

6. Worksheet: Among Site (Beta) Diversity – Part 1

Carter Stancil; Z620: Quantitative Biodiversity, Indiana University

09 February, 2026

OVERVIEW

In this worksheet, we move beyond the investigation of within-site α -diversity. We will explore β -diversity, which is defined as the diversity that occurs among sites. This requires that we examine the compositional similarity of assemblages that vary in space or time.

After completing this exercise you will know how to:

1. formally quantify β -diversity
2. visualize β -diversity with heatmaps, cluster analysis, and ordination
3. test hypotheses about β -diversity using multivariate statistics

Directions:

1. In the Markdown version of this document in your cloned repo, change “Student Name” on line 3 (above) with 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 examples of proper scripting needed to carry out the exercises.
4. Answer questions in the worksheet. Space for your answers is provided in this document and is 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 (**6.BetaDiversity_1_Worksheet.Rmd**) with all code blocks filled out and questions answered) and the PDF output of Knitr (**6.BetaDiversity_1_Worksheet.pdf**).

The completed exercise is due on **Wednesday, February 5th, 2025 before 12:00 PM (noon)**.

1) R SETUP

Typically, the first thing you will do in either an R script or an RMarkdown file is setup your environment. This includes things such as setting the working directory and loading any packages that you will need.

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 Week3-Beta/ folder folder, and
- 4) Load the `vegan` R package (be sure to install first if you have not already).

```

rm(list = ls())
getwd()

## [1] "C:/github/QB2026_Stancil/QB2026/Week4-Beta"

setwd("../Week4-Beta/") #change to wherever your wd is

package.list <- c('vegan', 'ade4', 'viridis', 'gplots', 'indicspecies')

for (package in package.list) {
  if (!require(package, character.only = TRUE, quietly = TRUE)) {
    install.packages(package)
    library(package, character.only = TRUE)
  }
}

## Warning: package 'vegan' was built under R version 4.5.2

## Warning: package 'permute' was built under R version 4.5.2

## Warning: package 'ade4' was built under R version 4.5.2

## Warning: package 'gplots' was built under R version 4.5.2

## -----
## gplots 3.3.0 loaded:
##   * Use citation('gplots') for citation info.
##   * Homepage: https://talgalili.github.io/gplots/
##   * Report issues: https://github.com/talgalili/gplots/issues
##   * Ask questions: https://stackoverflow.com/questions/tagged/gplots
##   * Suppress this message with: suppressPackageStartupMessages(library(gplots))
## -----


## 
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
## 
##     lowess

## Warning: package 'indicspecies' was built under R version 4.5.2

```

```
package.list
```

```
## [1] "vegan"          "ade4"           "viridis"        "gplots"         "indicspecies"  
  
#remove.packages("BiodiversityR")  
#install.packages(c("sem", "rgl", "multcomp"))  
#install.packages(c("lmtest", "leaps", "aplypack"))  
#install.packages("BiodiversityR", dependencies = TRUE, clean = TRUE)  
  
## TURN VOLUME OF COMPUTER ON  
  
library(BiodiversityR)  
  
## Warning: package 'BiodiversityR' was built under R version 4.5.2  
  
## Loading required package: tcltk  
  
## BiodiversityR 2.17-4: Use command BiodiversityRGUI() to launch the Graphical User Interface;  
## to see changes use BiodiversityRGUI(changeLog=TRUE, backward.compatibility.messages=TRUE)  
  
## THERE WILL BE A "POP-UP" BUT IT WILL NOT ACTUALLY POP UP  
## MINIMIZE RSTUDIO --> SEE POP-UP AND APPROVE DOWNLOAD FROM CRAN
```

2) LOADING DATA

Load dataset

In the R code chunk below, do the following:

1. load the `doubs` dataset from the `ade4` package, and
2. explore the structure of the dataset.

```
# note, please do not print the dataset when submitting  
data(doubs)  
str(doubs)
```

```
## List of 4  
## $ env    :'data.frame': 30 obs. of  11 variables:  
##   ..$ dfs: num [1:30] 3 22 102 185 215 ...  
##   ..$ alt: num [1:30] 934 932 914 ...  
##   ..$ slo: num [1:30] 6.18 3.43 3.64 ...  
##   ..$ flo: num [1:30] 84 100 180 253 ...  
##   ..$ pH : num [1:30] 79 80 83 80 ...  
##   ..$ har: num [1:30] 45 40 52 72 ...  
##   ..$ pho: num [1:30] 1 2 5 10 ...  
##   ..$ nit: num [1:30] 20 20 22 21 ...  
##   ..$ amm: num [1:30] 0 10 5 0 ...  
##   ..$ oxy: num [1:30] 122 103 105 110 ...  
##   ..$ bdo: num [1:30] 27 19 35 13 62 ...  
## $ fish   :'data.frame': 30 obs. of  27 variables:
```

```

## ..$ Cogo: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Satr: num [1:30] 3 5 5 4 2 3 5 0 0 1 ...
## ..$ Phph: num [1:30] 0 4 5 5 3 4 4 0 1 4 ...
## ..$ Neba: num [1:30] 0 3 5 5 2 5 5 0 3 4 ...
## ..$ Thth: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Teso: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Chna: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Chto: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Lele: num [1:30] 0 0 0 0 5 1 1 0 0 2 ...
## ..$ Lece: num [1:30] 0 0 0 1 2 2 1 0 5 2 ...
## ..$ Baba: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Spbi: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Gogo: num [1:30] 0 0 0 1 2 1 0 0 0 1 ...
## ..$ Eslu: num [1:30] 0 0 1 2 4 1 0 0 0 0 ...
## ..$ Pepl: num [1:30] 0 0 0 2 4 1 0 0 0 0 ...
## ..$ Rham: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Legi: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Scer: num [1:30] 0 0 0 0 2 0 0 0 0 0 ...
## ..$ Cyca: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Titi: num [1:30] 0 0 0 1 3 2 0 0 1 0 ...
## ..$ Abbr: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Icme: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Acce: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Ruru: num [1:30] 0 0 0 0 5 1 0 0 4 0 ...
## ..$ Blbj: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Alal: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## ..$ Anan: num [1:30] 0 0 0 0 0 0 0 0 0 0 ...
## $ xy      :'data.frame': 30 obs. of  2 variables:
##   ..$ x: num [1:30] 88 94 102 100 106 112 114 110 136 168 ...
##   ..$ y: num [1:30] 7 14 18 28 39 51 61 76 100 112 ...
## $ species:'data.frame': 27 obs. of  4 variables:
##   ..$ Scientific: chr [1:27] "Cottus gobio" "Salmo trutta fario" "Phoxinus phoxinus" "Nemacheilus ba...
##   ..$ French    : chr [1:27] "chabot" "truite fario" "vairon" "loche franche" ...
##   ..$ English   : chr [1:27] "european bullhead" "brown trout" "minnow" "stone loach" ...
##   ..$ code      : Factor w/ 27 levels "Abbr","Acce",...: 9 22 19 17 26 25 7 8 16 14 ...

```

Question 1: Describe some of the attributes of the **doubs** dataset.

- How many objects are in **doubs**?
- How many fish species are there in the **doubs** dataset?
- How many sites are in the **doubs** dataset?

Answer 1a: 4

Answer 1b: 27

Answer 1c: 30

Visualizing the Doubs River Dataset

Question 2: Answer the following questions based on the spatial patterns of richness (i.e., α -diversity) and Brown Trout (*Salmo trutta*) abundance in the Doubs River.

- How does fish richness vary along the sampled reach of the Doubs River?

