

# Week 3 Exercise

*Z620: Quantitative Biodiversity, Indiana University*

*January 23, 2015*

In this exercise, we will explore between site biodiversity. We will start by using the methods we learned for measuring biodiversity within a single site. We will then expand this to learn how to compare communities between communities. We will also learn various statistical tools for testing hypotheses.

## A. Initial Setup

### Retrieve and Set Your Working Directory

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

```
## [1] "C:/Users/Mario Muscarella/GitHub/QuantitativeBiodiversity/Assignments/Week3"
```

```
setwd("~/GitHub/QuantitativeBiodiversity/Assignments/Week3")
```

### Install Packages

People develop different packages for certain tasks that can be carried out in the R environment. This week we will be using a few different R packages. You can use the ‘help’ function to learn more about each package. Let’s go ahead and load a few of the packages that we will be using.

```
require("vegan") || install.packages("vegan"); require("vegan")  
require("ade4") || install.packages("ade4"); require("ade4")  
require("BiodiversityR") || install.packages("BiodiversityR"); require("BiodiversityR")
```

At this point you should be familiar with **vegan**. The other packages (**ade4** and **BiodiversityR**) contain data sets that we will use this week.

### User Defined Functions

It helps to start scripts with any user defined functions that you know that you will be using. For example, throughout this assignment we will be using the **sem()** function we wrote in week 1. Let’s go ahead and define this function.

```
sem <- function(x){  
  sd(x, na.rm=TRUE)/sqrt(length(na.omit(x)))  
}
```

## Import Data

We will again be using the BCI data set (in `vegan`) for part of this weeks exercises. In addition, we will be using environmental data for the BCI plots. The `BiodiversityR` package has some BCI environmental data (`BCI.env`). There is additional data available for BCI and your Weeek3 folder has soil data for each plot (`bci.soil.txt`). We will go ahead and import all of this data now.

```
data(BCI)                                # BCI Tree Abundance (vegan)
data(BCI.env)                            # BCI Environmental Data (BiodiversityR)
BCI.soil <- read.delim ('./bci.soil.txt') # BCI Soil Data
```

## B. Alpha Diversity (review)

Last week we learned about alpha diversity. We calculated things like species richness, evenness, and diversity. However, as you will soon see, these tools are not always adequate when comparing communities

Let's go back to the BCI dataset and calculate the tree richness across all sites.

```
bci.S <- specnumber(BCI)
mean(bci.S)
```

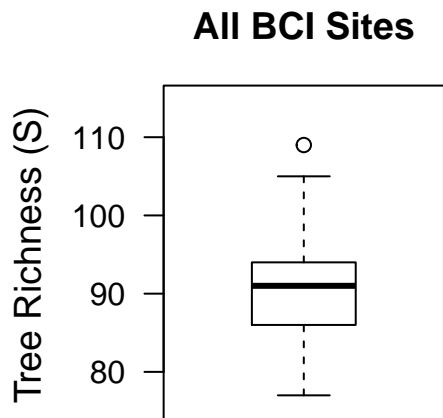
```
## [1] 90.78
```

```
sem(bci.S)
```

```
## [1] 0.9906872
```

We can visualize these data using a box and whisker plot. We are going to start our plot by changing the plotting environment using the `par()` function. Adjust the values in `mar` to see how it changes the plot. You can learn more about the diverse plotting options with the help files (`help(par)`)

```
par(mar=c(0.5, 4.5, 3, 0.5) + 0.1)
boxplot(bci.S, ylim=c(75, 115), las = 1, cex.lab = 1.2,
        main="All BCI Sites", ylab = "Tree Richness (S)")
```



However, this doesn't allow us to compare between site variation. One way we can visualize this is by separating sites. Since there aren't any factors that separate BCI sites into groups, we can just plot them based on location (based on XY-coordinates). To make it easy to visualize differences we can color code sites based on tree richness using a heatmap.

In R there are a few tools that allow you to make color palettes. Here we are going to use `terrain.colors` but you can learn about others by looking at the `Pallettes` help file `{r} help(Pallettes)`.

Let's start by making a color palette for both Richness (S) and Abundance (N)

```
BCI.S.color <- rev(terrain.colors(151))
BCI.N.color <- rev(terrain.colors(41))
```

Notice that we reversed the sequence of colors in the palette, this was done to make the colors more intuitive. You will have to make these decisions yourself so that your figure is intuitive to readers. Also, notice that we had to define the number of colors in each palette. Here we used 151 for S and 41 for N. You can adjust these values if needed. You will see soon why we used odd numbers.

## BCI Tree Richness

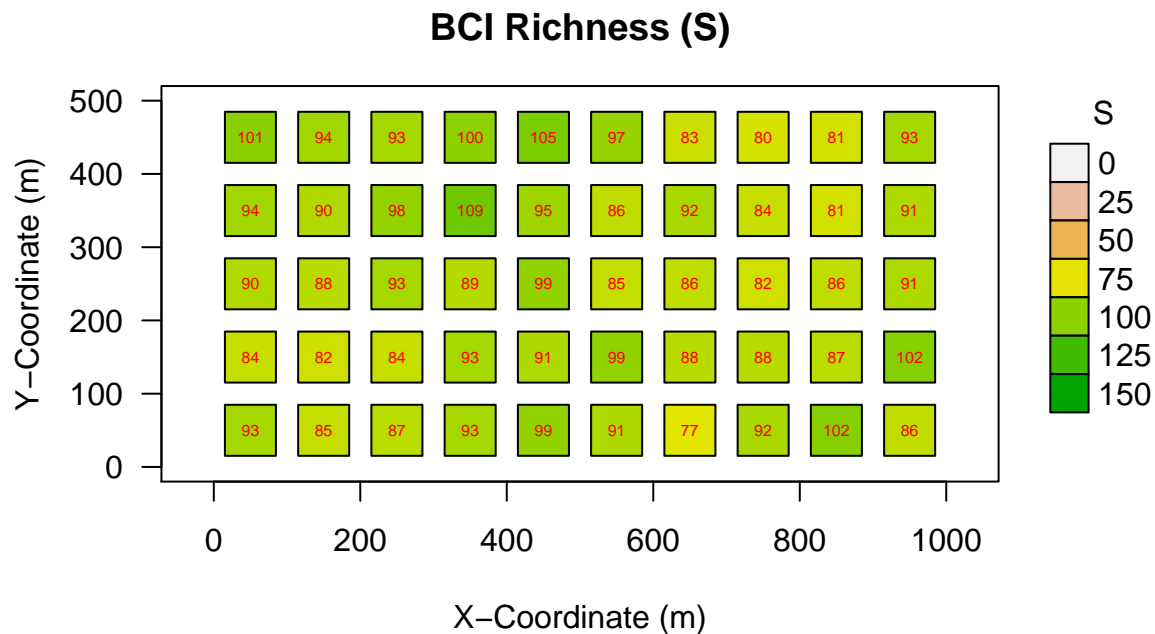
Let's start by plotting tree richness for each plot at BCI. But how do we know how these sites are arranged? Well lucky for us, the BCI soil dataset includes standardized XY-coordinates for each plot

```
head(BCI.soil, n = 2L)
```

```
##      x      y      Al      B      Ca      Cu      Fe      K      Mg
## 1 50    50 901.0908 0.79448 1680.021 6.20312 135.287 141.8813 279.1291
## 2 50   150 954.2488 0.66968 1503.365 6.03148 141.808 137.2393 280.4524
##           Mn      P      Zn      N      N.min.      pH
## 1 266.9997 1.95248 2.96948 18.46500 -3.88544 4.32432
## 2 320.4786 2.24740 2.53208 21.59896  5.64388 4.37548
```

