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

Aishwarya Vaidya; Z620: Quantitative Biodiversity, Indiana University

15 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_Vaidya/Week3-Beta"
```

```
setwd("/cloud/project/QB2025_Vaidya/Week3-Beta")
require(vegan)
```

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

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

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

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

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
package.list <- c('vegan', 'ade4', 'viridis', 'gplots', 'BiodiversityR', 'indicspecies')
for (package  in package.list){
 if (!require (package, character.only = TRUE, quietly = TRUE)){
install.packages (package)
library (package, character.only = TRUE)
 }
}
```

```
##
## Attaching package: 'gplots'
```

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

```
## Warning in fun(libname, pkgname): couldn't connect to display ":0"
```

```
## BiodiversityR 2.17-1.1: Use command BiodiversityRGUI() to launch the Graphical User Interface;
## to see changes use BiodiversityRGUI(changeLog=TRUE, backward.compatibility.messages=TRUE)
```

```
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: 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.

a. 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?

Answer 2a:The fish richness is normally higher in along the downstream while relavely lower alongthe river upstream as per the plots ***Answer 2b***: Brown Trout abundance is in Doubs upstream which is quite low on fish richness whereas when fish richness is in the Doubs downstream, Brown Trout abundance is has quite low for Doubs river downstream. ***Answer 2c***: Species richness just counts how many species there are but doesn't look at how many individuals of each species are present. So, even if the richness is higher for Doubs river downstream across different sites, it doesn't show if some species are more common or rare, the latter is the case of Brown Trout abundance which is absent in Doubs river downstream. This is something which could affect the health and diversity of the ecosystem and suggests species composition matters and not just their richness.

3) QUANTIFYING BETA-DIVERSITY

In the R code chunk below, do the following:

1. 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
2. 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 check, lets 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 )
    #Remove absencesa
    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, pairwise = FALSE)
```

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

```
gamma_diversity <- length(which(colSums(doubs$fish) > 0))
gamma_diversity
```

```
## [1] 27
```

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

a. Describe how local richness (α) and turnover (β) contribute to regional (γ) fish diversity in the Doubs.
b. Is the fish assemblage at site 1 more similar to the one at site 2 or site 10?
c. 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: By using the above function for beta diversity across the entire doubs river dataset and gamma diversity across the entire dataset, we get beta as 2.16, while gamma as 27(suggesting tere are 27 different speciees in the dataset). now using the formula, alpha = gamma/beta, we get alpha across the entire dataset as 12.5.Since alpha diversity is high in this case, i.e numbner of species across a site is higher, while beta diversity is moderate i.e species turnover or composition of species across sites in a particular area is not high, it suggests there is a fair chance of overlap between different sites. Gamma diversity is slightly closer to alpha diversity suggesting same

species are widespead across the sites of doubs river and that the variation in composition between sites is present however lower.

Answer 3b:Since the beta diversity beteen sites 1 and 2 is 0.5, there is soe overlap between the similarities between the two sites some some species within the sites are also a little distinct. The beta diversity between sites 1 and 10 is 0.74 suggesting the species between these sites are quite distinct from one another with more species variation than that found for between sites 1 and 2.

Answer 3c: Beta = gamma/alpha gives a factor that is a part of gamma, putting this differently, it gives a measure of beta as a specific multiple of alpha which contributes to gamma diversity giving a relative difference between regional and local diversity of species. On the other hand, beta = gamma-alpha gives a measure beta, which is a subset of gamma and suggesting at the same time that alpha is significantly different from to beta, in a way such that changes in alpha are mutually exclusive to beta but affect gamma only as it is an additive property. When in reality the former is true, where many alpha's together (species variation in sites) contribute to beta diversity(local species variation) and many beta's together contibute to gamma diversity(regional species variation). So the concept of beta diversity independent of apha in the latter is not true as it is pointing out which species are absent rather than about their turnover as is the case in the former.

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: Incidence matirix is looking at presence or absence of a particular species without giving importance to how abundant the species is. Thus even rare species that are present only once hold similar significance to dominant species that occur several times within the site suggesting that this incidence based matrix inflates biodiversity at times due to presence of rare species having similar importance as dominant species. While an abundance based matrix is looking at abundance based data where a rare species that was considered once was not given importnce as it is only looking at abundant species data and therefore undervalues the species difference due to a few rare species.