- b. How does Brown Trout (*Salmo trutta*) abundance vary along the sampled reach of the Doubs River?
- c. What do these patterns say about the limitations of using richness when examining patterns of biodiversity?

```

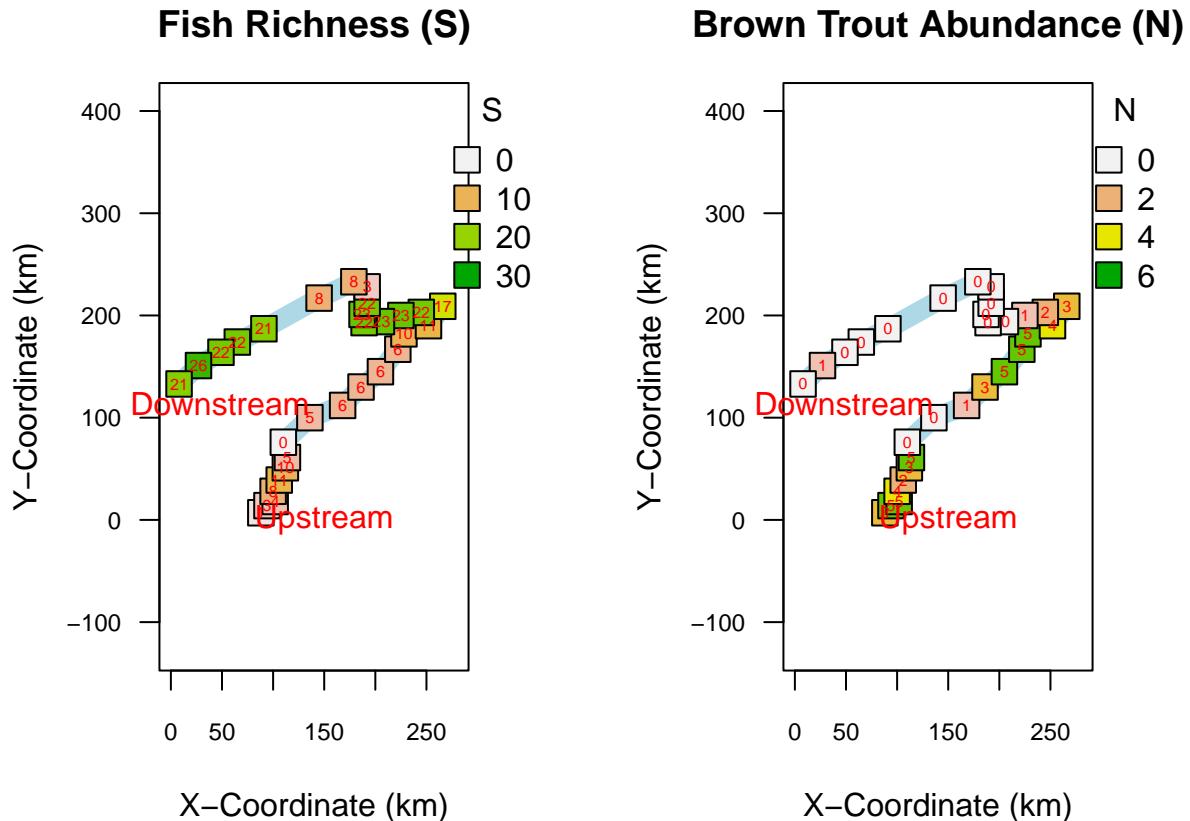
# Define Plot Parameters
opar <- par(no.readonly = TRUE)
par(mfrow = c(1,2), mar = c(4, 4, 3, 4) + 0.1, xpd = TRUE)

# Stream Fish
spa.S <- specnumber(doubs$fish)
spa.S.color <- rev(terrain.colors(31))      # Define Richness Color Pallette
spa.N.color <- rev(terrain.colors(7))        # Define Abundance Color Pallette

# Stream Fish Richness
plot(doubs$xy, asp = 1, type = 'l', col = "light blue", lwd = 10, las = 1,
      cex.axis = 0.75, xlim = c(0,280), ylim = c(0,280),
      main = "Fish Richness (S)",
      xlab = "X-Coordinate (km)", ylab = "Y-Coordinate (km)")
points(doubs$xy, pch = 22, cex=2, bg = spa.S.color[spa.S + 1])
text(doubs$xy, as.character(spa.S), cex = 0.5, col = "red")
text(150, 0, "Upstream", cex = 1, col = "red")
text(48, 114, "Downstream", cex = 1, col = "red")
legend("topright", inset=c(-0.25, 0), pch = 22, pt.cex = 2, bty = 'n',
       title = "S", legend = seq(0, 30, 10),
       pt.bg = spa.S.color[seq(1, 31, 10)])

# Brown Trout Abundance
plot(doubs$xy, asp = 1, type = 'l', col = "light blue", lwd = 10,
      las = 1, cex.axis = 0.75, xlim = c(0, 280), ylim = c(0, 280),
      main = "Brown Trout Abundance (N)",
      xlab = "X-Coordinate (km)", ylab = "Y-Coordinate (km)")
points(doubs$xy, pch = 22, cex = 2, bg = spa.N.color[doubs$fish$Satr + 1])
text(doubs$xy, as.character(doubs$fish$Satr), cex = 0.5, col="red")
text(150, 0, "Upstream", cex = 1, col = "red")
text(48, 114, "Downstream", cex = 1, col = "red")
legend("topright", inset=c(-0.25, 0), pch = 22, pt.cex = 2, bty = 'n',
       title = "N", legend = seq(0, 6, 2), pt.bg = spa.N.color[seq(1, 7, 2)])

```



3) QUANTIFYING BETA-DIVERSITY

In the R code chunk below, do the following:

- write a function (`beta.w()`) to calculate Whittaker's β -diversity (i.e., β_w) that accepts a site-by-species matrix with optional arguments to specify pairwise turnover between two sites, and
- use this function to analyze various aspects of β -diversity in the Doubs River.

```
beta.w <- function(site.by.species = ""){
  SbyS.pa <- decostand(site.by.species, method = "pa")
  # convert to presence-absence
  S <- ncol(SbyS.pa[,which(colSums(SbyS.pa) > 0)])
  # number of species in the region
  a.bar <- mean(specnumber(SbyS.pa))
  # average richness at each site
  b.w <- round(S/a.bar, 3)
  # round to 3 decimal places
  return(b.w)
}
```

```

}

beta.w <- function(site.by.species = "", sitenum1 = "", sitenum2 = "",
pairwise = FALSE){

# ONLY if we specify pairwise as TRUE, do this:
if (pairwise == TRUE){

# As a check, let's print an error if we do not provide needed arguments
if (sitenum1 == "" | sitenum2 == "") {
  print("Error: please specify sites to compare")
  return(NA)}

# If our function made it this far, let us calculate pairwise beta diversity
site1 = site.by.species[sitenum1,]
# Select site 1
site2 = site.by.species[sitenum2,]
# Select site 2
site1 = subset(site1, select = site1 > 0)
# Removes absences
site2 = subset(site2, select = site2 > 0)
# Removes absences
gamma = union(colnames(site1), colnames(site2))
# Gamma species pool
s = length(gamma)
# Gamma richness
a.bar = mean(c(specnumber(site1), specnumber(site2)))
# Mean sample richness
b.w = round(s/a.bar - 1, 3)
return(b.w)
}

# OTHERWISE pairwise defaults to FALSE, so do this, like before:
else{
  SbyS.pa <- decostand(site.by.species, method = "pa")
  # convert to presence-absence
  S <- ncol(SbyS.pa[,which(colSums(SbyS.pa) > 0)])
  # number of species in region
  a.bar <- mean(specnumber(SbyS.pa))
  # average richness at each site
  b.w <- round(S/a.bar, 3)
  return(b.w)
}
}

```

```
beta.w(doubs$fish)
```

```
## [1] 2.16
```

```
beta.w(
  site.by.species = doubs$fish,
  sitenum1 = "1",
```

```

    sitenum2 = "10",
    pairwise = TRUE
)

```

```
## [1] 0.714
```

Question 3: Using your `beta.w()` function above, answer the following questions:

- Describe how local richness (α) and turnover (β) contribute to regional (γ) fish diversity in the Doubs.
- Is the fish assemblage at site 1 more similar to the one at site 2 or site 10?
- Using your understanding of the equation $\beta_w = \gamma/\alpha$, how would your interpretation of β change if we instead defined beta additively (i.e., $\beta = \gamma - \alpha$)?