The BCI environmental dataset actually has the UTM coordinates for each plot, but the standardized version is easier to use for our purposes. <- Add info about plotting parameters

```
# BCI Tree Richness
par(mar=c(4,4,3,4) + 0.1, xpd=TRUE)
plot(BCI.soil$x, BCI.soil$y, asp = 1, pch = 22, cex = 4, las = 1,
     bg = BCI.S.color[bci.S + 1],
     xlim = c(0, 1000), ylim = c(0, 500),
     main = "BCI Richness (S)",
     xlab = "X-Coordinate (m)",
     ylab = "Y-Coordinate (m)")
text(BCI.soil$x, BCI.soil$y, bci.S, cex = 0.5, col = "red")
legend("topright", inset = c(-0.2, 0), pch = 22, pt.cex = 3, title = "S",
     bty='n', legend = seq(0, 150, 25),
     pt.bg = BCI.S.color[seq(1, 151, 25)])
```

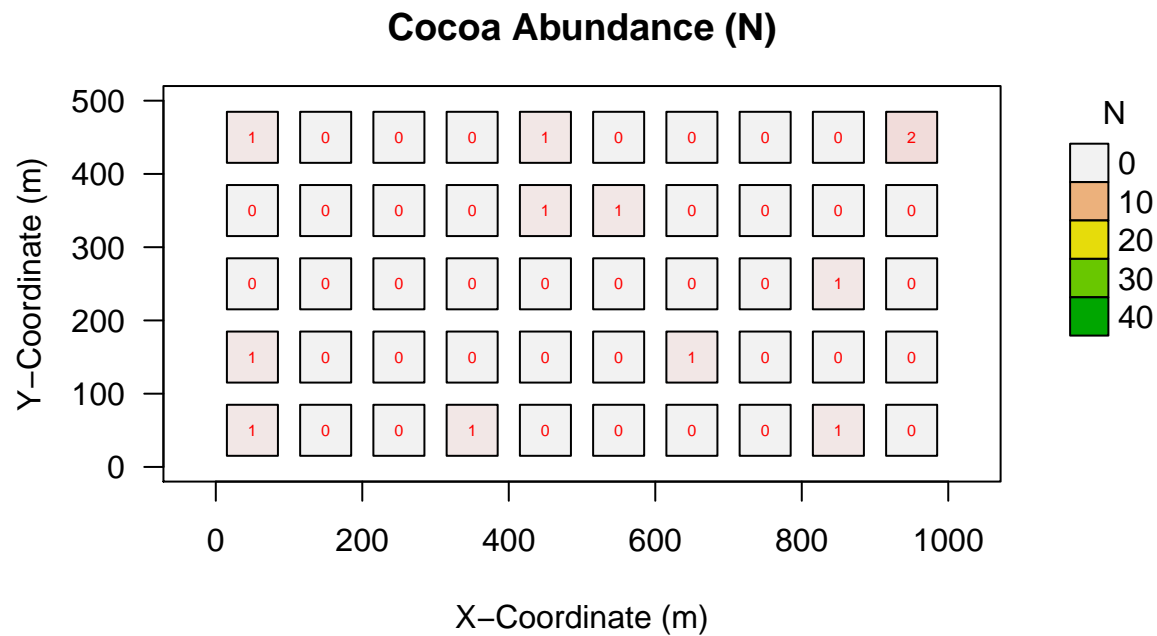


What do you notice when you compare all 50 sites? Is this a good way to compare the sites?