In the R code chunk below, do the following:

1. 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, ] #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 Sorensen
fish.ds <- vegdist (fish, method = "bray", binary = TRUE)
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
##             9         10         11         12         13         14         15
## 2
## 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
```

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

a. Does the resemblance matrix (`fish.db`) represent similarity or dissimilarity? What information in the resemblance matrix led you to arrive at your answer?
b. 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:Here fish.db represents a dissimilarity matrix as the highest value is 1 with many samples while the diagonal of numbers with themselves does not generate any value which suggests the numbers are probably 0. Since for instance row 9 generates an output 1 with column 1 or even row 30 generates an output of 1 with column 13 suggets that these pairs have very dissimilar species composition with each other. While a smaller output of 0.136 that row 27 generates with column 21 suggests a very similar species composition between the two sites.
Answer 5b: The correlation for Jaccard(fish.ds) seems quite opoosite to that of Bray Curtis. In case of the former, many correlations between sites generated a value of 0 suggesting very dissimilar species composition within the sites while a larger number suggests more similar species.

4) VISUALIZING BETA-DIVERSITY

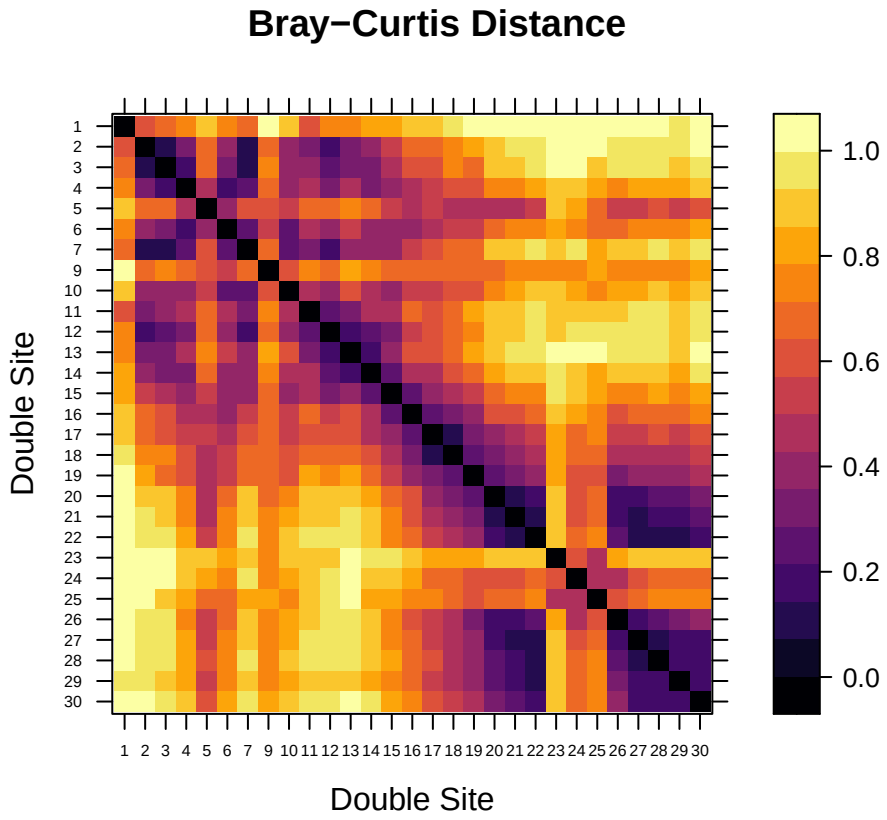
A. Heatmaps

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.

Answer1: When making heatmaps, there is a colour palette that is used to create gradients within the Heatmap. Viridis has 4 different colour palettes which are colourblind friendly. An inferno colour scale is used in this case

