

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

Student Name; Z620: Quantitative Biodiversity, Indiana University

06 February, 2025

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] "/cloud/project/QB2025_Spencer/Week3-Beta"

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)
  }
}

## This is vegan 2.6-8
##
## Attaching package: 'gplots'
## The following object is masked from 'package:stats':
##
##     lowess
```

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, max.level = 1)
```

```
## List of 4
## $ env      : 'data.frame': 30 obs. of  11 variables:
## $ fish      : 'data.frame': 30 obs. of  27 variables:
## $ xy        : 'data.frame': 30 obs. of  2 variables:
## $ species: 'data.frame': 27 obs. of  4 variables:
```

```
head(doubs$env)
```

```
##   dfs alt   slo flo pH har pho nit amm oxy bdo
## 1    3 934 6.176 84 79 45   1 20   0 122 27
## 2   22 932 3.434 100 80 40   2 20  10 103 19
## 3   102 914 3.638 180 83 52   5 22   5 105 35
## 4   185 854 3.497 253 80 72  10 21   0 110 13
## 5   215 849 3.178 264 81 84  38 52  20  80 62
## 6   324 846 3.497 286 79 60  20 15   0 102 53
```

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

- a. How many objects are in `doubs`?
- b. How many fish species are there in the `doubs` dataset?
- c. How many sites are in the `doubs` dataset?

Answer 1a: There are 4 different objects in `doubs`. **Answer 1b:** There are 25 fish species.

Answer 1c: There are 30 sites.

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

Answer 2a: Fish richness seems to be highest towards the middle of the reach (highest X and Y coordinates) and at the farthest downstream part of the reach. Richness is significantly lower upstream to about 90-250X, 0-175Y. **Answer 2b:** Richness is highest upstream (90-250X, 0-175Y), with a drop in the upstream portion. After the upstream portion, the richness drops significantly past the bend.

Answer 2c: 1) In the case of fish richness, there is no indication of how different species abundance changes across the reach and 2) Richness alone does not provide a good quantitative method of determining how diversity and species abundance vary across sites.

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")
  S <- ncol(SbyS.pa[,which(colSums(SbyS.pa) > 0)])
  a.bar <- mean(specnumber(SbyS.pa))
  b.w <- round(S/a.bar, 3)
  return(b.w)
}

beta.w <- function(site.by.species = "", sitenum1 = "", sitenum2 = "", pairwise = FALSE){
  if (pairwise == TRUE){
    if (sitenum1 == "" | sitenum2 == ""){
      print("Error:please specify sites to compare")
      return(NA) }
    site1 = site.by.species[sitenum1,]
    site2 = site.by.species[sitenum2,]
    site1 = subset(site1,select = site1 > 0)
    site2 = subset(site2, select = site2 > 0)
    gamma = union(colnames(site1), colnames(site2))
    s = length(gamma)
    a.bar = mean(c(specnumber(site1), specnumber(site2)))
    b.w = round(s/a.bar - 1, 3)
    return(b.w)
  }
  else{
    SbyS.pa <- decostand(site.by.species, method = "pa")
    S <- ncol(SbyS.pa[,which(colSums(SbyS.pa) > 0)])
    a.bar <- mean(specnumber(SbyS.pa))
    b.w <- round(S/a.bar, 3)
    return(b.w)
  }
}
```

```

}
}

#Alpha-diversity
specnumber(doubs$fish)

## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
## 1 3 4 8 11 10 5 0 5 6 6 6 6 10 11 17 22 23 23 22 23 22 3 8 8 21
## 27 28 29 30
## 22 22 26 21

#Beta-diversity
beta.w(doubs$fish)

## [1] 2.16

#Question-3b
beta.w(doubs$fish, sitenum1 = 1, sitenum2 = 2, pairwise = TRUE)

## [1] 0.5

beta.w(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: Using Whittaker's Beta-diversity equation, $\text{Beta.w} = \gamma/\alpha$, we can see that regional diversity is multiplicative of local diversity and turnover. Using the `specnumber()` function and the previous richness plots, we can see that the total local richness varies drastically along the reach of the river, starting from 0 and going to 26. Our beta-diversity value is 2.16, meaning that our gamma diversity is about 2.16 times higher than our alpha-diversity. **Answer 3b:** Comparing the beta-diversity values between pairwise comparisons of site1 vs site2 and site1 vs site10, we can see that site 1 is more similar to site 2 because the pairwise comparison has a lower beta-diversity measure. **Answer 3c:** If beta-diversity were defined additively, that would mean that gamma is simply a measure of how many more species there are when comparing across multiple sample sites than there is at a target site (alpha). Rather than observing proportional changes in diversity, we would have an absolute value for number of species.