Answer 3a: Gamma diversity = 27 species total along the river. Alpha diversity varies with multiple fish species at each site, and the sites themselves differ (Beta diversity). **Answer 3b:** Site 1 is more similar to site 2 than site 10. **Answer 3c:** If we used the additive definition, we would literally be discussing specific species as units. (i.e., Species A and B are missing between sites)

The Resemblance Matrix

In order to quantify β -diversity for more than two samples, we need to introduce a new primary ecological data structure: the **Resemblance Matrix**.

Question 4: How do incidence- and abundance-based metrics differ in their treatment of rare species?

Answer 4: Abundance-based metrics may underrepresent rare species, while incidence-based metrics may overrepresent rare species, because species are either present or absent.

In the R code chunk below, do the following:

- make a new object, `fish`, containing the fish abundance data for the Doubs River,
- remove any sites where no fish were observed (i.e., rows with sum of zero),
- construct a resemblance matrix based on Sørensen's Similarity ("fish.ds"), and
- construct a resemblance matrix based on Bray-Curtis Distance ("fish.db").

```

fish <- doubs$fish
fish

```

```

##   Cogo Satr Phph Neba Thth Teso Chna Chto Lele Lece Baba Spbi Gogo Eslu Pefl
## 1    0    3    0    0    0    0    0    0    0    0    0    0    0    0    0    0
## 2    0    5    4    3    0    0    0    0    0    0    0    0    0    0    0    0
## 3    0    5    5    5    0    0    0    0    0    0    0    0    0    0    0    1
## 4    0    4    5    5    0    0    0    0    0    1    0    0    1    2    2
## 5    0    2    3    2    0    0    0    0    5    2    0    0    2    4    4
## 6    0    3    4    5    0    0    0    0    1    2    0    0    1    1    1
## 7    0    5    4    5    0    0    0    0    1    1    0    0    0    0    0
## 8    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
## 9    0    0    1    3    0    0    0    0    0    5    0    0    0    0    0
## 10   0    1    4    4    0    0    0    0    2    2    0    0    1    0    0
## 11   1    3    4    1    1    0    0    0    0    1    0    0    0    0    0

```

## 12	2	5	4	4	2	0	0	0	1	0	0	0	0	0
## 13	2	5	5	2	3	2	0	0	0	0	0	0	0	0
## 14	3	5	5	4	4	3	0	0	0	1	1	0	1	0
## 15	3	4	4	5	2	4	0	0	3	3	2	0	2	0
## 16	2	3	3	5	0	5	0	4	5	2	2	1	2	1
## 17	1	2	4	4	1	2	1	4	3	2	3	4	1	1
## 18	1	1	3	3	1	1	1	3	2	3	3	3	2	1
## 19	0	0	3	5	0	1	2	3	2	1	2	2	4	1
## 20	0	0	1	2	0	0	2	2	2	3	4	3	4	2
## 21	0	0	1	1	0	0	2	2	2	2	4	2	5	3
## 22	0	0	0	1	0	0	3	2	3	4	5	1	5	3
## 23	0	0	0	0	0	0	0	0	0	1	0	0	0	0
## 24	0	0	0	0	0	0	1	0	0	2	0	0	1	0
## 25	0	0	0	0	0	0	0	0	1	1	0	0	2	1
## 26	0	0	0	1	0	0	1	0	1	2	2	1	3	2
## 27	0	0	0	1	0	0	1	1	2	3	4	1	4	4
## 28	0	0	0	1	0	0	1	1	2	4	3	1	4	3
## 29	0	1	1	1	1	1	2	2	3	4	5	3	5	4
## 30	0	0	0	0	0	1	2	3	3	3	5	5	4	5
##	Rham	Legi	Scer	Cyca	Titi	Abbr	Icme	Acce	Ruru	Bbj	Alal	Anan		
## 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 4	0	0	0	0	1	0	0	0	0	0	0	0	0	0
## 5	0	0	2	0	3	0	0	0	5	0	0	0	0	0
## 6	0	0	0	0	2	0	0	0	1	0	0	0	0	0
## 7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 9	0	0	0	0	1	0	0	0	4	0	0	0	0	0
## 10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 11	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 12	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 13	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 14	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 15	0	0	0	0	1	0	0	0	0	0	0	0	0	0
## 16	0	1	0	1	1	0	0	0	1	0	0	0	0	0
## 17	1	1	0	1	1	0	0	0	2	0	2	1		
## 18	2	1	0	1	1	0	0	1	2	0	2	1		
## 19	2	1	1	1	2	1	0	1	5	1	3	1		
## 20	3	2	2	1	4	1	0	2	5	2	5	2		
## 21	3	2	2	2	4	3	1	3	5	3	5	2		
## 22	3	3	2	3	4	4	2	4	5	4	5	2		
## 23	0	0	0	0	0	0	0	0	1	0	2	0		
## 24	0	1	0	0	0	0	0	2	2	1	5	0		
## 25	0	0	1	0	0	0	0	1	1	0	3	0		
## 26	2	2	1	1	3	2	1	4	4	2	5	2		
## 27	3	3	1	2	5	3	2	5	5	4	5	3		
## 28	4	4	2	4	4	3	3	5	5	5	5	4		
## 29	5	5	2	3	3	4	4	5	5	4	5	4		
## 30	5	3	5	5	5	5	5	5	5	5	5	5		

```
fish <- fish[-8, ] # Remove site 8 from data
```

```
# Calculate Jaccard
```

```

fish.dj <- vegdist(fish, method = "jaccard", binary = TRUE)

# Calculate Bray-Curtis
fish.db <- vegdist(fish, method = "bray")

# Calculate Sørensen
fish.ds <- vegdist(fish, method = "bray", binary = TRUE)

fish.dj

