figures using tools from previous weeks, along with wrangling techniques that we learned about in class.

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. Refer to previous handouts to help with developing of questions and writing of code.
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 `7.DiversitySynthesis_Worskheet.Rmd` and the PDF output of `Knitr` (`DiversitySynthesis_Worskheet.pdf`).

QUANTITATIVE CONFECTIONOLOGY

We will construct a site-by-species matrix using confectionery taxa (i.e, jelly beans). The instructors have created a **source community** with known abundance (N) and richness (S). Like a real biological community, the species abundances are unevenly distributed such that a few jelly bean types are common while most are rare. Each student will sample the source community and bin their jelly beans into operational taxonomic units (OTUs).

SAMPLING PROTOCOL: SITE-BY-SPECIES MATRIX

1. From the well-mixed source community, each student should take one Dixie Cup full of individuals.
2. At your desk, sort the jelly beans into different types (i.e., OTUs), and quantify the abundance of each OTU.
3. Working with other students, merge data into a site-by-species matrix with dimensions equal to the number of students (rows) and taxa (columns)
4. Create a worksheet (e.g., Google sheet) and share the site-by-species matrix with the class.

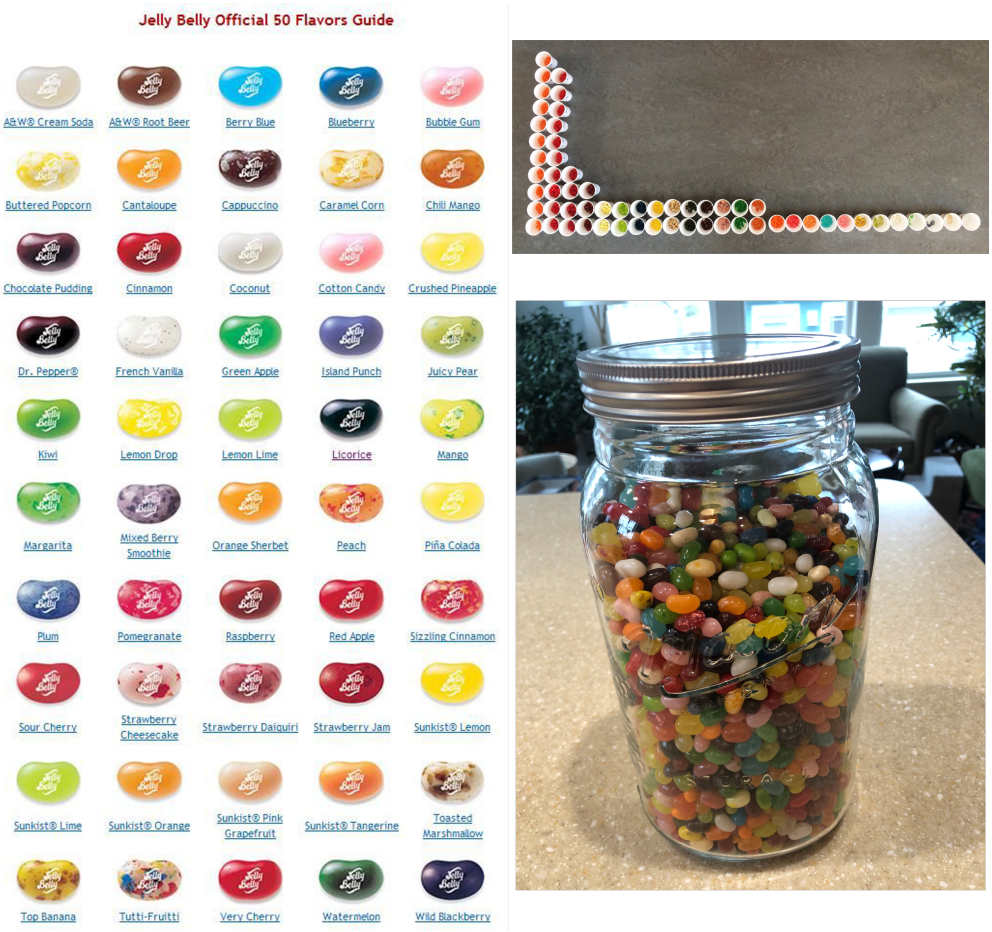


Figure 1: **Left**: taxonomic key, **Top right**: rank abundance distribution, **Bottom right**: source community