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: Because incidence-based metrics simply catalogue either the presence or absence, they weight abundance and rare taxa exactly the same. The only way incidence-based metrics vary in their calculations is in the weight placed on taxa shared between sites. Abundance-based metrics, however, take into account the abundance of each taxa and are thus more influenced by common or abundant taxa.

In the R code chunk below, do the following:

- make a new object, `fish`, containing the fish abundance data for the Doubs River,

2. remove any sites where no fish were observed (i.e., rows with sum of zero),
3. construct a resemblance matrix based on Sørensen's Similarity ("fish.ds"), and
4. construct a resemblance matrix based on Bray-Curtis Distance ("fish.db").

```
fish <- doubs$fish
fish <- fish[-8, ]
```

```
fish.ds <- vegdist(fish, method = "bray", diag = TRUE, binary = TRUE)
fish.db <- vegdist(fish, method = "bray", diag = TRUE)
```

```
print(fish.ds)
```

```
##           1           2           3           4           5           6           7
## 1  0.00000000
## 2  0.50000000 0.00000000
## 3  0.60000000 0.14285714 0.00000000
## 4  0.77777778 0.45454545 0.33333333 0.00000000
## 5  0.83333333 0.57142857 0.46666667 0.15789474 0.00000000
## 6  0.81818182 0.53846154 0.42857143 0.11111111 0.04761905 0.00000000
## 7  0.66666667 0.25000000 0.33333333 0.38461538 0.37500000 0.33333333 0.00000000
## 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 1.00000000 1.00000000 1.00000000 0.81818182 0.71428571 0.69230769 0.75000000
## 24 1.00000000 1.00000000 1.00000000 0.75000000 0.68421053 0.66666667 0.84615385
## 25 1.00000000 1.00000000 0.83333333 0.62500000 0.36842105 0.44444444 0.69230769
## 26 1.00000000 0.91666667 0.84000000 0.58620690 0.43750000 0.48387097 0.76923077
## 27 1.00000000 0.92000000 0.84615385 0.60000000 0.45454545 0.50000000 0.77777778
## 28 1.00000000 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 1.00000000 1.00000000 0.92000000 0.65517241 0.50000000 0.54838710 0.84615385
##           9          10          11          12          13          14          15
## 1
## 2
## 3
## 4
## 5
## 6
## 7
## 9  0.00000000
## 10 0.45454545 0.00000000
## 11 0.45454545 0.33333333 0.00000000
## 12 0.45454545 0.33333333 0.00000000 0.00000000
## 13 0.63636364 0.50000000 0.16666667 0.16666667 0.00000000
```

```

## 14 0.60000000 0.37500000 0.25000000 0.25000000 0.25000000 0.00000000
## 15 0.50000000 0.29411765 0.29411765 0.29411765 0.29411765 0.14285714 0.00000000
## 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
## 1
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16 0.00000000
## 17 0.12820513 0.00000000
## 18 0.15000000 0.02222222 0.00000000
## 19 0.25000000 0.15555556 0.13043478 0.00000000
## 20 0.28205128 0.18181818 0.15555556 0.02222222 0.00000000
## 21 0.30000000 0.20000000 0.17391304 0.04347826 0.02222222 0.00000000
## 22 0.33333333 0.22727273 0.20000000 0.06666667 0.04545455 0.02222222 0.00000000
## 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
## 1
## 2
## 3
## 4
## 5
## 6

```

```

## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23 0.00000000
## 24 0.45454545 0.00000000
## 25 0.45454545 0.37500000 0.00000000
## 26 0.75000000 0.44827586 0.44827586 0.00000000
## 27 0.76000000 0.46666667 0.46666667 0.02325581 0.00000000
## 28 0.76000000 0.46666667 0.46666667 0.02325581 0.00000000 0.00000000
## 29 0.79310345 0.52941176 0.52941176 0.10638298 0.08333333 0.08333333 0.00000000
## 30 0.75000000 0.44827586 0.44827586 0.04761905 0.02325581 0.02325581 0.10638298
##      30
## 1
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24
## 25
## 26
## 27
## 28
## 29
## 30 0.00000000