```
#

#Define order
order <- rev(attr(fish.db, "Labels"))
#Define Heatmap
levelplot(as.matrix(fish.db) [, order], aspect = "iso", col.regions = inferno, xlab = "Double Site", yl
```

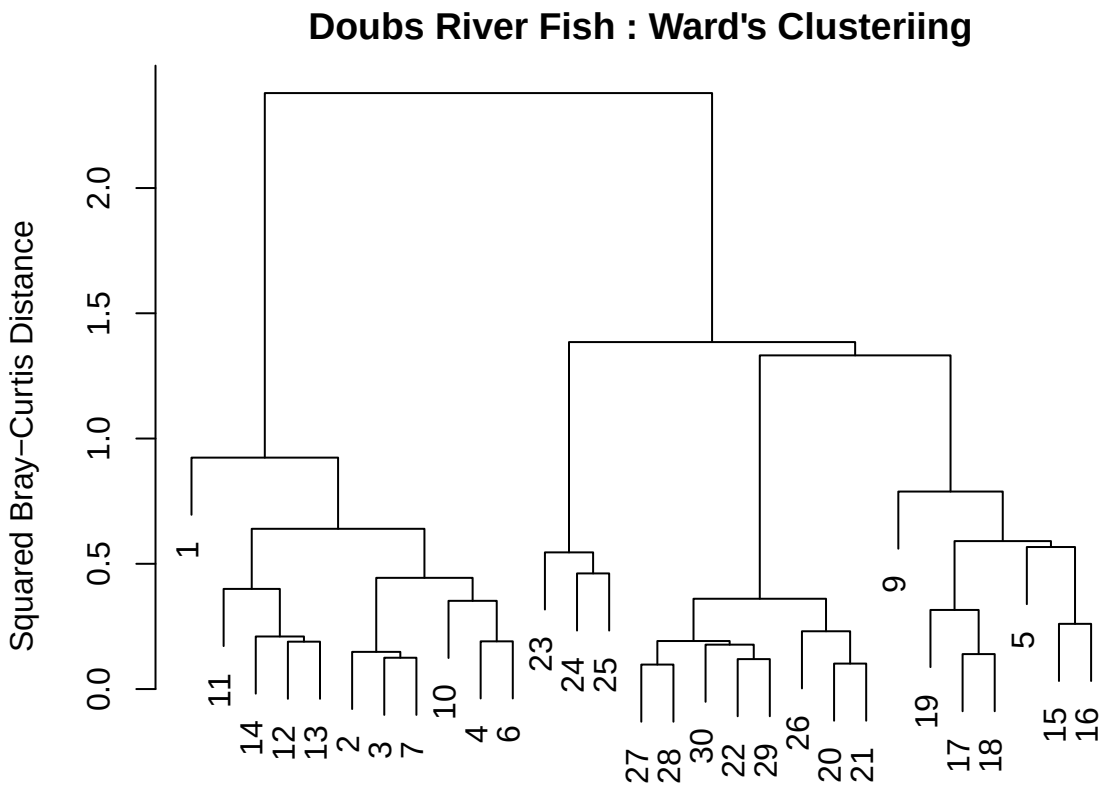


B. Cluster Analysis

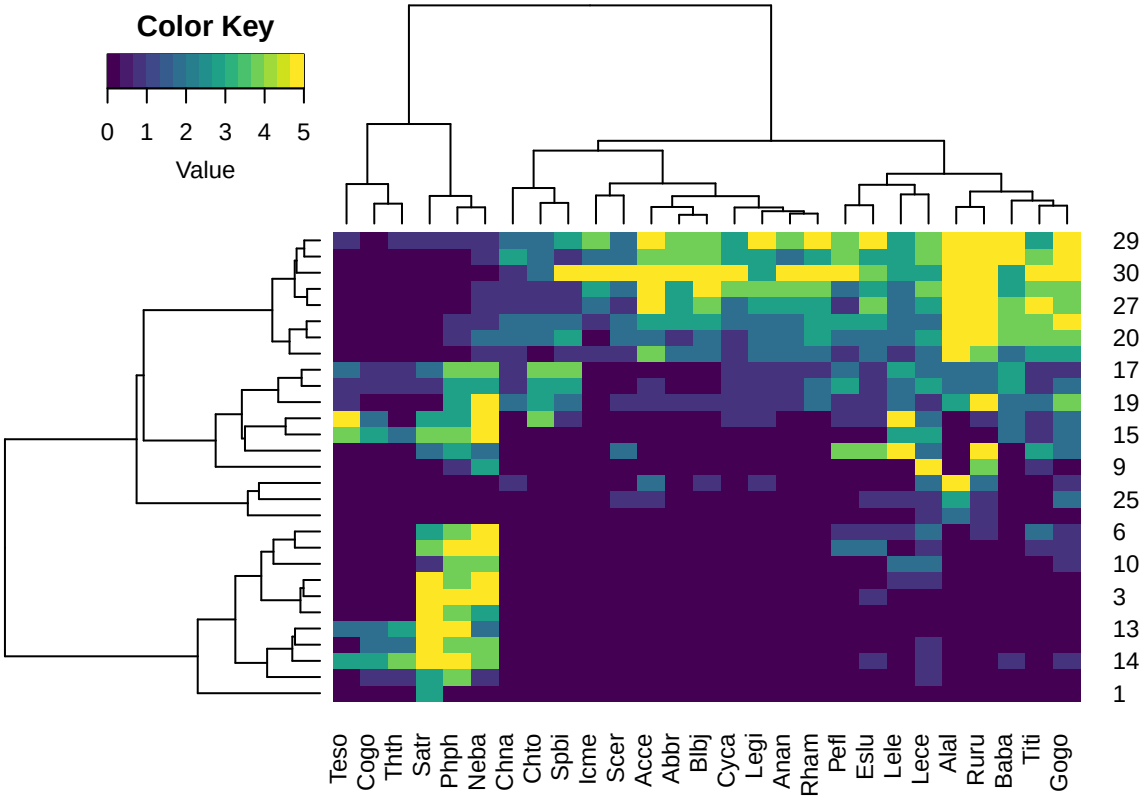
In the R code chunk below, do the following:

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 Clusteriing", ylab = "Squared Bray-Curtis Distance")
```




```
#Plot as heatmap2