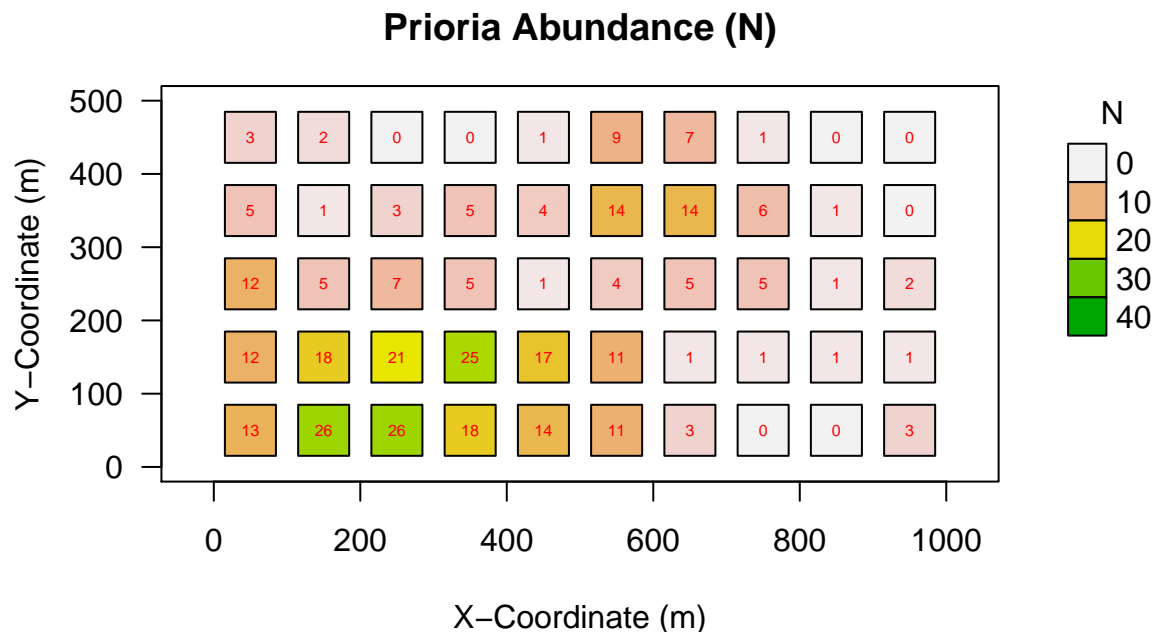
### BCI Individual Tree Abundance

What would happen if we looked at only specific tree species? Let's use a similar plotting method and just focus on the abundance of two specific trees: the Cocoa tree and the Prioria tree. We can use the abundance heatmap palette we created earlier.

```
# BCI Cocoa Abundance
par(mar=c(4,4,3,4) + 0.1, xpd=TRUE)
plot(BCI.soil$x, BCI.soil$y, asp = 1, pch = 22, cex = 4, las = 1,
     bg = BCI.N.color[BCI$Theobroma.cacao + 1],
     xlim = c(0, 1000), ylim = c(0, 500),
     main = "Cocoa Abundance (N)",
     xlab = "X-Coordinate (m)",
     ylab = "Y-Coordinate (m)")
text(BCI.soil$x, BCI.soil$y, BCI$Theobroma.cacao, cex = 0.5, col = "red")
legend("topright", inset=c(-0.2, 0), pch=22, pt.cex=3, bty='n',
     title="N", legend=seq(0, 40, 10),
     pt.bg = BCI.N.color[seq(1, 41, 10)])
```



```
# BCI Prioria Abundance
par(mar=c(4,4,3,4) + 0.1, xpd=TRUE)
plot(BCI.soil$x, BCI.soil$y, asp = 1, pch = 22, cex = 4, las = 1,
     bg = BCI.N.color[BCI$Prioria.copaifera + 1],
     xlim = c(0,1000), ylim = c(0, 500),
     main = "Prioria Abundance (N)",
     xlab = "X-Coordinate (m)",
     ylab = "Y-Coordinate (m)")
text(BCI.soil$x, BCI.soil$y, BCI$Prioria.copaifera, cex = 0.5, col="red")
legend("topright", inset=c(-0.2, 0), pch=22, pt.cex=3, bty='n',
     title="N", legend=seq(0, 40, 10),
     pt.bg = BCI.N.color[seq(1, 41, 10)])
```



What do you observe across BCI?

\*\*\* Possible Homework: Create similar plots but for Evenness and Diversity (you pick the measure, but justify)

### Doubs River Fish Abundance

Similarly, some datasets might show variation in alpha diversity across samples or sites. The doubs data set in the `ade4` package has fish abundances, environmental variables, and spatial coordinates for 30 sites in the Doubs River (runs near the France-Switzerland border in the Jura Mountains). This data set has been used to show that fish communities can be a good indicator of ecological zones in rivers and streams. We will use this data for similar purposes throughout today's exercise.

First we need to import the data from the `ade4` package. We can also look at the dataset to see what it contains. We will use the `{r}` `str()` function to see what information is in the dataset.

```
data(doubs)
str(doubs, max.level = 1) # What does max.level do?

## 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:
```

Notice that the doubs data set is actually a list with 4 components (have we worked with this data structure yet?). The first component is the environmental data for each of the 30 sites (`doubs$env`). There are 11 environmental variables in this dataset. See the help file for more information on each (including units). The second component are the abundances at each site for 27 fish species. The third component has the xy spatial coordinates for each site. The last component contains the names of each fish species.

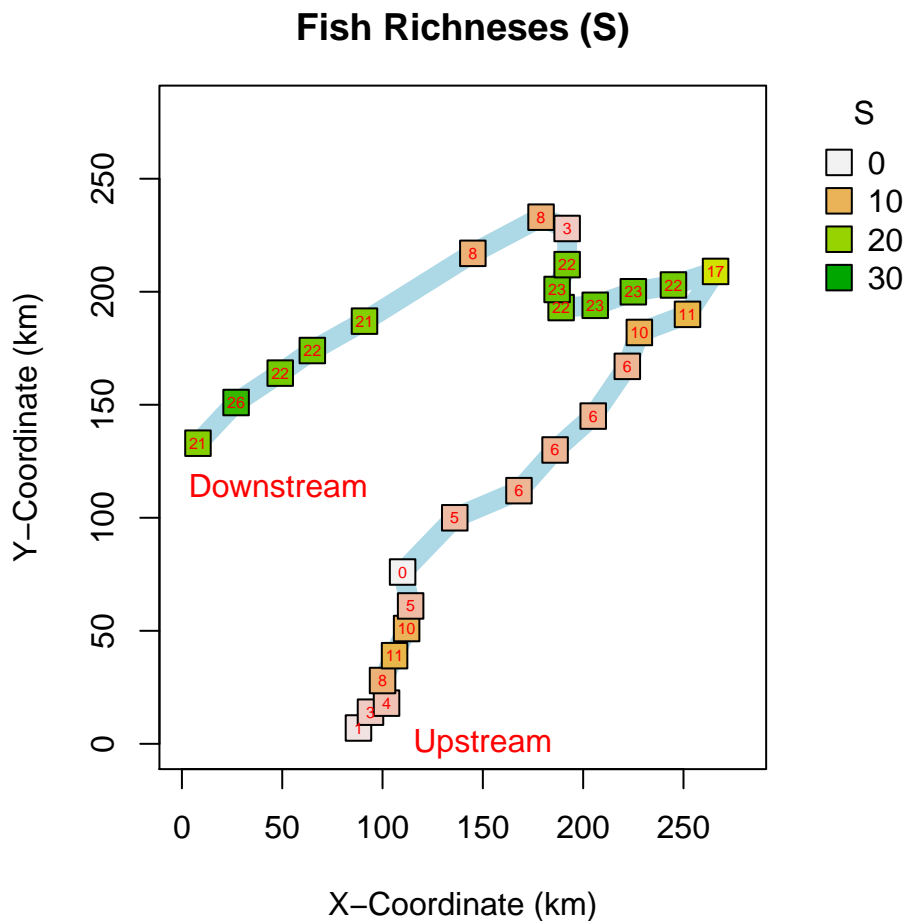
Let's plot fish richness in these stream communities. We will first need to calculate richness and define our color palettes. Once we have these items we can plot stream fish richness in a similar fashion as we did with BCI tree communities.

```

# Stream Fish
par(mar = c(4, 4, 3, 5) + 0.1, xpd = TRUE)          # Define Plot Parameters
spa.S <- specnumber(doubs$fish)                        # Calculate Richness
spa.S.color <- rev(terrain.colors(31))                # Define Richness Color Palette
spa.N.color <- rev(terrain.colors(7))                 # Define Abundance Color Palette

# Stream Fish Richness
plot(doubs$xy, type = 'l', col = "light blue", lwd = 10,
     xlim = c(0,280), ylim = c(0,280),
     main = "Fish Richneses (S)",
     xlab = "X-Coordinate (km)",
     ylab = "Y-Coordinate (km)")
points(doubs$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)])