```

```
print(fish.db)
```

##	1	2	3	4	5	6	7
## 1	0.00000000						
## 2	0.60000000	0.00000000					
## 3	0.68421053	0.14285714	0.00000000				
## 4	0.75000000	0.33333333	0.18918919	0.00000000			
## 5	0.89189189	0.69565217	0.68000000	0.49090909	0.00000000		
## 6	0.75000000	0.39393939	0.29729730	0.19047619	0.41818182	0.00000000	
## 7	0.68421053	0.14285714	0.12500000	0.24324324	0.64000000	0.24324324	0.00000000
## 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
##	9	10	11	12	13	14	15
## 1							
## 2							
## 3							
## 4							
## 5							
## 6							
## 7							
## 9	0.00000000						
## 10	0.57142857	0.00000000					
## 11	0.76000000	0.44000000	0.00000000				
## 12	0.68750000	0.37500000	0.24137931	0.00000000			
## 13	0.81818182	0.57575758	0.33333333	0.18918919	0.00000000		
## 14	0.76190476	0.47619048	0.43589744	0.21739130	0.19148936	0.00000000	
## 15	0.65957447	0.40425532	0.50000000	0.33333333	0.38461538	0.24590164	0.00000000
## 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
## 1
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16 0.00000000
## 17 0.26190476 0.00000000
## 18 0.34146341 0.13953488 0.00000000
## 19 0.39534884 0.31111111 0.25000000 0.00000000
## 20 0.58333333 0.42000000 0.32653061 0.23529412 0.00000000
## 21 0.62745098 0.49056604 0.40384615 0.29629630 0.10169492 0.00000000
## 22 0.66071429 0.55172414 0.47368421 0.38983051 0.18750000 0.10447761 0.00000000
## 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
## 1
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16

```

```

## 17
## 18
## 19
## 20
## 21
## 22
## 23 0.00000000
## 24 0.57894737 0.00000000
## 25 0.46666667 0.46153846 0.00000000
## 26 0.82978723 0.48275862 0.59259259 0.00000000
## 27 0.88059701 0.61538462 0.70270270 0.18867925 0.00000000
## 28 0.89189189 0.64705882 0.72839506 0.23893805 0.09774436 0.00000000
## 29 0.91208791 0.70588235 0.77551020 0.33846154 0.18666667 0.14649682 0.00000000
## 30 0.91397849 0.71153846 0.78000000 0.36363636 0.19736842 0.15723270 0.14772727
##          30
## 1
## 2
## 3
## 4
## 5
## 6
## 7
## 9
## 10
## 11
## 12
## 13
## 14
## 15
## 16
## 17
## 18
## 19
## 20
## 21
## 22
## 23
## 24
## 25
## 26
## 27
## 28
## 29
## 30 0.00000000

```

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: The resemblance matrix represents similarity - I got to this conclusion because the diagonal values (comparisons between the same sites) are zero. **Answer 5b:** Qualitatively, it seems like the resemblance matrix generated from the Bray-Curtis distance values are slightly

lower than that of the Sorenson metric. This is most likely because the BC distance places more weight on abundant taxa and may mask the effect of rare taxa on beta-diversity, at least more so than the incidence-based metric Sorenson.

4) VISUALIZING BETA-DIVERSITY

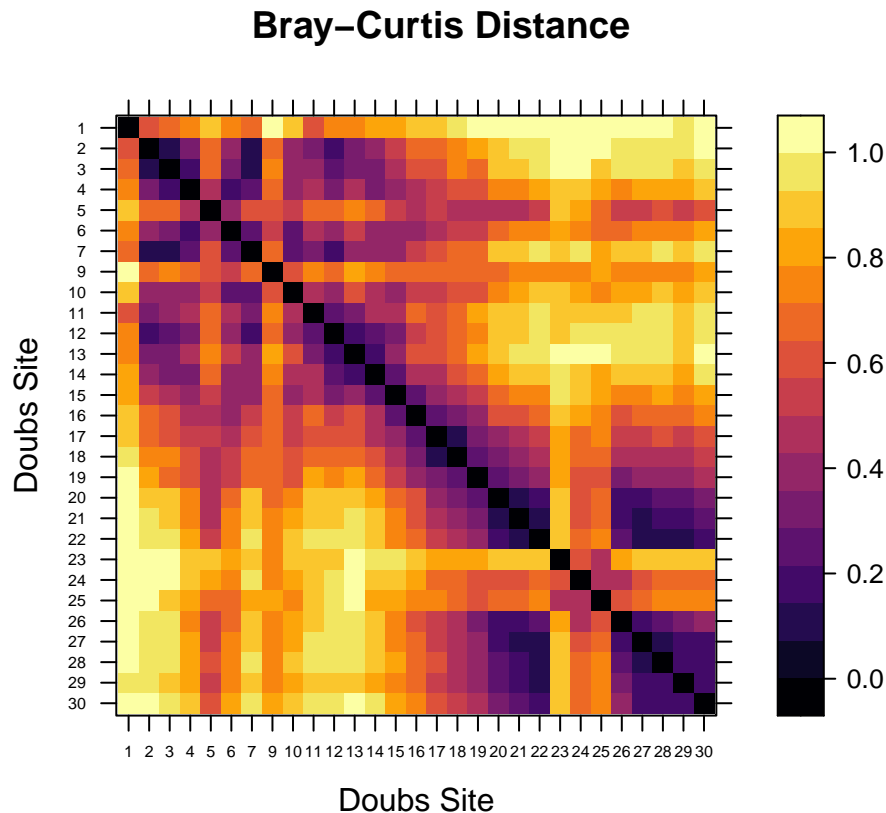
A. Heatmaps

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.