```

```

##          1         2         3         4         5         6         7
## 2  0.66666667
## 3  0.75000000 0.25000000
## 4  0.87500000 0.62500000 0.50000000
## 5  0.90909091 0.72727273 0.63636364 0.27272727
## 6  0.90000000 0.70000000 0.60000000 0.20000000 0.09090909
## 7  0.80000000 0.40000000 0.50000000 0.55555556 0.54545455 0.50000000
## 9  1.00000000 0.66666667 0.71428571 0.55555556 0.54545455 0.50000000 0.57142857
## 10 0.83333333 0.50000000 0.57142857 0.44444444 0.45454545 0.40000000 0.16666667
## 11 0.83333333 0.50000000 0.57142857 0.60000000 0.69230769 0.66666667 0.42857143
## 12 0.83333333 0.50000000 0.57142857 0.60000000 0.69230769 0.66666667 0.42857143
## 13 0.83333333 0.50000000 0.57142857 0.72727273 0.78571429 0.76923077 0.62500000
## 14 0.90000000 0.70000000 0.60000000 0.50000000 0.60000000 0.57142857 0.63636364
## 15 0.90909091 0.72727273 0.75000000 0.53846154 0.53333333 0.50000000 0.54545455
## 16 0.94117647 0.82352941 0.76470588 0.52941176 0.44444444 0.41176471 0.70588235
## 17 0.95454545 0.86363636 0.81818182 0.63636364 0.56521739 0.54545455 0.77272727
## 18 0.95652174 0.86956522 0.82608696 0.65217391 0.58333333 0.56521739 0.78260870
## 19 1.00000000 0.91666667 0.87500000 0.70833333 0.58333333 0.62500000 0.83333333
## 20 1.00000000 0.91304348 0.86956522 0.69565217 0.56521739 0.60869565 0.82608696
## 21 1.00000000 0.91666667 0.87500000 0.70833333 0.58333333 0.62500000 0.83333333
## 22 1.00000000 0.95833333 0.91666667 0.75000000 0.62500000 0.66666667 0.87500000
## 23 1.00000000 1.00000000 1.00000000 0.90000000 0.83333333 0.81818182 0.85714286
## 24 1.00000000 1.00000000 1.00000000 0.85714286 0.81250000 0.80000000 0.91666667
## 25 1.00000000 1.00000000 0.90909091 0.76923077 0.53846154 0.61538462 0.81818182
## 26 1.00000000 0.95652174 0.91304348 0.73913043 0.60869565 0.65217391 0.86956522
## 27 1.00000000 0.95833333 0.91666667 0.75000000 0.62500000 0.66666667 0.87500000
## 28 1.00000000 0.95833333 0.91666667 0.75000000 0.62500000 0.66666667 0.87500000
## 29 0.96153846 0.88461538 0.84615385 0.69230769 0.57692308 0.61538462 0.80769231
## 30 1.00000000 1.00000000 0.95833333 0.79166667 0.66666667 0.70833333 0.91666667
##          9        10        11        12        13        14        15
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10 0.62500000
## 11 0.62500000 0.50000000
## 12 0.62500000 0.50000000 0.00000000
## 13 0.77777778 0.66666667 0.28571429 0.28571429
## 14 0.75000000 0.54545455 0.40000000 0.40000000 0.40000000
## 15 0.66666667 0.45454545 0.45454545 0.45454545 0.45454545 0.25000000

```

```

## 16 0.70588235 0.64705882 0.72222222 0.72222222 0.72222222 0.50000000 0.44444444
## 17 0.77272727 0.72727273 0.72727273 0.72727273 0.72727273 0.54545455 0.50000000
## 18 0.78260870 0.73913043 0.73913043 0.73913043 0.73913043 0.56521739 0.52173913
## 19 0.78260870 0.79166667 0.88461538 0.88461538 0.88461538 0.73076923 0.69230769
## 20 0.77272727 0.78260870 0.88000000 0.88000000 0.92307692 0.76923077 0.73076923
## 21 0.78260870 0.79166667 0.88461538 0.88461538 0.92592593 0.77777778 0.74074074
## 22 0.82608696 0.83333333 0.92307692 0.92307692 0.96296296 0.81481481 0.77777778
## 23 0.66666667 0.87500000 0.87500000 0.87500000 1.00000000 0.91666667 0.92307692
## 24 0.81818182 0.83333333 0.92307692 0.92307692 1.00000000 0.87500000 0.88235294
## 25 0.81818182 0.72727273 0.92307692 0.92307692 1.00000000 0.80000000 0.81250000
## 26 0.81818182 0.82608696 0.92000000 0.92000000 0.96153846 0.80769231 0.76923077
## 27 0.82608696 0.83333333 0.92307692 0.92307692 0.96296296 0.81481481 0.77777778
## 28 0.82608696 0.83333333 0.92307692 0.92307692 0.96296296 0.81481481 0.77777778
## 29 0.80769231 0.76923077 0.81481481 0.81481481 0.81481481 0.66666667 0.62962963
## 30 0.86956522 0.87500000 0.96153846 0.96153846 1.00000000 0.85185185 0.81481481
##               16          17          18          19          20          21          22
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17 0.22727273
## 18 0.26086957 0.04347826
## 19 0.40000000 0.26923077 0.23076923
## 20 0.44000000 0.30769231 0.26923077 0.04347826
## 21 0.46153846 0.33333333 0.29629630 0.08333333 0.04347826
## 22 0.50000000 0.37037037 0.33333333 0.12500000 0.08695652 0.04347826
## 23 0.88888889 0.86363636 0.86956522 0.86956522 0.86363636 0.86956522 0.86363636
## 24 0.80952381 0.75000000 0.70833333 0.65217391 0.63636364 0.65217391 0.63636364
## 25 0.75000000 0.75000000 0.70833333 0.65217391 0.63636364 0.65217391 0.63636364
## 26 0.53846154 0.40740741 0.37037037 0.16666667 0.13043478 0.08695652 0.04545455
## 27 0.50000000 0.37037037 0.33333333 0.12500000 0.08695652 0.04347826 0.00000000
## 28 0.50000000 0.37037037 0.33333333 0.12500000 0.08695652 0.04347826 0.00000000
## 29 0.40740741 0.22222222 0.18518519 0.11538462 0.15384615 0.11538462 0.15384615
## 30 0.53846154 0.40740741 0.37037037 0.16666667 0.13043478 0.08695652 0.04545455
##               23          24          25          26          27          28          29
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11

```

```

## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24 0.62500000
## 25 0.62500000 0.54545455
## 26 0.85714286 0.61904762 0.61904762
## 27 0.86363636 0.63636364 0.63636364 0.04545455
## 28 0.86363636 0.63636364 0.63636364 0.04545455 0.00000000
## 29 0.88461538 0.69230769 0.69230769 0.19230769 0.15384615 0.15384615
## 30 0.85714286 0.61904762 0.61904762 0.09090909 0.04545455 0.04545455 0.19230769