SAMPLING PROTOCOL: PREFERENCE-PROFILE MATRIX

1. With your individual sample only, each student should choose their top 5-10 preferred taxa based on flavor, color, sheen, etc.
2. Working with other students, merge data into preference-profile incidence matrix where 1 = preferred and 0 = non-preferred taxa.
3. Create a worksheet (e.g., Google sheet) and share the preference-profile matrix with the class.

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 `Week5-Confection/` 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_ALennon/Week5-Confection"
```

```
setwd("/cloud/project/QB2025_ALennon/Week5-Confection")
library(vegan)
```

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

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

```
library(ade4)
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(indicspecies)
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
jelly <- read.csv("/cloud/project/QB2025_ALennon/Week5-Confection/data/SS_Jelly.csv")
jelly <-jelly[1:12,1:52]
pref <- read.csv("/cloud/project/QB2025_ALennon/Week5-Confection/data/SS_preference.csv")
pref <-pref[c(1:3, 5:13),1:52]
source <-read.csv("/cloud/project/QB2025_ALennon/Week5-Confection/data/jelly.source.comm.csv")

`add.spec.scores.class` <-
  function(ordi,comm,method="cor.scores",multi=1,Rscale=F,scaling="1") {
    ordiscores <- scores(ordi,display="sites")
    n <- ncol(comm)
    p <- ncol(ordiscores)
    specscores <- array(NA,dim=c(n,p))
    rownames(specscores) <- colnames(comm)
    colnames(specscores) <- colnames(ordiscores)
    if (method == "cor.scores") {
      for (i in 1:n) {
        for (j in 1:p) {specscores[i,j] <- cor(comm[,i],ordiscores[,j],method="pearson")}
      }
    }
    if (method == "wa.scores") {specscores <- wascores(ordiscores,comm)}
    if (method == "pcoa.scores") {
      rownames(ordiscores) <- rownames(comm)
```