```
order <- rev(attr(fish.db, "Labels"))

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

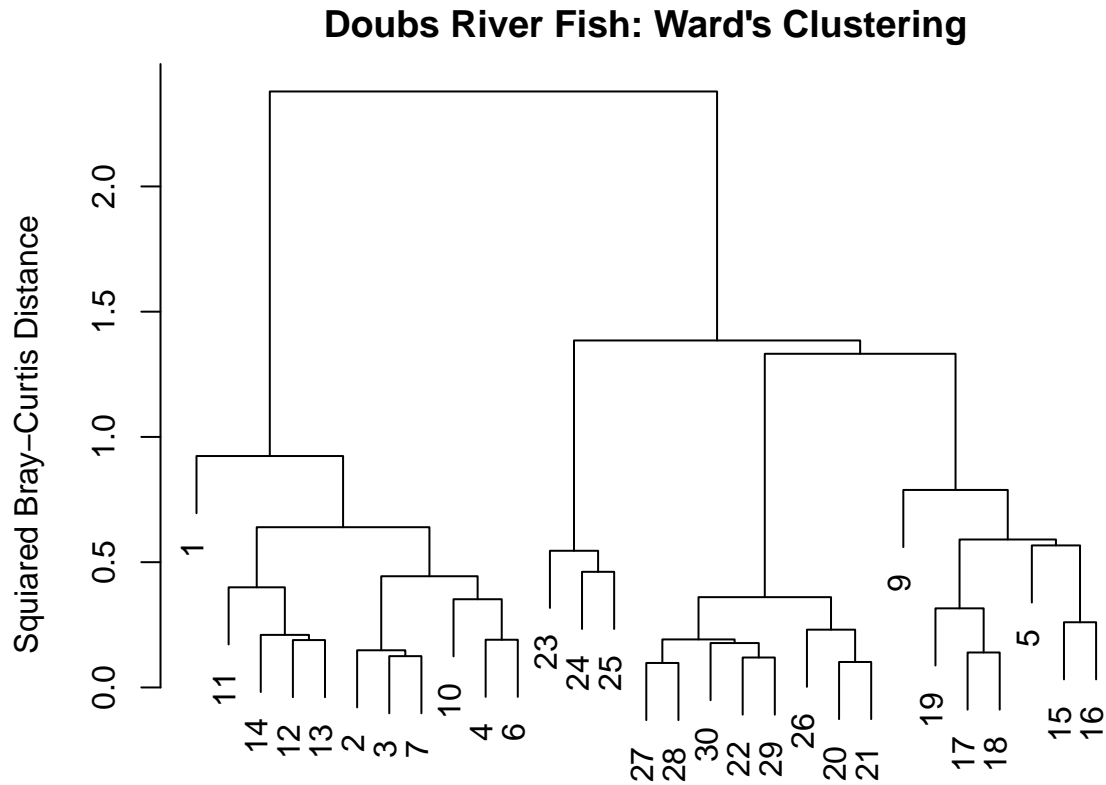
In the R code chunk below, do the following:

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

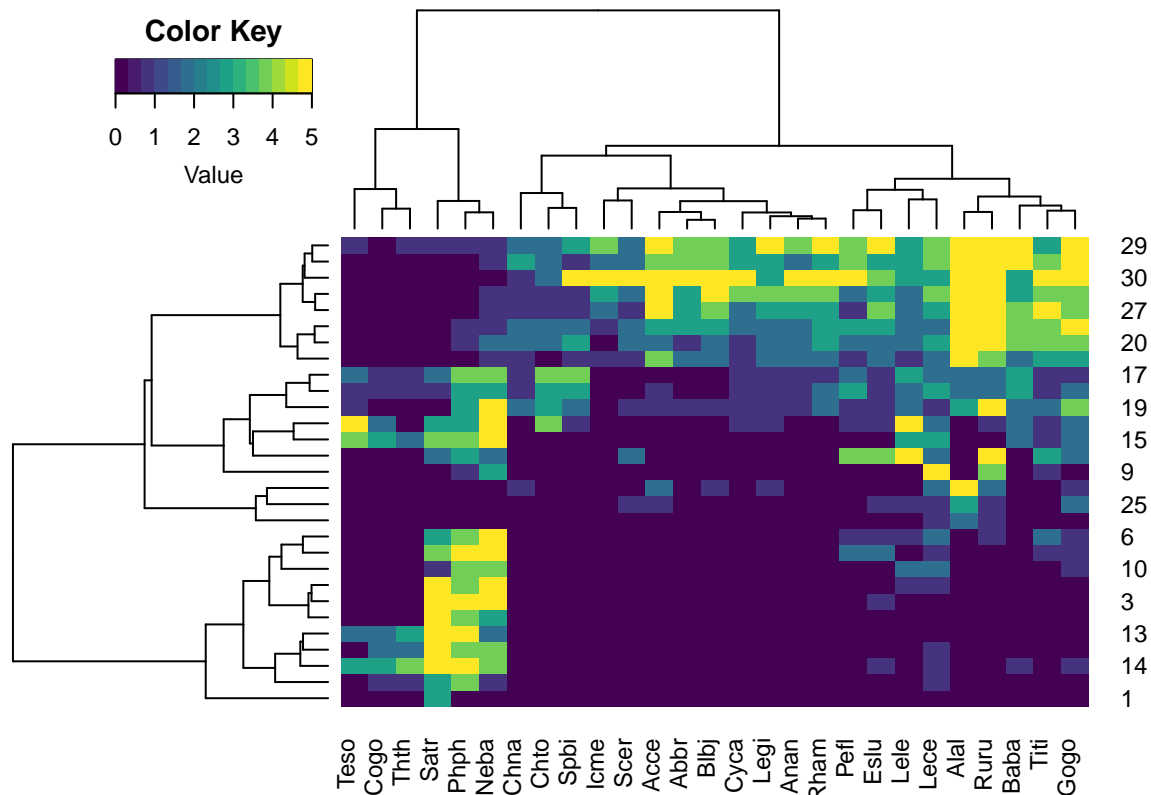
```
fish.ward <- hclust(fish.db, method = "ward.D2")

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: While we cannot make any firm conclusions on ecology, evolutionary history, etc. using only beta-diversity metrics, I hypothesize that fish species located more closely together in the Doubs reach are more closely related than species that are farther apart because there are more species shared amongst closer sites than farther apart.

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)
print(fish.pcoa)
```

```
## $points
##      [,1]      [,2]      [,3]
## 1 -0.32043291  0.135094526  0.610444135
## 2 -0.50271272 -0.053204485  0.115084582
## 3 -0.47207593 -0.075509864  0.025191639
## 4 -0.31629787 -0.065793925 -0.054743001
## 5  0.06659050 -0.103685307 -0.061632012
## 6 -0.24164704  0.022330383 -0.117689415
```

```
## 7 -0.44197058 0.013721347 -0.019578989
## 9 -0.02864444 0.138633466 -0.189715336
## 10 -0.29305689 0.037682458 -0.202079336
## 11 -0.43875065 0.071482520 0.153768001
## 12 -0.47279103 -0.013160075 -0.006637071
## 13 -0.47550783 -0.100573921 0.079082050
## 14 -0.37472705 -0.070111200 -0.076607528
## 15 -0.24370479 -0.143518516 -0.131459333
## 16 -0.07348942 -0.157547743 -0.149621439
## 17 0.04061679 -0.130989703 -0.133635222
## 18 0.14445084 -0.101088883 -0.146573604
## 19 0.22960480 -0.102509909 -0.156586724
## 20 0.40100791 -0.101317037 -0.022574780
## 21 0.43837569 -0.111705508 