```



```

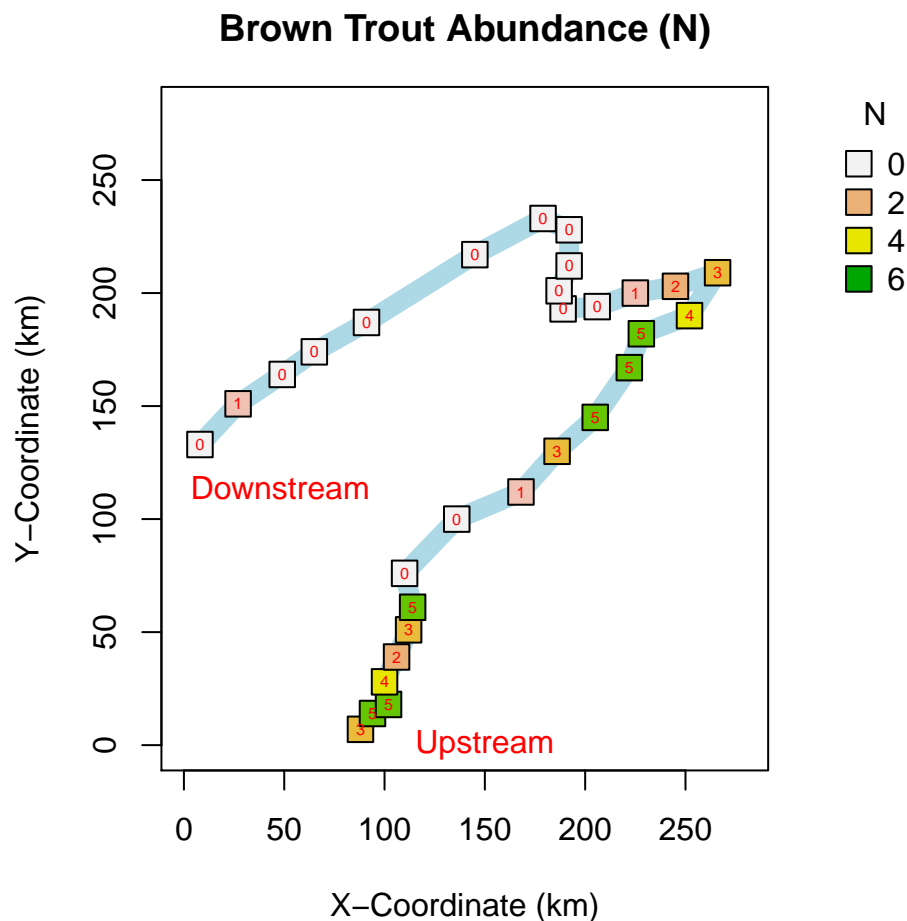
# Brown Trout Abundance
plot(doubs$xy, type = 'l', col = "light blue", lwd = 10,

```

```

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)])

```



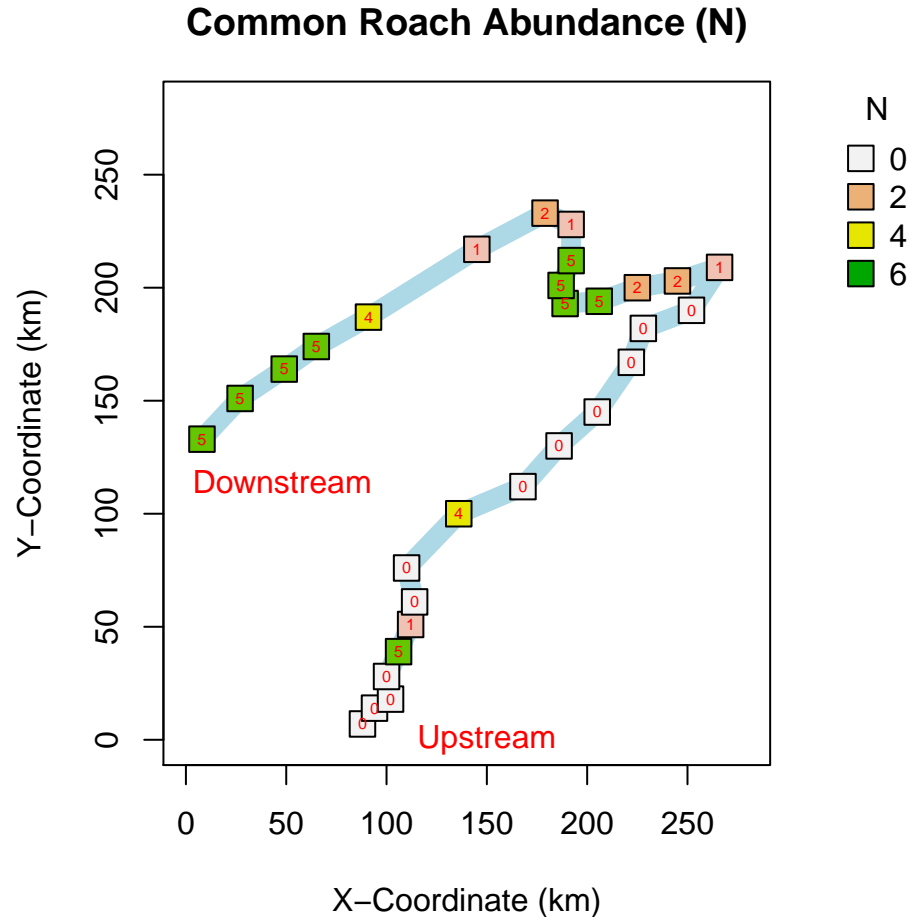
```

# Common Roach Abundance
plot(doubs$xy, type = 'l', col = "light blue", lwd = 10,
      xlim = c(0, 280), ylim = c(0, 280),
      main = "Common Roach Abundance (N)",
      xlab = "X-Coordinate (km)",
      ylab = "Y-Coordinate (km)")
points(doubs$xy, pch = 22, cex=2, bg = spa.N.color[doubs$fish$Ruru + 1])
text(doubs$xy, as.character(doubs$fish$Ruru), 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)])
```



How does this dataset differ from the BCI data? Is richness the most appropriate tool to compare communities? What does it miss?