```

fish.db

	1	2	3	4	5	6	7
## 2	0.60000000						
## 3	0.68421053	0.14285714					
## 4	0.75000000	0.33333333	0.18918919				
## 5	0.89189189	0.69565217	0.68000000	0.49090909			
## 6	0.75000000	0.39393939	0.29729730	0.19047619	0.41818182		
## 7	0.68421053	0.14285714	0.12500000	0.24324324	0.64000000	0.24324324	
## 9	1.00000000	0.69230769	0.73333333	0.65714286	0.58333333	0.54285714	0.66666667
## 10	0.88235294	0.38461538	0.40000000	0.37142857	0.54166667	0.25714286	0.26666667
## 11	0.57142857	0.30434783	0.40740741	0.43750000	0.68888889	0.43750000	0.33333333
## 12	0.71428571	0.20000000	0.23529412	0.33333333	0.69230769	0.38461538	0.17647059
## 13	0.72727273	0.29032258	0.31428571	0.45000000	0.73584906	0.55000000	0.37142857
## 14	0.80645161	0.40000000	0.31818182	0.34693878	0.67741935	0.42857143	0.36363636
## 15	0.83333333	0.51111111	0.46938776	0.40740741	0.55223881	0.37037037	0.38775510
## 16	0.86046512	0.65384615	0.57142857	0.47540984	0.45945946	0.37704918	0.53571429
## 17	0.91489362	0.67857143	0.63333333	0.50769231	0.51282051	0.44615385	0.60000000
## 18	0.95555556	0.74074074	0.72413793	0.58730159	0.50000000	0.52380952	0.68965517
## 19	1.00000000	0.79310345	0.70967742	0.61194030	0.50000000	0.52238806	0.67741935
## 20	1.00000000	0.91176471	0.88888889	0.74025974	0.48888889	0.68831169	0.86111111
## 21	1.00000000	0.94594595	0.92307692	0.78313253	0.50000000	0.73493976	0.89743590
## 22	1.00000000	0.97619048	0.95454545	0.82795699	0.52830189	0.78494624	0.93181818
## 23	1.00000000	1.00000000	1.00000000	0.92000000	0.89473684	0.84000000	0.90000000
## 24	1.00000000	1.00000000	1.00000000	0.88888889	0.79591837	0.77777778	0.93548387
## 25	1.00000000	1.00000000	0.92592593	0.81250000	0.68888889	0.68750000	0.85185185
## 26	1.00000000	0.96363636	0.93220339	0.78125000	0.55844156	0.68750000	0.89830508
## 27	1.00000000	0.97333333	0.94936709	0.83333333	0.56701031	0.76190476	0.92405063
## 28	1.00000000	0.97560976	0.95348837	0.82417582	0.57692308	0.78021978	0.93023256
## 29	0.97777778	0.93939394	0.92233010	0.81481481	0.53719008	0.77777778	0.90291262
## 30	1.00000000	1.00000000	0.98095238	0.87272727	0.59349593	0.83636364	0.96190476
## 2	9	10	11	12	13	14	15
## 3							

```

## 4
## 5
## 6
## 7
## 9
## 10 0.57142857
## 11 0.76000000 0.44000000
## 12 0.68750000 0.37500000 0.24137931
## 13 0.81818182 0.57575758 0.33333333 0.18918919
## 14 0.76190476 0.47619048 0.43589744 0.21739130 0.19148936
## 15 0.65957447 0.40425532 0.50000000 0.33333333 0.38461538 0.24590164
## 16 0.70370370 0.51851852 0.64705882 0.55172414 0.59322034 0.44117647 0.26027397
## 17 0.68965517 0.51724138 0.63636364 0.58064516 0.61904762 0.50000000 0.40259740
## 18 0.64285714 0.57142857 0.69811321 0.66666667 0.70491803 0.60000000 0.46666667
## 19 0.66666667 0.63333333 0.82456140 0.75000000 0.81538462 0.67567568 0.56962025
## 20 0.68571429 0.77142857 0.91044776 0.89189189 0.92000000 0.83333333 0.70786517
## 21 0.76315789 0.81578947 0.91780822 0.92500000 0.95061728 0.86666667 0.76842105
## 22 0.76744186 0.86046512 0.95180723 0.95555556 0.97802198 0.90000000 0.77142857
## 23 0.77777778 0.88888889 0.86666667 0.90909091 1.00000000 0.93750000 0.94594595
## 24 0.72413793 0.79310345 0.92307692 0.93939394 1.00000000 0.90697674 0.87500000
## 25 0.84000000 0.76000000 0.90909091 0.93103448 1.00000000 0.84615385 0.81818182
## 26 0.71929825 0.82456140 0.92592593 0.93442623 0.96774194 0.85915493 0.76315789
## 27 0.76623377 0.84415584 0.94594595 0.95061728 0.97560976 0.89010989 0.77083333
## 28 0.76190476 0.85714286 0.95061728 0.95454545 0.97752809 0.89795918 0.78640777
## 29 0.78217822 0.84158416 0.89795918 0.90476190 0.90566038 0.84347826 0.73333333
## 30 0.84466019 0.90291262 0.98000000 0.98130841 1.00000000 0.93162393 0.81967213
## 16      17      18      19      20      21      22
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17 0.26190476
## 18 0.34146341 0.13953488
## 19 0.39534884 0.31111111 0.25000000
## 20 0.58333333 0.42000000 0.32653061 0.23529412
## 21 0.62745098 0.49056604 0.40384615 0.29629630 0.10169492
## 22 0.66071429 0.55172414 0.47368421 0.38983051 0.18750000 0.10447761
## 23 0.90909091 0.83333333 0.82608696 0.84000000 0.86666667 0.87878788 0.89473684
## 24 0.81818182 0.69491525 0.64912281 0.63934426 0.57746479 0.61038961 0.65517241
## 25 0.76470588 0.74545455 0.66037736 0.61403509 0.67164179 0.69863014 0.73493976
## 26 0.63855422 0.54022989 0.45882353 0.32584270 0.21212121 0.20000000 0.25217391
## 27 0.66990291 0.57009346 0.48571429 0.37614679 0.19327731 0.13600000 0.12592593
## 28 0.69090909 0.57894737 0.50000000 0.41379310 0.22222222 0.16666667 0.12676056
## 29 0.65354331 0.51145038 0.44186047 0.41353383 0.24475524 0.18120805 0.11949686

```

```

## 30 0.72093023 0.57894737 0.52671756 0.48148148 0.29655172 0.23178808 0.18012422
##          23        24        25        26        27        28        29
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24 0.57894737
## 25 0.46666667 0.46153846
## 26 0.82978723 0.48275862 0.59259259
## 27 0.88059701 0.61538462 0.70270270 0.18867925
## 28 0.89189189 0.64705882 0.72839506 0.23893805 0.09774436
## 29 0.91208791 0.70588235 0.77551020 0.33846154 0.18666667 0.14649682
## 30 0.91397849 0.71153846 0.78000000 0.36363636 0.19736842 0.15723270 0.14772727

```

fish.ds

```

##          1        2        3        4        5        6        7
## 2 0.5000000
## 3 0.6000000 0.14285714
## 4 0.77777778 0.45454545 0.33333333
## 5 0.83333333 0.57142857 0.46666667 0.15789474
## 6 0.81818182 0.53846154 0.42857143 0.11111111 0.04761905
## 7 0.66666667 0.25000000 0.33333333 0.38461538 0.37500000 0.33333333
## 9 1.00000000 0.50000000 0.55555556 0.38461538 0.37500000 0.33333333 0.40000000
## 10 0.71428571 0.33333333 0.40000000 0.28571429 0.29411765 0.25000000 0.09090909
## 11 0.71428571 0.33333333 0.40000000 0.42857143 0.52941176 0.50000000 0.27272727
## 12 0.71428571 0.33333333 0.40000000 0.42857143 0.52941176 0.50000000 0.27272727
## 13 0.71428571 0.33333333 0.40000000 0.57142857 0.64705882 0.62500000 0.45454545
## 14 0.81818182 0.53846154 0.42857143 0.33333333 0.42857143 0.40000000 0.46666667
## 15 0.83333333 0.57142857 0.60000000 0.36842105 0.36363636 0.33333333 0.37500000
## 16 0.88888889 0.70000000 0.61904762 0.36000000 0.28571429 0.25925926 0.54545455
## 17 0.91304348 0.76000000 0.69230769 0.46666667 0.39393939 0.37500000 0.62962963
## 18 0.91666667 0.76923077 0.70370370 0.48387097 0.41176471 0.39393939 0.64285714
## 19 1.00000000 0.84615385 0.77777778 0.54838710 0.41176471 0.45454545 0.71428571
## 20 1.00000000 0.84000000 0.76923077 0.53333333 0.39393939 0.43750000 0.70370370
## 21 1.00000000 0.84615385 0.77777778 0.54838710 0.41176471 0.45454545 0.71428571
## 22 1.00000000 0.92000000 0.84615385 0.60000000 0.45454545 0.50000000 0.77777778