0.048669123
## 22 0.46516232 -0.124748909 0.094770962
## 23 0.11656630 0.659234875 -0.006196177
## 24 0.32004280 0.418009171 -0.030531755
## 25 0.22601066 0.463713151 -0.110983530
## 26 0.43466174 0.005166762 0.006723518
## 27 0.46255346 -0.090842861 0.086824252
## 28 0.46286016 -0.106465286 0.108084941
## 29 0.40942763 -0.173891275 0.123971141
## 30 0.47787757 -0.138404251 0.164230911
##
## $eig
## [1] 3.695331e+00 1.098472e+00 7.104740e-01 4.149729e-01 3.045604e-01
## [6] 1.917884e-01 1.569703e-01 1.319099e-01 1.294251e-01 8.667896e-02
## [11] 4.615780e-02 3.864487e-02 2.745800e-02 1.306508e-02 7.087896e-03
## [16] 4.039469e-03 1.300594e-03 9.520650e-18 -3.534426e-05 -3.940676e-03
## [21] -8.956051e-03 -1.461149e-02 -1.598905e-02 -2.145686e-02 -3.017013e-02
## [26] -3.432671e-02 -3.760052e-02 -6.037087e-02 -6.880061e-02
##
## $x
## NULL
##
## $ac
## [1] 0
##
## $GOF
## [1] 0.7484133 0.7798263
```

```
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)
print(sum.eig)
```

```
## [1] 81.3
```

```
par(mar = c(5, 5, 1, 2) + 0.1)
```

```
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)

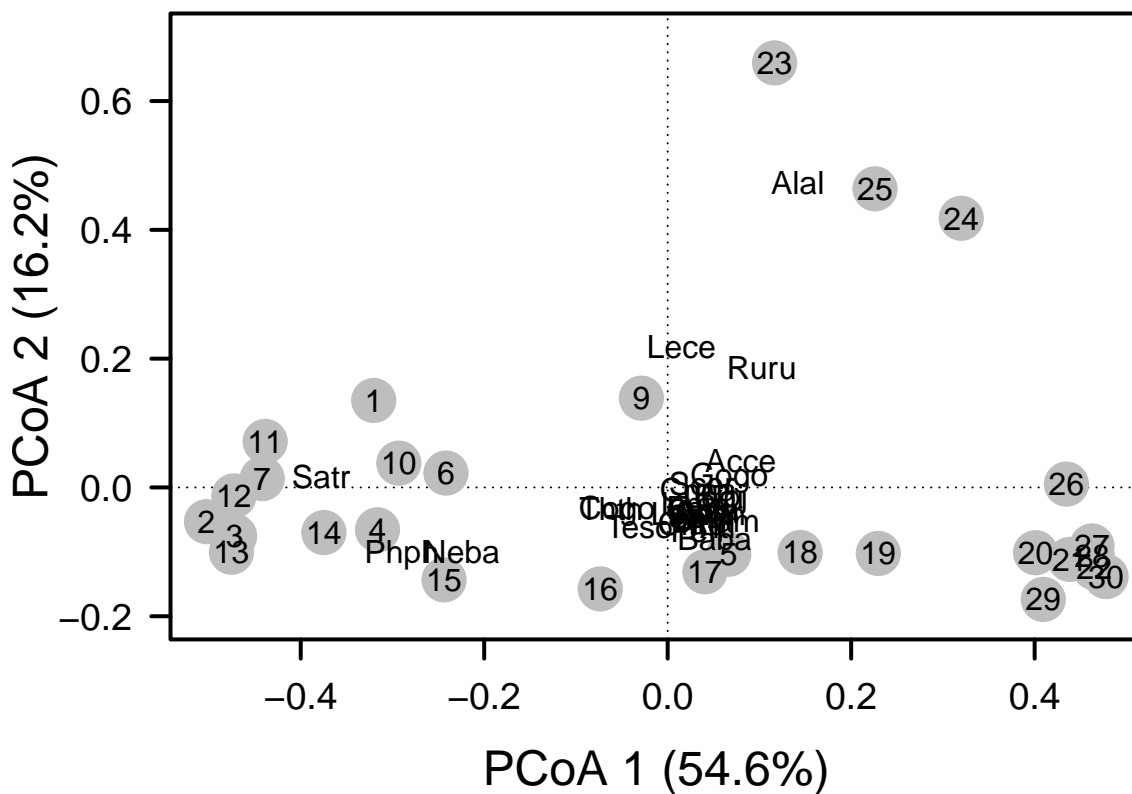
axis(side = 1, labels = T, lwd.ticks = 2, cex.axis = 1.2, las = 1)
axis(side = 2, labels = T, lwd.ticks = 2, cex.axis = 1.2, las = 1)
abline(h = 0, v = 0, lty = 3)
box(lwd = 2)

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))

fishREL <- fish
for(i in 1:nrow(fish)){
  fishREL[i, ] = fish[i, ] / sum(fish[i, ])
}

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