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***: Do species composition change from upstream to downstream based on predator-prey relationship, based on need and competition for resources and also based on te adjoining habitat whether the riverbed is in disturbed or undisturbed environemnt that is representative of survived fish species.

C. Ordination

Principal Coordinates Analysis (PCoA)

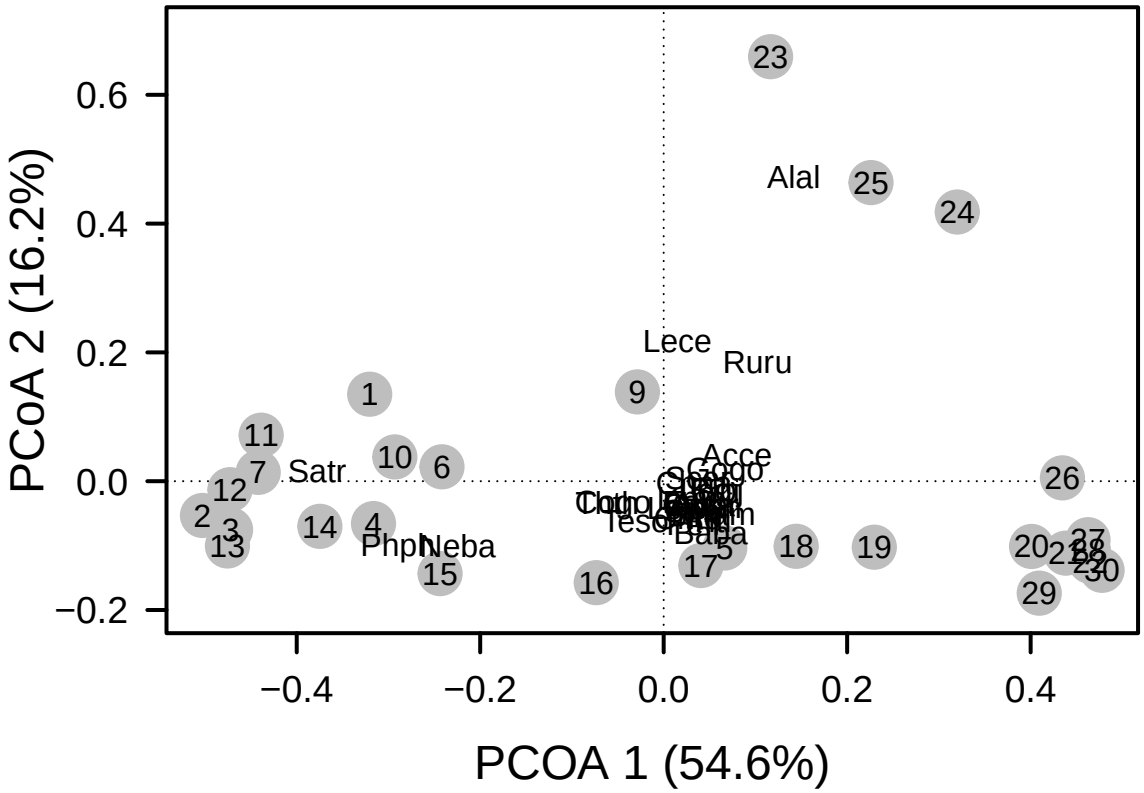
In the R code chunk below, do the following:

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.