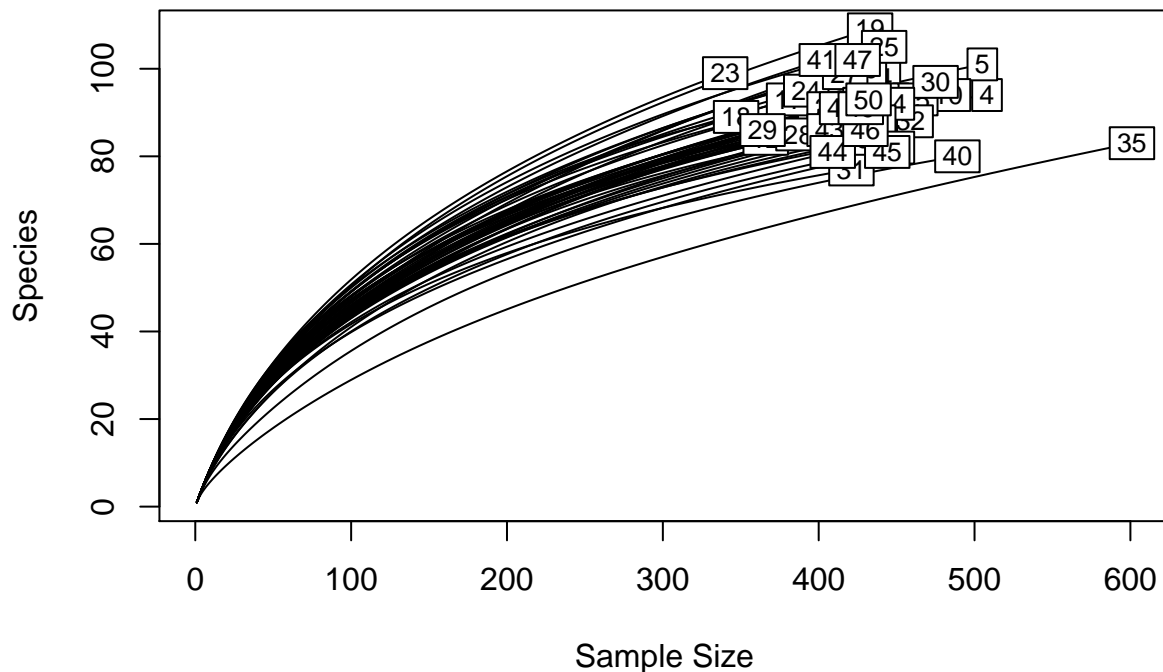
This week we are going to compare the diversity communities across sites. We will start by comparing diversity across sites. We will learn about the issues associated with making these comparisons and sampling.

\*\*\* We need a statement about sampling. See Chap 5 in Magurran

## Comparing Communities with Rarefaction

First, we can plot the number of

```
rarecurve(BCI)
```



So sampling effort is not the same for all of the plot. Remember that in the BCI plots all individuals are surveyed. However, Cannon **et al.** 1998 points out that we can easily confused richness and individual density when we make such observations. So, we can standardize the number of individuals in each plot with a technique known as rarefaction

Often, it is common to rarefy all samples to the lowest abundance

```
min(apply(BCI, 1, sum))
```

```
## [1] 340
```

```
rarefy(BCI, 340, se = TRUE)[, 1:4] # We can just look at a few to compare
```

```
##           1           2           3           4
## S  84.339919 76.531650 79.115036 82.465714
## se  2.470344  2.325785  2.708913  2.735269
```

## C. Beta Diversity

Beta diversity is a measure of between-habitat diversity. <- We need more details here

### Turnover

One way to think about beta diversity is the change in communities over time and space. This concept is generally referred to as *turnover*.

$t = \frac{b+c}{S_1+S_2}$  Is this the type of turnover we want to calculate?

## Other measures of betadiversity

One of the classic measures of  $\beta$  diversity was developed by Whittaker (1960):  $\beta_W = \frac{S}{\bar{\alpha}}$  where  $S$  = the total number of species recorded in a system and  $\bar{\alpha}$  as the average sample richness

```
S <- specnumber(colSums(BCI[1:2,]))
a.avg <- mean(specnumber(BCI[1:2,]))
B <- S/a.avg
B - 1
```

```
## [1] 0.2768362
```

$\beta_w$  ranges from 0 (minimum  $\beta$  diversity) to 1 (maximum  $\beta$  diversity). However, this measure works best for comparing two sites.

## Measures of compositional similarity

When comparing multiple communities, measures of resemblance (similarity). These measures can either be based on incidence (presence-absence; qualitative) data, or abundance (absolute or total; quantitative) data.

### 1. Incidence-Based

Index	Equation	Properties	Reference
Jaccard	$S_7 = \frac{a}{a+b+c}$		Jaccard 1900
Sørensen	$S_8 = \frac{2a}{(2a+b+c)}$		Sørensen 1948

where  $a$  = the number of shared species,  $b$  = the number of unique species in the first assemblage, and  $c$  = the number of unique species in the second assemblage

### 2. Abundance-Based

Index	Equation	Properties	Reference
Bray-Curtis Dissimilarity	$D_{14} = \frac{\sum_{j=1}^p  y_{1j} \cdot y_{2j} }{\sum_{j=1}^p (y_{1j} + y_{2j})}$		
Chord Distance	$D_3 = \sqrt{2 \left( 1 - \frac{\sum_{j=1}^p y_{1j} \cdot y_{2j}}{\sum_{j=1}^p y_{1j}^2 \cdot \sum_{j=1}^p y_{2j}^2} \right)}$	Range: 0 to $\sqrt{2}$ (two sites share the same species in the same proportions to no species in common)	Orlci 1967
Chi-Squared Distance	$D_{16} = \frac{1}{2}$		

### 3. Other Measures (may not be appropriate for species abundances) | Euclidean

#### Calculate Bray Curtis Dissimilarity

Many of these similarity metrixs are in the vegan `vegdist()` function

```
spe <- doubs$fish
env <- doubs$env

# It is always good to check your sites
rowSums(spe)      # Notice site 8 is empty

##  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
##  3 12 16 21 34 21 16  0 14 14 11 18 19 28 33 40 44 42 46 56 62 72  4 15 11
## 26 27 28 29 30
## 43 63 70 87 89

spe <- spe[-8, ]
env <- env[-8, ]

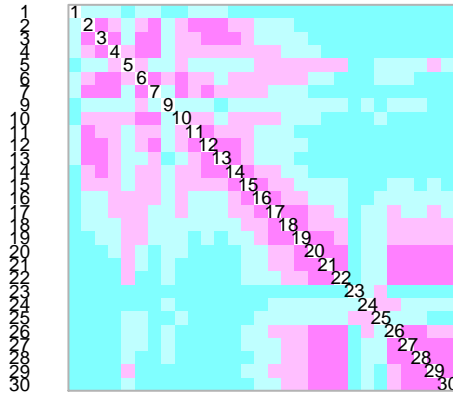
# Calculate Bray-Curtis Dissimilarity between doubs river sites
spe.db <- vegdist(spe, method="bray")

# Calculate Jaccard Dissimilarity
spe.dj <- vegdist(spe, method="jaccard", binary=TRUE)

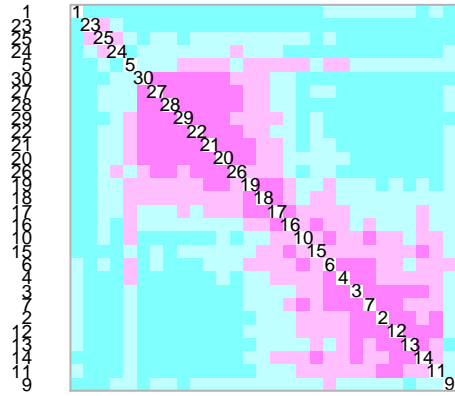
# Visualize association matrices
source("coldiss.R")
coldiss(spe.db, byrank=FALSE, diag=TRUE)

## Loading required package: gclus
## Loading required package: cluster
```