```
      eigenv <- ordi$eig
      accounted <- sum(eigenv)
      tot <- 2*(accounted/ordi$GOF[2])-(accounted/ordi$GOF[1])
      eigen.var <- eigenv/(nrow(comm)-1)
      neg <- length(eigenv[eigenv<0])
      pos <- length(eigenv[eigenv>0])
      tot <- tot/(nrow(comm)-1)
      eigen.percen <- 100*eigen.var/tot
      eigen.cumpercen <- cumsum(eigen.percen)
      constant <- ((nrow(comm)-1)*tot)^0.25
      ordiscores <- ordiscores * (nrow(comm)-1)^-0.5 * tot^-0.5 * constant
      p1 <- min(p, pos)
      for (i in 1:n) {
        for (j in 1:p1) {
          specscores[i,j] <- cor(comm[,i],ordiscores[,j])*sd(comm[,i])/sd(ordiscores[,j])
          if(is.na(specscores[i,j])) {specscores[i,j]<-0}
        }
      }
      if (Rscale==T && scaling=="2") {
        percen <- eigen.var/tot
        percen <- percen^0.5
        ordiscores <- sweep(ordiscores,2,percen,"/")
        specscores <- sweep(specscores,2,percen,"*")
      }
      if (Rscale==F) {
        specscores <- specscores / constant
        ordiscores <- ordi$points
      }
      ordi$points <- ordiscores
      ordi$eig <- eigen.var
      ordi$eig.percen <- eigen.percen
      ordi$eig.cumpercen <- eigen.cumpercen
      ordi$eigen.total <- tot
      ordi$R.constant <- constant
      ordi$Rscale <- Rscale
      ordi$scaling <- scaling
    }
    specscores <- specscores * multi
    ordi$cproj <- specscores
    return(ordi)
  }
```

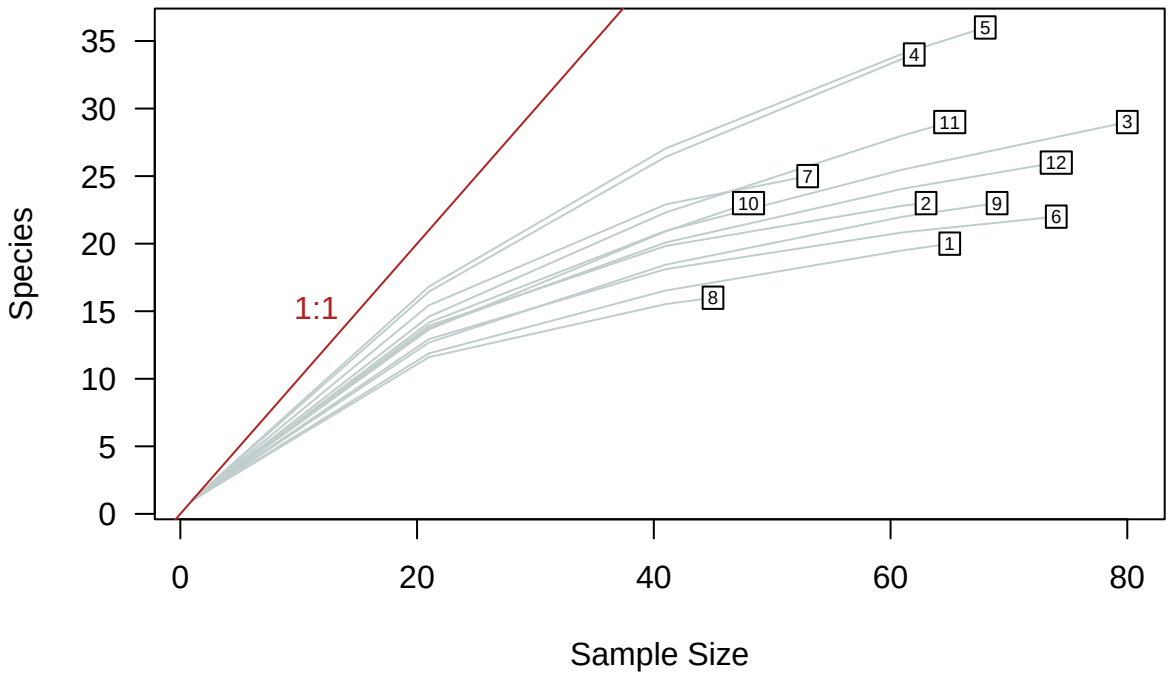
DATA ANALYSIS

Question 1: In the space below, generate a rarefaction plot, for all samples of the source community. Based on these results, discuss how individual vs. collective sampling efforts capture the diversity of the source community.

```
#observed richness
S.obs <- function(x = ""){
   rowSums(x > 0) *1
}
jelly.df <- as.data.frame(jelly[,-1])
S.obs(jelly.df)
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12
## 20 23 29 34 36 22 25 16 23 23 29 26
```

```
min.N <-min(rowSums(jelly.df))
jelly.rare <- rarefy(x = jelly.df, sample = min.N, se = TRUE)
rarecurve(x = jelly.df, step = 20, col = "azure3", cex = .6, las =1)
abline(0,1, col = "firebrick")
text(15, 15, "1:1", pos = 2, col = "firebrick")
```



Answer 1: Collectively, the group sampling efforts managed to capture some the diversity of the sample community. However, there is a braod range to an indvidual's ability to capture the diversity of the jelly bean samples. For example, 4/5 (Jocelyn and El) captured the most diversity of the sample community while 1/8 (Bryan and Madison) captured the least. If done indvidually, the true diversity of the sample community would not have been captured. Together, the class was able to capture some of the sample community. There is still a curve to the samples, highlighting the fact some of the diversity was lost in sampling.

Question 2: Starting with the site-by-species matrix, visualize beta diversity. In the code chunk below, conduct principal coordinates analyses (PCoA) using both an abundance- and incidence-based resemblance matrix. Plot the sample scores in species space using different colors, symbols, or labels. Which "species" are contributing the patterns in the ordinations? How does the choice of resemblance matrix affect your interpretation?