```

```

## 23 1.0000000 1.0000000 1.0000000 0.81818182 0.71428571 0.69230769 0.75000000
## 24 1.0000000 1.0000000 1.0000000 0.75000000 0.68421053 0.66666667 0.84615385
## 25 1.0000000 1.0000000 0.83333333 0.62500000 0.36842105 0.44444444 0.69230769
## 26 1.0000000 0.91666667 0.84000000 0.58620690 0.43750000 0.48387097 0.76923077
## 27 1.0000000 0.92000000 0.84615385 0.60000000 0.45454545 0.50000000 0.77777778
## 28 1.0000000 0.92000000 0.84615385 0.60000000 0.45454545 0.50000000 0.77777778
## 29 0.92592593 0.79310345 0.73333333 0.52941176 0.40540541 0.44444444 0.67741935
## 30 1.0000000 1.0000000 0.92000000 0.65517241 0.50000000 0.54838710 0.84615385
##               9          10          11          12          13          14          15
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10 0.45454545
## 11 0.45454545 0.33333333
## 12 0.45454545 0.33333333 0.00000000
## 13 0.63636364 0.50000000 0.16666667 0.16666667
## 14 0.60000000 0.37500000 0.25000000 0.25000000 0.25000000
## 15 0.50000000 0.29411765 0.29411765 0.29411765 0.29411765 0.14285714
## 16 0.54545455 0.47826087 0.56521739 0.56521739 0.56521739 0.33333333 0.28571429
## 17 0.62962963 0.57142857 0.57142857 0.57142857 0.57142857 0.37500000 0.33333333
## 18 0.64285714 0.58620690 0.58620690 0.58620690 0.58620690 0.39393939 0.35294118
## 19 0.64285714 0.65517241 0.79310345 0.79310345 0.79310345 0.57575758 0.52941176
## 20 0.62962963 0.64285714 0.78571429 0.78571429 0.85714286 0.62500000 0.57575758
## 21 0.64285714 0.65517241 0.79310345 0.79310345 0.86206897 0.63636364 0.58823529
## 22 0.70370370 0.71428571 0.85714286 0.85714286 0.92857143 0.68750000 0.63636364
## 23 0.50000000 0.77777778 0.77777778 0.77777778 1.00000000 0.84615385 0.85714286
## 24 0.69230769 0.71428571 0.85714286 0.85714286 1.00000000 0.77777778 0.78947368
## 25 0.69230769 0.57142857 0.85714286 0.85714286 1.00000000 0.66666667 0.68421053
## 26 0.69230769 0.70370370 0.85185185 0.85185185 0.92592593 0.67741935 0.62500000
## 27 0.70370370 0.71428571 0.85714286 0.85714286 0.92857143 0.68750000 0.63636364
## 28 0.70370370 0.71428571 0.85714286 0.85714286 0.92857143 0.68750000 0.63636364
## 29 0.67741935 0.62500000 0.68750000 0.68750000 0.68750000 0.50000000 0.45945946
## 30 0.76923077 0.77777778 0.92592593 0.92592593 1.00000000 0.74193548 0.68750000
##               16          17          18          19          20          21          22
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17 0.12820513
## 18 0.15000000 0.02222222

```

```

## 19 0.25000000 0.15555556 0.13043478
## 20 0.28205128 0.18181818 0.15555556 0.02222222
## 21 0.30000000 0.20000000 0.17391304 0.04347826 0.02222222
## 22 0.33333333 0.22727273 0.20000000 0.06666667 0.04545455 0.02222222
## 23 0.80000000 0.76000000 0.76923077 0.76923077 0.76000000 0.76923077 0.76000000
## 24 0.68000000 0.60000000 0.54838710 0.48387097 0.46666667 0.48387097 0.46666667
## 25 0.60000000 0.60000000 0.54838710 0.48387097 0.46666667 0.48387097 0.46666667
## 26 0.36842105 0.25581395 0.22727273 0.09090909 0.06976744 0.04545455 0.02325581
## 27 0.33333333 0.22727273 0.20000000 0.06666667 0.04545455 0.02222222 0.00000000
## 28 0.33333333 0.22727273 0.20000000 0.06666667 0.04545455 0.02222222 0.00000000
## 29 0.25581395 0.12500000 0.10204082 0.06122449 0.08333333 0.06122449 0.08333333
## 30 0.36842105 0.25581395 0.22727273 0.09090909 0.06976744 0.04545455 0.02325581
##          23      24      25      26      27      28      29
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24 0.45454545
## 25 0.45454545 0.37500000
## 26 0.75000000 0.44827586 0.44827586
## 27 0.76000000 0.46666667 0.46666667 0.02325581
## 28 0.76000000 0.46666667 0.46666667 0.02325581 0.00000000
## 29 0.79310345 0.52941176 0.52941176 0.10638298 0.08333333 0.08333333
## 30 0.75000000 0.44827586 0.44827586 0.04761905 0.02325581 0.02325581 0.10638298

```

Question 5: Using the distance matrices from above, answer the following questions:

- Does the resemblance matrix (`fish.db`) represent similarity or dissimilarity? What information in the resemblance matrix led you to arrive at your answer?
- Compare the resemblance matrices (`fish.db` or `fish.ds`) you just created. How does the choice of the Sørensen or Bray-Curtis distance influence your interpretation of site (dis)similarity?

Answer 5a: I think it represents dissimilarity. I tried looking at sites within `fish` that looked very different and very similar, and ones that looked different tended to have higher values in `db`. **Answer 5b:** I think that the Bray-Curtis distance makes sites look more similar, while the Sørensen distance makes sites look more dissimilar.

4) VISUALIZING BETA-DIVERSITY

A. Heatmaps

In the R code chunk below, do the following:

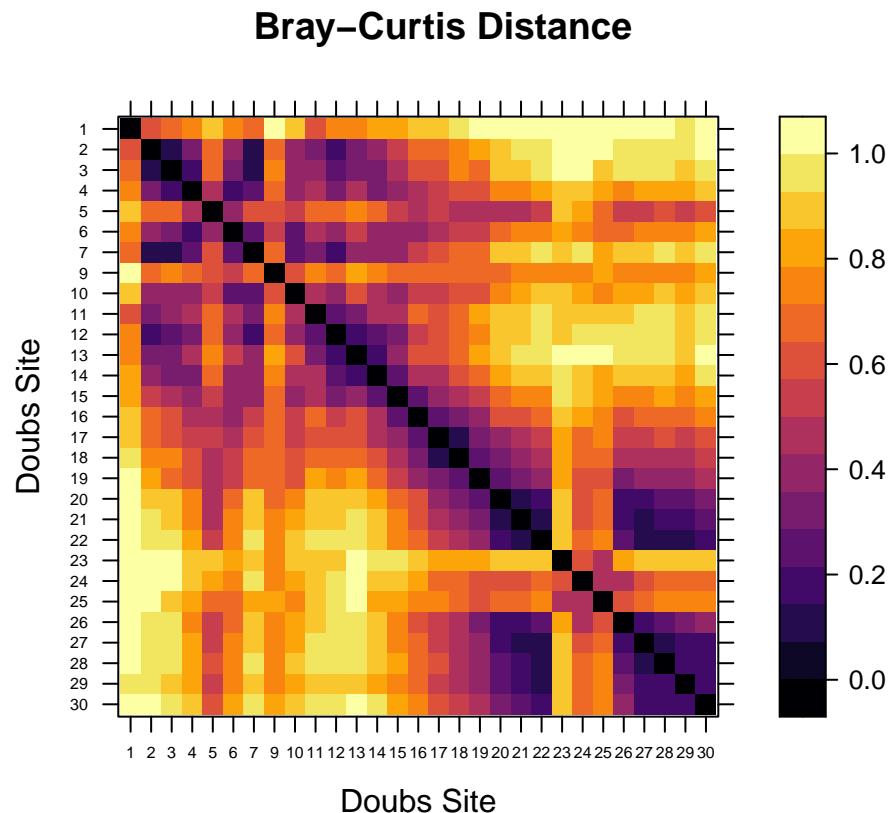
1. define a color palette,
2. define the order of sites in the Doubs River, and
3. use the `levelplot()` function to create a heatmap of fish abundances in the Doubs River.

```
# Define Order of Sites
order <- rev(attr(fish.db, "Labels"))

#install.packages("lattice")
library(lattice)

## Warning: package 'lattice' was built under R version 4.5.2

# Plot Heatmap
levelplot(as.matrix(fish.db)[, order], aspect = "iso", col.regions = inferno,
         xlab = "Doubs Site", ylab = "Doubs Site", scales = list(cex = 0.5),
         main = "Bray-Curtis Distance")
```