```
`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)
  }

#Perform PCoA
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)
```

```r
#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 and 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))
#Calculate the relative abundance 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 score
fish.pcoa <- add.spec.scores.class(fish.pcoa,fishREL, method = "pcoa.scores")
text(fish.pcoa$cproj[ ,1], fish.pcoa$cpro [ ,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.

```
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)
  }
ordiscores <- fish.pcoa$points
add.spec.scores.class <- function(ordi, comm, method="cor.scores", multi=1, Rscale=FALSE, scaling="1") {
  # Check if ordi has 'points' (PCoA case) or requires scores()
  if ("points" %in% names(ordi)) {
    ordiscores <- ordi$points  # Extract site scores manually
  } else {
    ordiscores <- scores(ordi, display="sites")  # Default for vegan ordination
  }

  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)
    constant <- ((nrow(comm) - 1) * tot)^0.25
    ordiscores <- ordiscores * (nrow(comm) - 1)^-0.5 * tot^-0.5 * constant

    p1 <- min(p, length(eigenv[eigenv > 0]))
    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 && scaling == "2") {
      percen <- sqrt(eigen.var / tot)
      ordiscores <- sweep(ordiscores, 2, percen, "/")
      specscores <- sweep(specscores, 2, percen, "*")
    } else {
      specscores <- specscores / constant
      ordiscores <- ordi$points
    }

    ordi$points <- ordiscores
    ordi$eig <- eigen.var
    ordi$eigen.total <- tot
    ordi$R.constant <- constant
  }

  specscores <- specscores * multi
  ordi$cproj <- specscores
  return(ordi)
}

spe.corr <- add.spec.scores.class(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, ]
imp.spp <- spe.corr[abs(spe.corr[ ,1]) >= corrcut | abs(spe.corr[, 2]) >= corrcut, ]


#Permutation Test for Species Abundance Across Axes
fish.pcoa_vegan <- vegan::rda(fish.pcoa$points)
fit <- envfit(fish.pcoa_vegan, 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:The groups on the negative of x axis that are clubbed together probably beong to the upstream as it suggests quite similar sites hinting at a homogenous population perhaps, while on the right side i.e in the positive quadrant of x axis, the species seem quite widely spread suggestive of a varied population. Some populations like 9, 5, 17, 16 that are closer to the 0 of the x axis suggest probably a bit of an overlap between the two groups but being dominant on the side in which they are present.
Answer 7b: Fish positively correlated with PCOA1 axis(or x axis) could have developed more resistance to the outside environment probably evidenced by their presence of quite varied species and diverse species scattered in different areas based on avaiability of resources and also favourable conditions in different sites.

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.

The habitat based on Ward's clustering and PCoA plot seemed like the microbacterial population representing each of the habitat was similar and overlapping since they were quite closely placed to a different group within the habitat. Hypothesis could be that there is no significant difference between the population of microbacterial species solely based on habitat as per bray curtis distance.