```
jelly.dj <- vegdist(jelly.df, method = "jaccard", binary = TRUE)  # Incidence
jelly.bc <- vegdist(jelly.df, method = "bray")  # Abundance

jelly.pcoa.dj <- cmdscale(jelly.dj, eig = TRUE, k = 3)
jelly.pcoa.bc <- cmdscale(jelly.bc, eig = TRUE, k = 3)

var1 <- round(jelly.pcoa.dj$eig[1] / sum(jelly.pcoa.dj$eig), 3) * 100
var2 <- round(jelly.pcoa.dj$eig[2] / sum(jelly.pcoa.dj$eig), 3) * 100

jellyREL <- jelly.df
for(i in 1:nrow(jelly.df)) {
```

```
  jellyREL[i, ] <- jelly.df[i, ] / sum(jelly.df[i, ])
}

species_contrib <- colMeans(jellyREL, na.rm = TRUE)

top_species <- names(species_contrib[species_contrib > 0.05])

species_numbers <- seq_len(nrow(jelly.df))


jelly.pcoa.dj <- add.spec.scores.class(jelly.pcoa.dj, jellyREL, method = "pcoa.scores")
```

```
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
```

```
jelly.pcoa.bc <- add.spec.scores.class(jelly.pcoa.bc, jellyREL, method = "pcoa.scores")
```

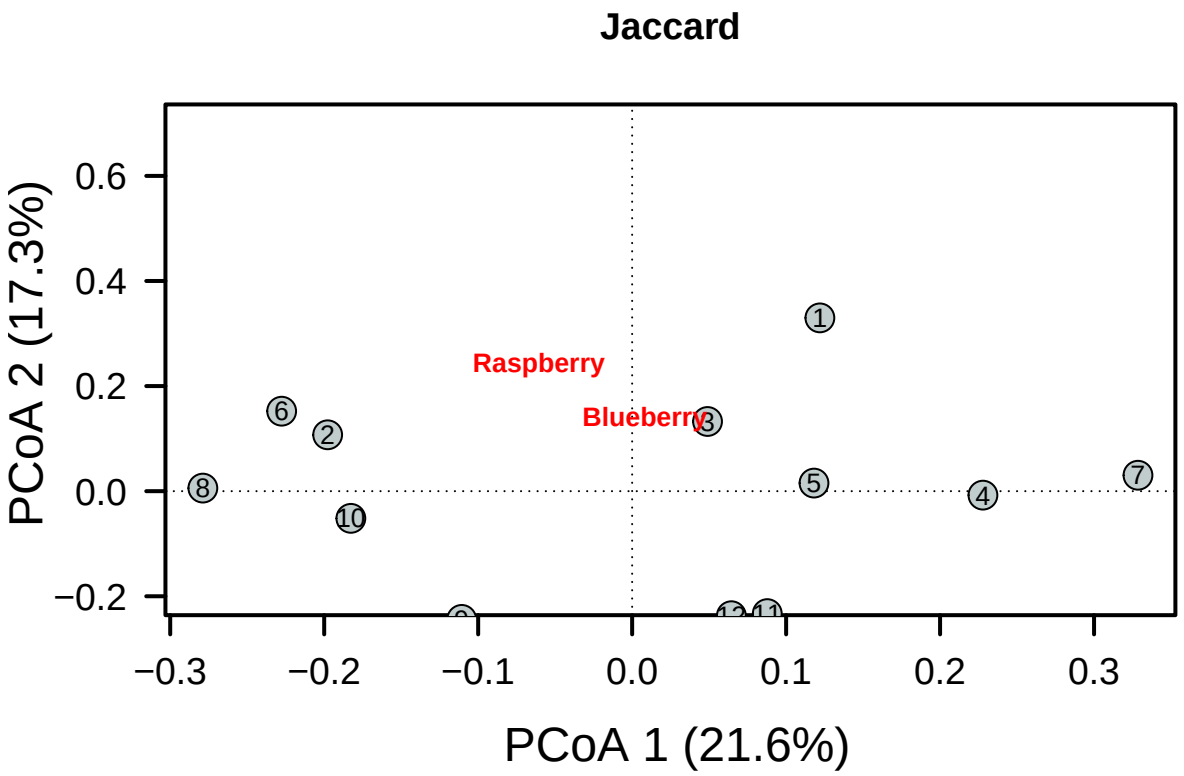
```
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
```