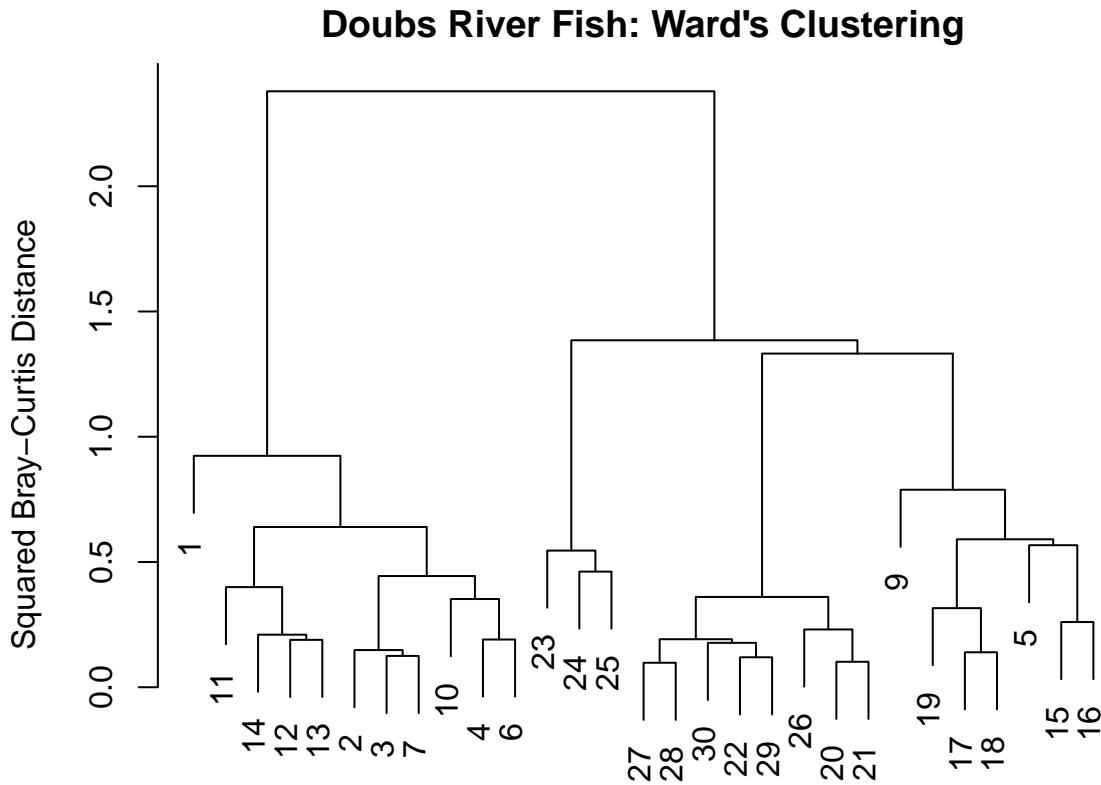
B. Cluster Analysis

In the R code chunk below, do the following:

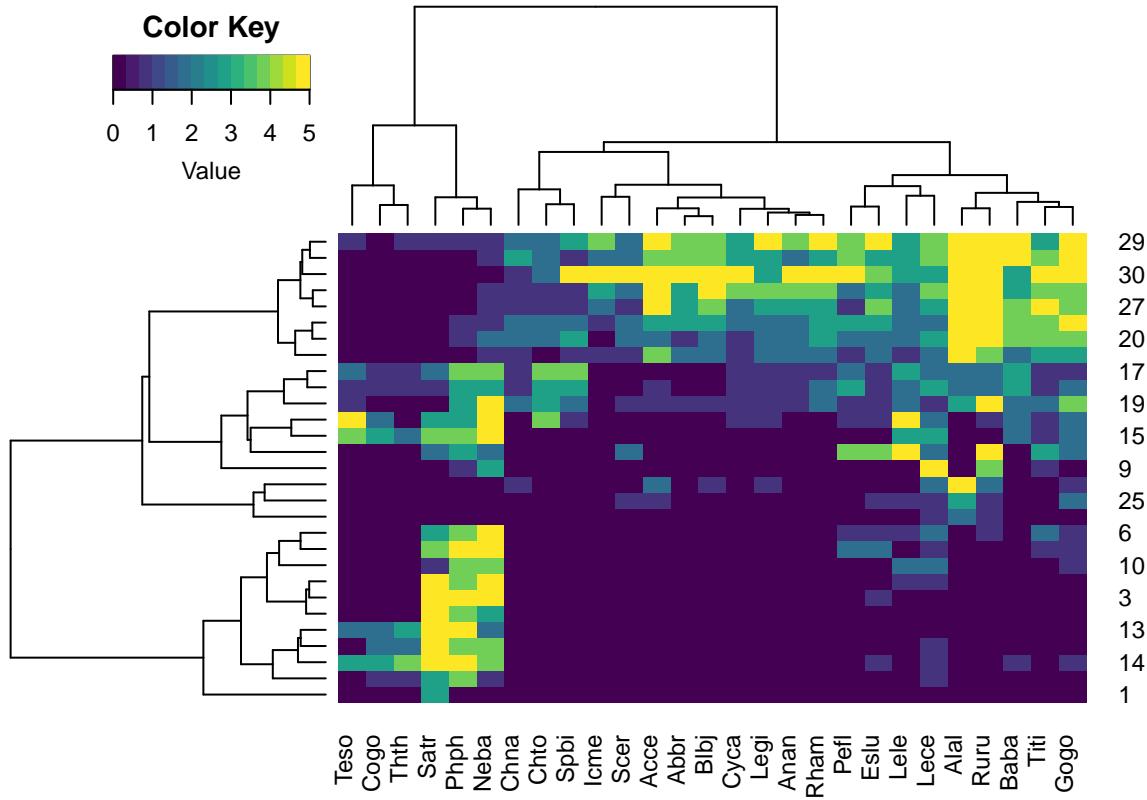
1. perform a cluster analysis using Ward's Clustering, and
2. plot your cluster analysis (use either `hclust` or `heatmap.2`).

```
# Perform Cluster Analysis
fish.ward <- hclust(fish.db, method = "ward.D2")

# Plot Cluster
par(mar = c(1, 5, 2, 2) + 0.1)
plot(fish.ward, main = "Doubs River Fish: Ward's Clustering",
     ylab = "Squared Bray-Curtis Distance")
```



```
gplots:::heatmap.2(as.matrix(fish),
  distfun = function(x) vegdist(x, method = "bray"),
  hclustfun = function(x) hclust(x, method = "ward.D2"),
  col = viridis, trace = "none", density.info = "none")
```



Question 6: Based on cluster analyses and the introductory plots that we generated after loading the data, develop an ecological hypothesis for fish diversity the `doubs` data set?

Answer 6: Due to the unique shape of the Doubs river, the bend and downstream are hotspots of diversity as nutrients get caught there.