```

```
##          Dim1      Dim2      Dim3
## Phph -0.8674640 -0.1699316 -0.12463098
## Neba -0.7674114 -0.1855678 -0.36963830
## Rham  0.8088751 -0.4192567  0.14136301
## Legi  0.8201759 -0.1701803  0.12423941
## Cyca  0.7595122 -0.4442926  0.17313658
## Abbr  0.7704744 -0.3452714  0.29277803
## Acce  0.7635195  0.2155765  0.10288179
## Blbj  0.8118483 -0.1324698  0.25581178
## Alal  0.4471283  0.8119843 -0.05167131
## Anan  0.7974122 -0.3918972  0.20944968
```

```
fit <- envfit(fish.pcoa, fishREL, perm = 999)
print(fit)
```

```
##
## ***VECTORS
##
##          Dim1      Dim2      r2 Pr(>r)
## Cogo -0.83884 -0.54438 0.2982 0.011 *
## Satr -0.99904  0.04371 0.4326 0.004 **
## Phph -0.94110 -0.33813 0.7814 0.001 ***
## Neba -0.91413 -0.40543 0.6234 0.002 **
## Thth -0.87692 -0.48063 0.2634 0.021 *
## Teso -0.44704 -0.89452 0.1700 0.080 .
## Chna  0.99707 -0.07644 0.4612 0.003 **
## Chto  0.42032 -0.90738 0.2579 0.022 *
## Lele  0.33041 -0.94384 0.0495 0.507
## Lece  0.06856  0.99765 0.3399 0.009 **
## Baba  0.54118 -0.84091 0.6752 0.001 ***
## Spbi  0.57341 -0.81927 0.4138 0.002 **
## Gogo  0.97507  0.22188 0.3753 0.001 ***
## Eslu  0.72044 -0.69352 0.1673 0.087 .
## Pefl  0.43762 -0.89916 0.3048 0.012 *
## Rham  0.72476 -0.68901 0.8301 0.001 ***
## Legi  0.93461 -0.35568 0.7016 0.001 ***
## Scer  0.98569  0.16858 0.3533 0.009 **
## Cyca  0.68181 -0.73153 0.7743 0.001 ***
## Titi  0.64378 -0.76521 0.4586 0.001 ***
## Abbr  0.77254 -0.63497 0.7128 0.001 ***
## Icme  0.75626 -0.65427 0.5270 0.002 **
## Acce  0.88799  0.45986 0.6294 0.001 ***
## Ruru  0.48379  0.87518 0.5177 0.003 **
## Blbj  0.95802 -0.28671 0.6766 0.001 ***
## Alal  0.28755  0.95777 0.8592 0.001 ***
## Anan  0.74277 -0.66954 0.7894 0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Permutation: free
## Number of permutations: 999
```