```
jelly.pcoa.dj$cproj <- jelly.pcoa.dj$cproj[top_species, ]
jelly.pcoa.bc$cproj <- jelly.pcoa.bc$cproj[top_species, ]

#Jaccard PCoA
plot(jelly.pcoa.dj$points[, 1], jelly.pcoa.dj$points[, 2], ylim = c(-0.2, 0.7),
     xlab = paste(" PCoA 1 (", var1, "%)", sep = ""),
     ylab = paste(" PCoA 2 (", var2, "%)", sep = ""),
     main = "Jaccard",
     pch = 16, cex = 2.0, type = "n", cex.lab = 1.5,
     cex.axis = 1.2, axes = FALSE)

axis(side = 1, labels = TRUE, lwd.ticks = 2, cex.axis = 1.2, las = 1)
axis(side = 2, labels = TRUE, lwd.ticks = 2, cex.axis = 1.2, las = 1)
abline(h = 0, v = 0, lty = 3)
box(lwd = 2)

points(jelly.pcoa.dj$points[, 1], jelly.pcoa.dj$points[, 2],
       pch = 21, cex = 2, col = "black", bg = "azure3")
text(jelly.pcoa.dj$points[, 1], jelly.pcoa.dj$points[, 2],
     labels = species_numbers, cex = .8, col = "black")
text(jelly.pcoa.dj$cproj[, 1], jelly.pcoa.dj$cproj[, 2],
     labels = row.names(jelly.pcoa.dj$cproj), col = "red", cex = 0.8, font = 2)
```

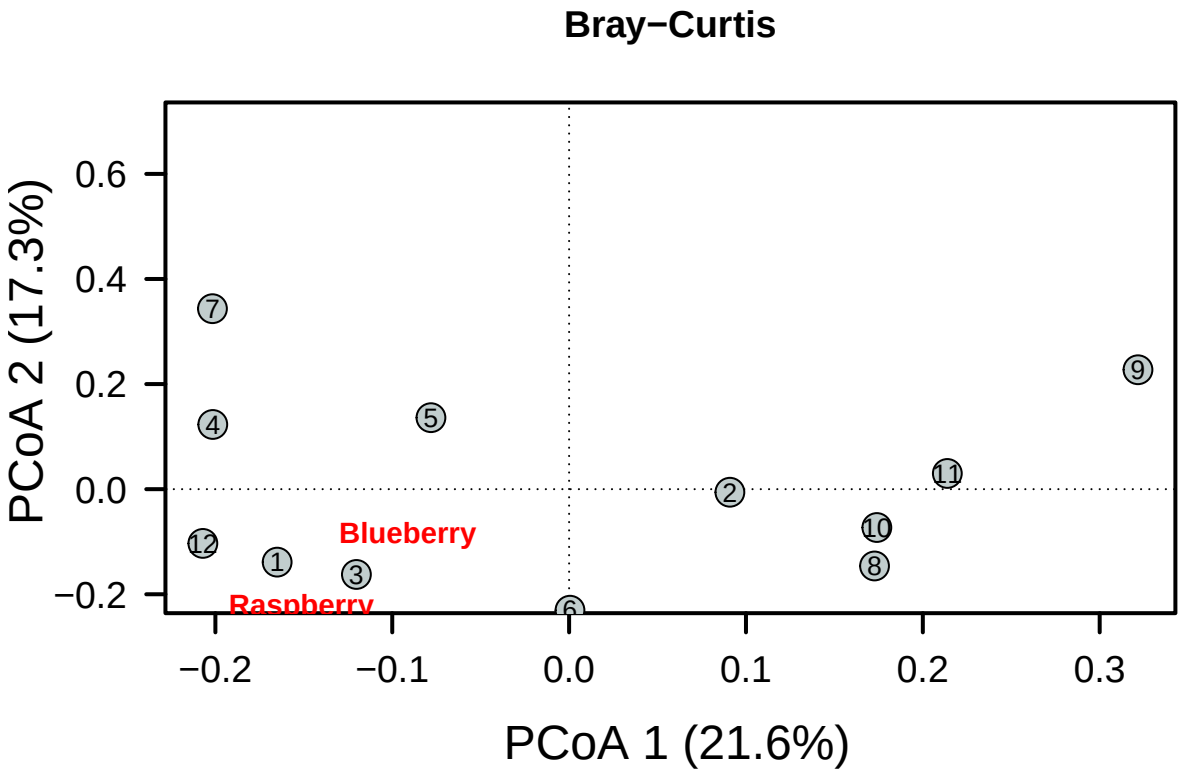