### Dissimilarity Matrix

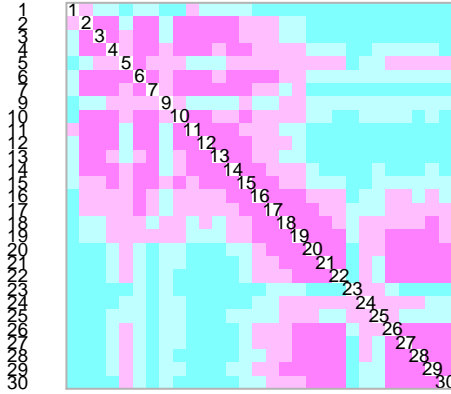


### Ordered Dissimilarity Matrix

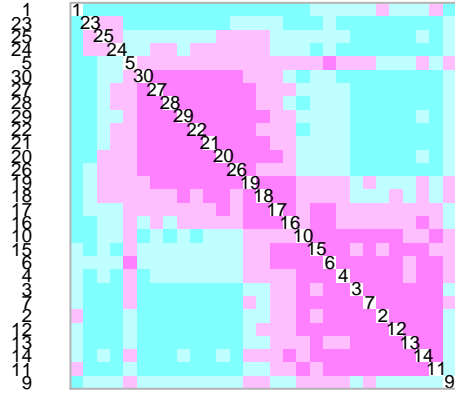


```
coldiss(spe.db, byrank=TRUE, diag=TRUE)
```

## Dissimilarity Matrix



## Ordered Dissimilarity Matrix



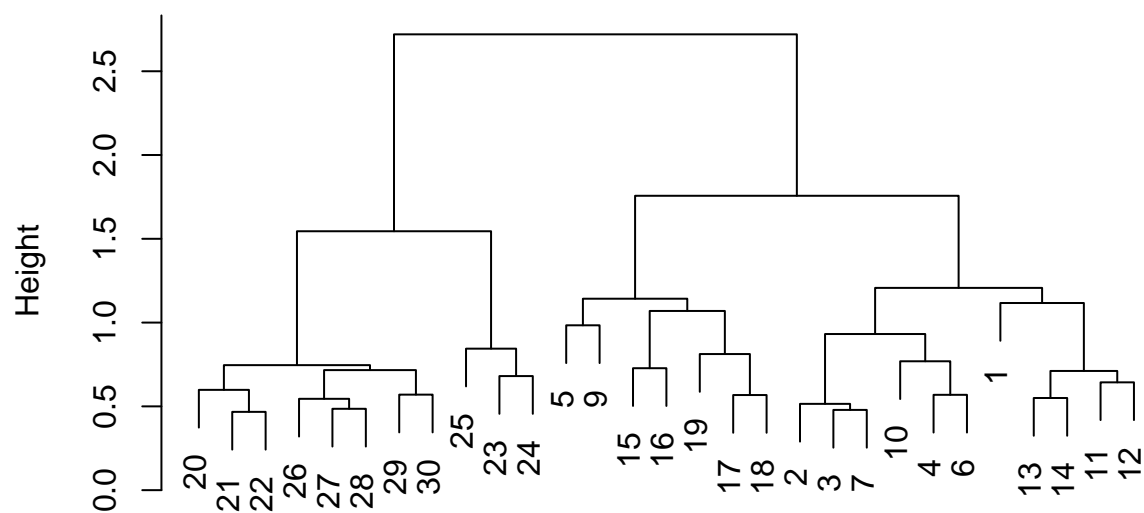
## D. Visualizaton

### Cluster Analysis

#### Hierarchical Clustering (Ward Clustering)

```
spe.norm <- decostand(spe, "normalize")
spe.ch <- vegdist(spe.norm, "euc")
spe.ch.ward <- hclust(spe.ch, method="ward.D")
spe.ch.ward$height <- sqrt(spe.ch.ward$height)
plot(spe.ch.ward)
```

## Cluster Dendrogram



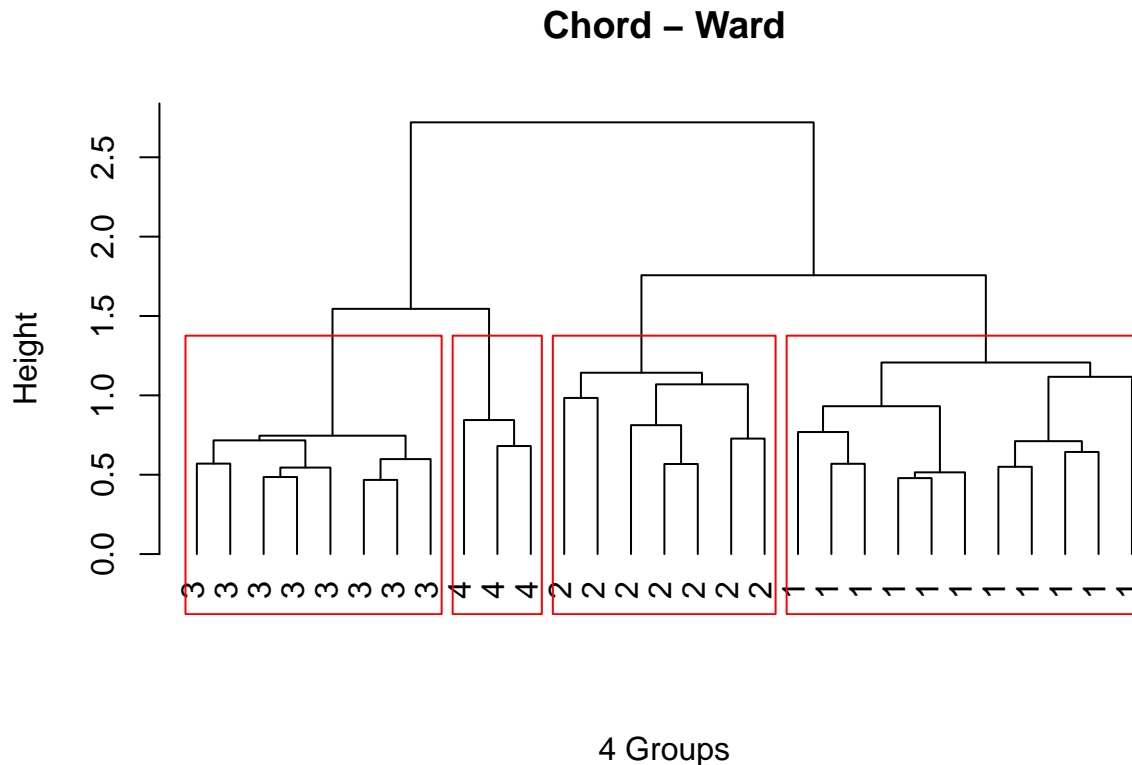
spe.ch  
hclust (\*, "ward.D")