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

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

Answer 7a: It appears that the grouping correlates to the location of the species in the reach. The upstream sites cluster in the bottom PCoA2 and left PCoA1 corner of the graph, whereas the most downstream sites cluster on the far left of the PCoA1 axis. As seen earlier using heat-maps and Ward's clustering, we know the beta-diversity in areas closer together are more similar and thus have more shared species. Sites 23, 24, and 25 seem to break this trend and have more variation along the PCoA2 axes than the other clusters.

Answer 7b: Shown much earlier in the species richness plots, we saw there was very little richness in the sites most upstream (1-15) and midstream (23-25). This may correlate to poorer river quality. We also see in our PCoA that these sites tend to cluster separately from the more species rich sites. The species *Salmo trutta fario* (Satr), *Phoxinus phoxinus* (Phph), *Nemacheilus barbatulus* (Neba) seem to majorly influence the clustering of the first 15 sites, whereas *Alburnus alburnus* (Alal) seems to influence the clustering of sites 23-25. I therefore hypothesize that these species may be indicators of river quality.

SYNTHESIS

Load the dataset from that you and your partner are using for the team project. Use one of the tools introduced in the beta diversity module to visualize your data. Describe any interesting patterns and identify a hypothesis is relevant to the principles of biodiversity.

```
Fungi_path <- "/cloud/project/QB2025_Spencer/MAT_fungal_abundances.txt"
Fungi_data <- read.table(Fungi_path, header = TRUE, sep = "\t", stringsAsFactors = TRUE)
```

```
#install.packages(dplyr)
install.packages("tidyr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
## (as 'lib' is unspecified)
```

```
require(dplyr)
```

```
## Loading required package: dplyr
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
## filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## intersect, setdiff, setequal, union
```

```
require(tidyr)
```

```
## Loading required package: tidyr
```

```
Fungi_path <- "/cloud/project/QB2025_Spencer/MAT_fungal_abundances.txt"
```

```
Fungi_data <- read.table(Fungi_path, header = TRUE, sep = "\t", stringsAsFactors = TRUE)
```

```
Fungi_data$Relative.Abandance <- as.numeric(Fungi_data$Relative.Abandance)
```

```
# Slim down the data to only columns needed
```

```
slim <- Fungi_data %>%
```

```
  select(Plot, Treatment, Species, Relative.Abandance)
```

```
# Change data so that each species is a column, values are the relative.abundance, and rows are each pl
ExpandFungi_data <- slim %>%
```

```

pivot_wider(names_from = Species, values_from = Relative.Abandance, values_fill = list(Relative.Abandance = 0))

# Sort the expanded data
Sorted_ExpandFungi_data <- ExpandFungi_data %>%
  arrange(Treatment)

# Calculate Bray-Curtis distance
ExpandFun_data.db <- vegdist(Sorted_ExpandFungi_data[, -c(1,2)], method = "bray", upper = TRUE, diag = TRUE)

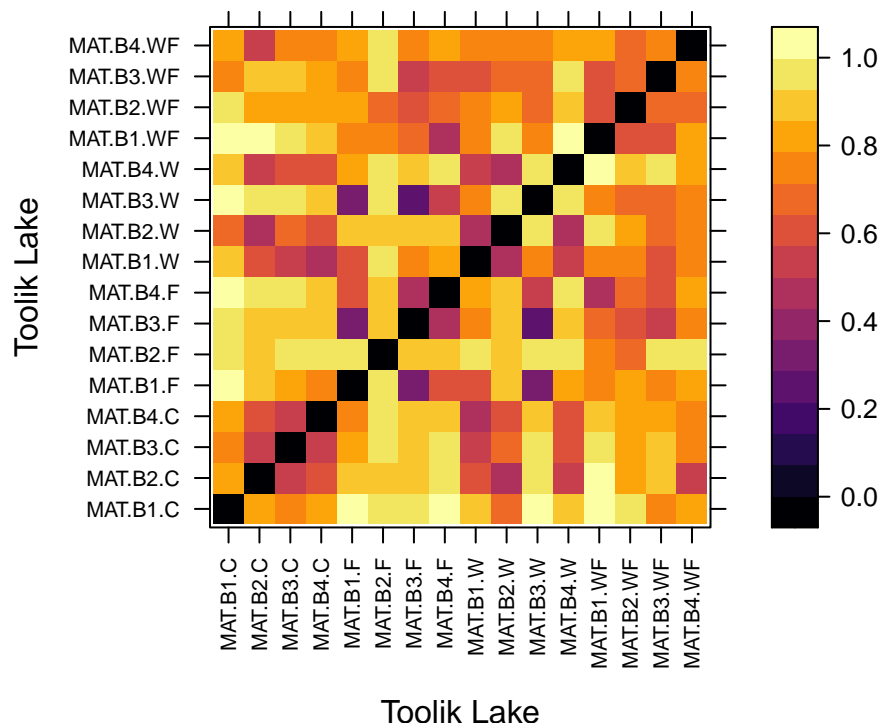
# Create a new variable and classify it as a matrix
heatmap_matrix <- as.matrix(ExpandFun_data.db)

# Assign row and column names using the "Plot" column from Sorted_ExpandFungi_data
rownames(heatmap_matrix) <- Sorted_ExpandFungi_data$Plot
colnames(heatmap_matrix) <- Sorted_ExpandFungi_data$Plot

# Plot the heatmap using the matrix created
levelplot(heatmap_matrix, aspect = "iso", col.regions = inferno,
  xlab = "Toolik Lake", ylab = "Toolik Lake",
  scales = list(x = list(rot = 90, cex = 0.7), y = list(cex = 0.7)),
  main = "Bray-Curtis Distance")

```

Bray-Curtis Distance



There are no clearly distinct patterns as many of the comparisons fall in 0.6 to 1 in similarity. There seems to be more dissimilarity between the warmed plots and the control, while the fertilized plots are extremely similar to the control. Interestingly, the combined treatment seems to vary less from control than the only warmed plots. Based on this, I would hypothesize that warming disrupts the root communities, whereas fertilizer has little to no effect on root communities.