```r
#Bray-Curtis PCoA
plot(jelly.pcoa.bc$points[, 1], jelly.pcoa.bc$points[, 2], ylim = c(-0.2, 0.7),
     xlab = paste(" PCoA 1 (", var1, "%)", sep = ""),
     ylab = paste(" PCoA 2 (", var2, "%)", sep = ""),
     main = "Bray-Curtis",
     pch = 16, cex = 2.0, type = "n", cex.lab = 1.5,
     cex.axis = 1.2, axes = FALSE)

axis(side = 1, labels = TRUE, lwd.ticks = 2, cex.axis = 1.2, las = 1)
axis(side = 2, labels = TRUE, lwd.ticks = 2, cex.axis = 1.2, las = 1)
abline(h = 0, v = 0, lty = 3)
box(lwd = 2)

# Plot points with species numbers
points(jelly.pcoa.bc$points[, 1], jelly.pcoa.bc$points[, 2],
       pch = 21, cex = 2, col = "black", bg = "azure3")
text(jelly.pcoa.bc$points[, 1], jelly.pcoa.bc$points[, 2],
     labels = species_numbers, cex = .8, col = "black")

# Add text labels only for influential species
text(jelly.pcoa.bc$cproj[, 1], jelly.pcoa.bc$cproj[, 2],
     labels = row.names(jelly.pcoa.bc$cproj), col = "red", cex = 0.9, font = 2)
```



>

Answer 2: >The most influential species are rasberry and blueberry. However, how this is represented in the PcoA varies depending on the source resemblence matrix. In the Bray-Curtis PCoA, there is a wide scatter of points with the only one cluster (10 and 8) forming. Comparatively, the Jaccard based PCoA has a larger cluster to the left of the plot as well as a small cluster at the bottom of the graph. Rasberry and blueberry are still the most influential but rasberry does not appear as a point in the Jaccard graph.

Question 3 Using the preference-profile matrix, determine the most popular jelly bean in the class using a control structure (e.g., for loop, if statement, function, etc).