*# Groups*

```
spe.chwo <- reorder.hclust(spe.ch.ward, spe.ch)
```

```
plot(spe.chwo, hang=-1, xlab="4 Groups", sub="", ylab="Height", main="Chord - Ward", labels=cutree(spe.chwo, k=4))
```

```
rect.hclust(spe.chwo, k=4)
```



### Visualization using Ordination

Various techniques | Method | Distance Preserved | Variables | | Principapl component analysis (PCA) | Euclidean distance | Quantiative data, linear relationships | | Correspondence analysis (CA) |  $\chi^2$  distance | Non-negative, dimensionally homogeneous quantita6tive or binary data; species frequencies or presence/absence data | | Principal coordinate analysis (PCoA), metric (multidimensional) scalling, classical scaling | Any distance measure | Quantitative, semiquantitative, qualitative, or mixed | |Nonmetric multidimensional scalling (nMDS) | Any distance measure | Quantitative, semiquantitative, qualitative, or mixed |

(REF: L&L , Table 9.1)

How does ordination work?

### NMDS

```
spe.nmnds <- metaMDS(spe, distance="bray")
```

```
## Run 0 stress 0.07477822
## Run 1 stress 0.08841678
## Run 2 stress 0.1125748
## Run 3 stress 0.1208171
## Run 4 stress 0.08844028
## Run 5 stress 0.07429335
## ... New best solution
```



```
## ... procrustes: rmse 0.02398756  max resid 0.092217
## Run 6 stress 0.07478511
## ... procrustes: rmse 0.02363684  max resid 0.09304886
## Run 7 stress 0.07429542
## ... procrustes: rmse 0.000831658  max resid 0.003953496
## *** Solution reached
```

```
spe.nmds
```

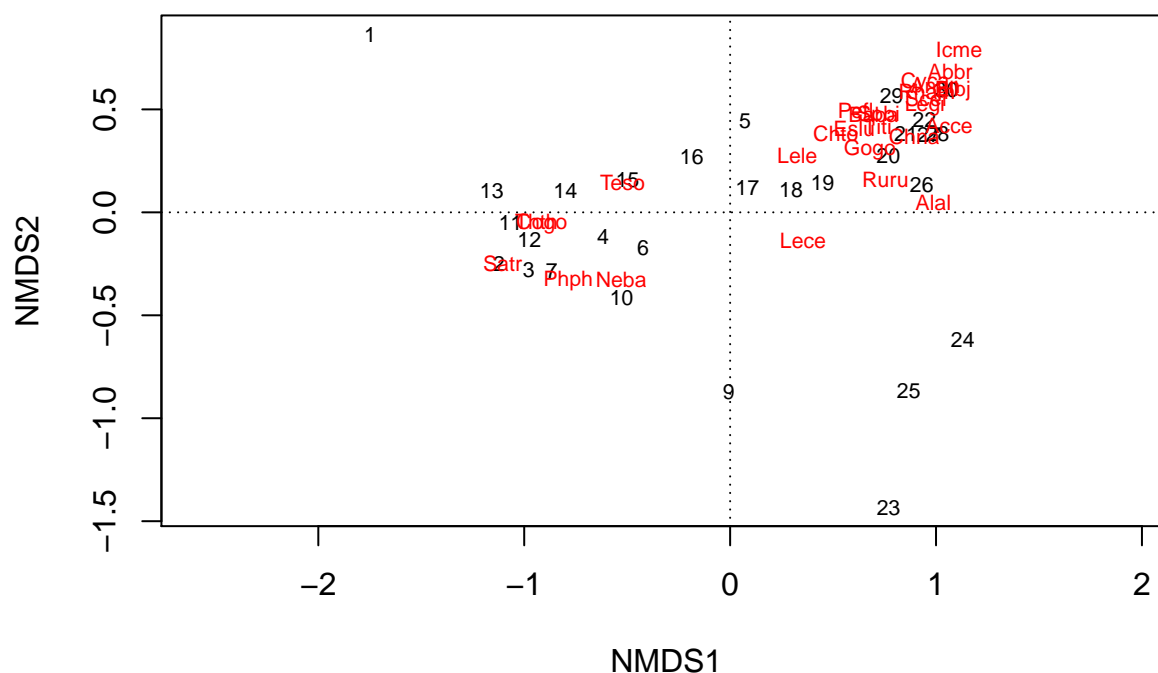
```
##
## Call:
## metaMDS(comm = spe, distance = "bray")
##
## global Multidimensional Scaling using monoMDS
##
## Data:      spe
## Distance: bray
##
## Dimensions: 2
## Stress:      0.07429335
## Stress type 1, weak ties
## Two convergent solutions found after 7 tries
## Scaling: centring, PC rotation, halfchange scaling
## Species: expanded scores based on 'spe'
```

```
spe.nmds$stress
```

```
## [1] 0.07429335
```

```
plot(spe.nmds, type="t", main=paste("nMDS/Bray - Stress =", round(spe.nmds$stress, 3)))
abline(h=0, lty=3)
abline(v=0, lty=3)
```