C. Ordination

Principal Coordinates Analysis (PCoA)

In the R code chunk below, do the following:

1. perform a Principal Coordinates Analysis to visualize beta-diversity
2. calculate the variation explained by the first three axes in your ordination
3. plot the PCoA ordination,
4. label the sites as points using the Doubs River site number, and
5. identify influential species and add species coordinates to PCoA plot.

```
fish.pcoa <- cmdscale(fish.db, eig = TRUE, k = 3)

explainvar1 <- round(fish.pcoa$eig[1] / sum(fish.pcoa$eig), 3) * 100
explainvar2 <- round(fish.pcoa$eig[2] / sum(fish.pcoa$eig), 3) * 100
explainvar3 <- round(fish.pcoa$eig[3] / sum(fish.pcoa$eig), 3) * 100
sum.eig <- sum(explainvar1, explainvar2, explainvar3)
```

```

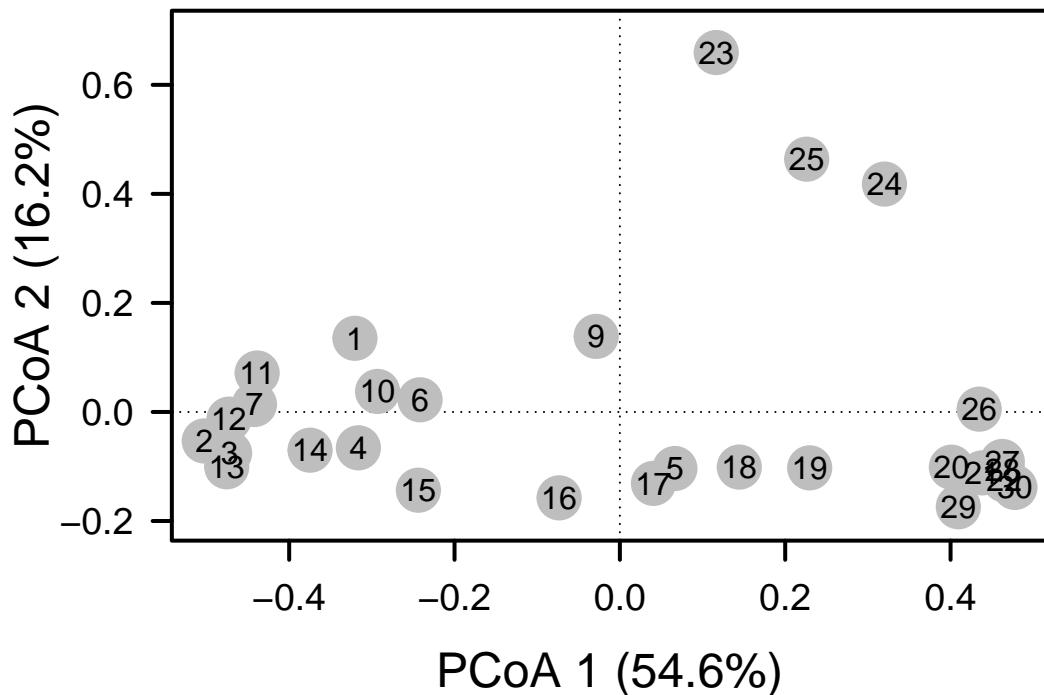
# Define Plot Parameters
par(mar = c(5, 5, 1, 2) + 0.1)

# Initiate Plot
plot(fish.pcoa$points[,1], fish.pcoa$points[,2], ylim = c(-0.2, 0.7),
      xlab = paste("PCoA 1 (", explainvar1, "%)", sep = ""),
      ylab = paste("PCoA 2 (", explainvar2, "%)", sep = ""),
      pch = 16, cex = 2.0, type = "n", cex.lab = 1.5,
      cex.axis = 1.2, axes = FALSE)

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

# Add Points & Labels
points(fish.pcoa$points[,1], fish.pcoa$points[,2],
       pch = 19, cex = 3, bg = "gray", col = "gray")
text(fish.pcoa$points[,1], fish.pcoa$points[,2],
     labels = row.names(fish.pcoa$points))

```

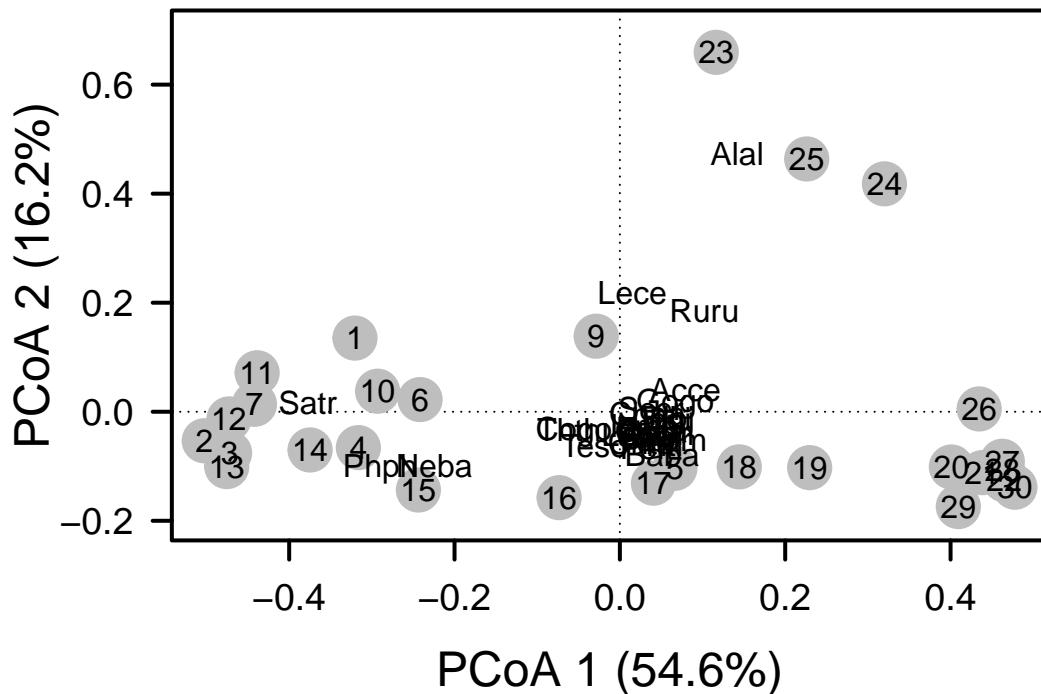


```

# First we calculate the relative abundances of each species at each site
fishREL <- fish
for(i in 1:nrow(fish)){
  fishREL[i, ] = fish[i, ] / sum(fish[i, ])
}

```

```
# Now, we use this information to calculate and add species scores
fish.pcoa <- add.spec.scores(fish.pcoa,fishREL,method = "pcoa.scores")
text(fish.pcoa$cproj[,1], fish.pcoa$cproj[,2],
     labels = row.names(fish.pcoa$cproj), col = "black")
```



In the R code chunk below, do the following:

1. identify influential species based on correlations along each PCoA axis (use a cutoff of 0.70), and
2. use a permutation test (999 permutations) to test the correlations of each species along each axis.

```
spe.corr <- add.spec.scores(fish.pcoa, fishREL, method = "cor.scores")$cproj
corrcut <- 0.7      # user defined cutoff
imp.spp <- spe.corr[abs(spe.corr[, 1]) >= corrcut | abs(spe.corr[, 2]) >= corrcut, ]

# Permutation Test for Species Abundances Across Axes
fit <- envfit(fish.pcoa, fishREL, perm = 999)
```

Question 7: Address the following questions about the ordination results of the `doubs` data set:

- a. Describe the grouping of sites in the Doubs River based on fish community composition.
- b. Generate a hypothesis about which fish species are potential indicators of river quality.

Answer 7a: There are 3 obvious groups of sites. The downstream portion of the river (sites 20-26), the bend of the river (sites 16-19), and the upstream portion of the river (sites 1-8). These sites have similar species richness and abundance values.

Answer 7b: Two very important fish species appear to be PhPh and Blbj. These species may be more susceptible to subtle nutrient differences along the river, so an increase or decrease in their abundances may be useful in assessing overall river health.