```
pref.obs <- function(x) {
  return(colSums(x > 0, na.rm = TRUE))
}
pref.L <-pref[,-1]
pref.obs <-pref.obs(pref.L)
pref.df <-as.data.frame(pref.obs)
```

> ***Answer 3***: Wild Blackberry is the prefered Jelly Bean flavor of the QB students. This answer is incorrect. It should be coconut ;).

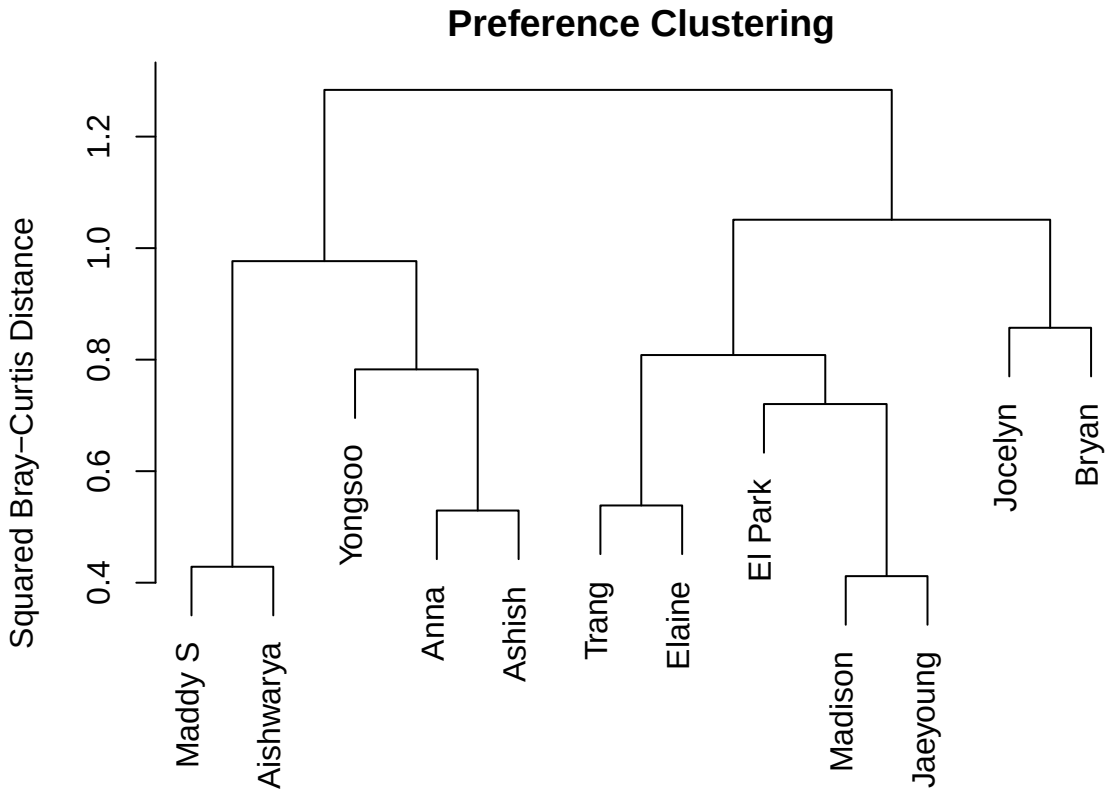
Question 4 In the code chunk below, identify the student in QB who has a preference-profile that is most like yours. Quantitatively, how similar are you to your "jelly buddy"? Visualize the preference profiles of the class by creating a cluster dendogram. Label each terminal node (a.k.a., tip or "leaf") with the student's name or initials. Make some observations about the preference-profiles of the class.

```
pref.bc <- vegdist(pref.L, method = "bray")
student_names <- pref[,1]
dist_matrix <- as.matrix(pref.bc)

rownames(dist_matrix) <- colnames(dist_matrix) <- student_names

pref.ward <-hclust(pref.bc, method = "ward.D2")
par(mar = c(1,5,2,2) +0.1)
```

```r
plot(pref.ward, main = "Preference Clustering",
     ylab = "Squared Bray-Curtis Distance",
     labels = student_names)
```



Answer 4: Using a Bray-Curtis dissimilarity matrix, Ashish is the most similar to me. Youngsoo also has similar preferences to me but Ashish and I are much closer. Maddy S and Bryan are the most dissimilar being the farthest away from one another. In two cases, (Youngsoo-Anna-Ashish; and El-Madison-Jaeyoung) there is a particular pattern where Youngsoo and El serve as an "outgroup" to two "sister groups" (Youngsoo-Anna; Madison-Jaeyoung). This is unique as El and Youngsoo do not have a very close "jelly buddy;" rather their buddy is less similar to them compared to other buddy groups.

SUBMITTING YOUR ASSIGNMENT

Use Knitr to create a PDF of your completed `7.DiversitySynthesis_Worksheet.Rmd` document, push it to GitHub, and create a pull request. Please make sure your updated repo includes both the pdf and RMarkdown files.

Unless otherwise noted, this assignment is due on **Wednesday, February 19th, 2025 at 12:00 PM (noon)**.