nMDS/Bray – Stress = 0.074



What is the nMDS stress? How is this used to judge the quality of the ordination

### Classic Multidimensional Scalling (PCoA)

```
spe.bray <- vegdist(spe, method="bray")
spe.b.pcoa <- cmdscale(spe.bray, eig=TRUE)

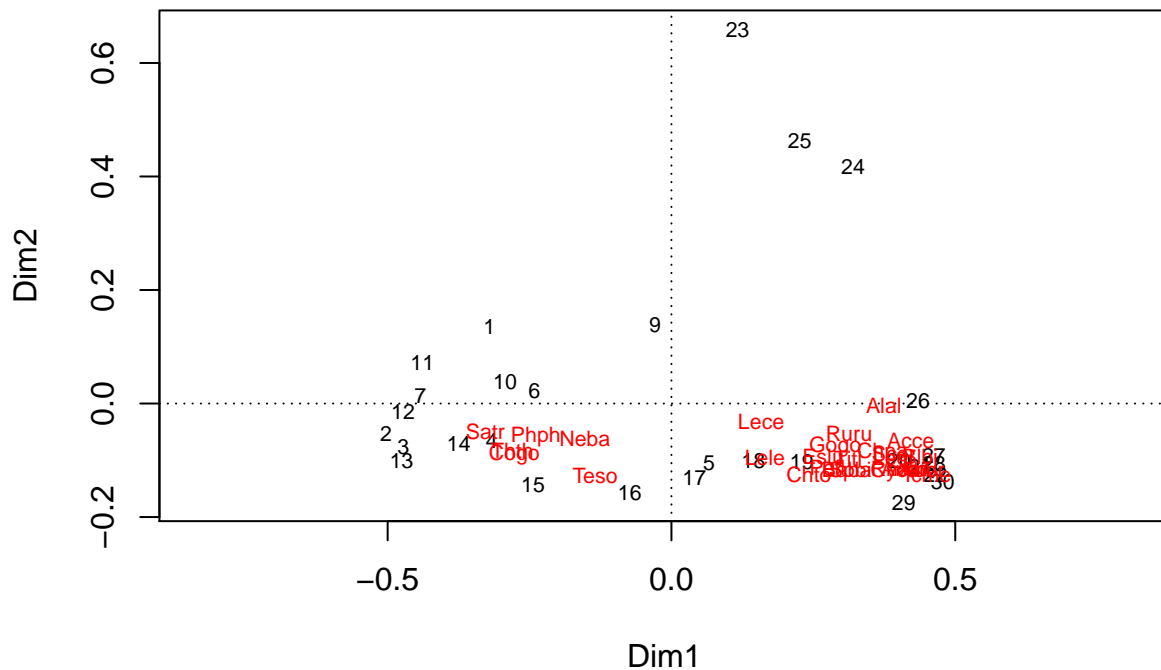
ordiplot(scores(spe.b.pcoa)[,c(1,2)], type='t', main="PCoA with species")

## Warning in ordiplot(scores(spe.b.pcoa)[, c(1, 2)], type = "t", main =
## "PCoA with species"): Species scores not available

abline(h=0, lty=3)
abline(v=0, lty=3)

# Add Species
spe.wa <- wascores(spe.b.pcoa$points[,1:2], spe)
text(spe.wa, rownames(spe.wa), cex=0.7, col="red")
```

## PCoA with species



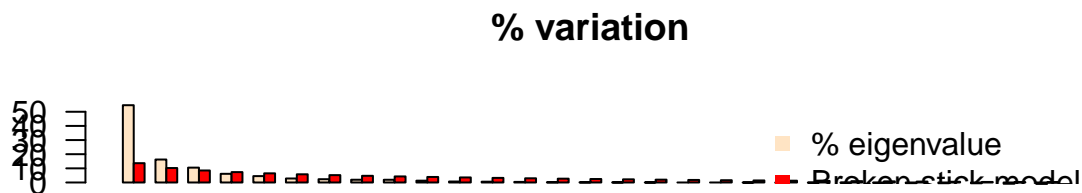
```

evplot <- function(ev)
{
  # Broken stick model (MacArthur 1957)
  n <- length(ev)
  bsm <- data.frame(j=seq(1:n), p=0)
  bsm$p[1] <- 1/n
  for (i in 2:n) bsm$p[i] <- bsm$p[i-1] + (1/(n + 1 - i))
  bsm$p <- 100*bsm$p/n
  # Plot eigenvalues and % of variation for each axis
  op <- par(mfrow=c(2,1))
  barplot(ev, main="Eigenvalues", col="bisque", las=2)
  abline(h=mean(ev), col="red")
  legend("topright", "Average eigenvalue", lwd=1, col=2, bty="n")
  barplot(t(cbind(100*ev/sum(ev), bsm$p[n:1])), beside=TRUE,
    main="% variation", col=c("bisque",2), las=2)
  legend("topright", c("% eigenvalue", "Broken stick model"),
    pch=15, col=c("bisque",2), bty="n")
  par(op)
}

evplot(spe.b.pcoa$eig)

spe.pcoa.env <- envfit(spe.b.pcoa, env)
evplot(spe.b.pcoa$eig)

```



Why not PCA?

Constrained Ordination

```
# subset explanatory variables
envdas <- env[,1]
envtopo <- env[,c(2:4)]
envchem <- env[,c(5:11)]
spe.hel <- decostand(spe, method="hellinger")

spe.rda <- rda(spe.hel ~ ., envchem)
coef(spe.rda)
```

	RDA1	RDA2	RDA3	RDA4	RDA5
pH	0.0020384098	-0.0031031275	-0.0190886957	-0.016823415	0.004127536
har	-0.0011912221	0.0007238813	-0.0066409901	-0.010576449	0.002295200
pho	0.0008030375	0.0006560255	0.0002170545	0.002050325	0.007613774
nit	-0.0011933640	0.0011698470	-0.0005397298	0.001754538	-0.001273360
amm	0.0024457337	-0.0049289180	0.0006951098	-0.004139305	-0.004732773
oxy	0.0066262263	-0.0008245053	-0.0127449854	0.006086134	-0.005810343
bdo	-0.0002256933	-0.0042878090	-0.0058318080	-0.000140698	-0.011305550
	RDA6	RDA7			

```
## pH -0.0455461425 0.0996685848
## har 0.0074031347 -0.0009897574
## pho -0.0045440954 -0.0038977199
## nit -0.0008977745 -0.0002185856
## amm 0.0193330610 0.0115315915
## oxy -0.0015043312 -0.0046358191
## bdo -0.0080555706 -0.0033015665
```

```
spechem.physio <- rda(spe.hel, envchem, envtopo)
```

```
# Permutatoin Test
```

```
anova.cca(spe.rda, step=1000)
```

```
## Permutation test for rda under reduced model
```

```
## Permutation: free
```

```
## Number of permutations: 999
```

```
##
```

```
## Model: rda(formula = spe.hel ~ pH + har + pho + nit + amm + oxy + bdo, data = envchem)
```

```
##          Df Variance      F Pr(>F)
```

```
## Model      7 0.30442 4.6102 0.001 ***
```

```
## Residual 21 0.19809
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Variance Partitioning

```
spe.part.all <- varpart(spe.hel, envchem, envtopo)
```

## Hypothesis Testing

## Homework